# Real-time Performance Evaluation of BGSLibrary Algorithms for Intelligent Surveillance

**Imane Benraya[1], Nadjia Benblidia[2], Yasmine Amara[3]**
[1,2] Electronics Departement (LATSI Laboratory),University of Saad Dahleb, Blida 09000, Algeria
[3] Electronics Department, University of Saad Dahleb, Blida 09000, Algeria

**Article Info**

***Article history:***

Received
Revised
Accepted

***Keywords:***

Background Subtraction
Moving Detection
Evaluation
Foreground
BgsLibrary

**ABSTRACT**

Background subtraction is the first and basic stage in video analysis and smart surveillance to extract moving objects. In fact, the background subtraction library was created by andrews sobral in 2012, which currently combines 43 background subtraction algorithms from the most popular and widely used in the field of video analysis. Each algorithm has its own characteristics, strengths, and weaknesses in extracting moving objects. The **e**valuation allows the identification of these characteristics and helps researchers to design the best methods. Unfortunately, the literature lacks a comprehensive evaluation of the algorithms included in the library. Accordingly, the present work will evaluate these algorithms in the bgs-library through the segmentation performance, execution time, and processor, so as to, achieve a perfect, comprehensive, real-time evaluation of the system. Indeed, a BMC (Background Modeling Challenge) dataset was selected using the synthetic video with the presence of noise. Results are presented in tables, columns, and foreground masks.



***Corresponding Author:***

Imane Benraya,
Electronics Department (LATSI Laboratory),
Blida 1 University,
09000 Blida, Algeria.
Email: benrayaimane@gmail.com

## 1. INTRODUCTION

The smart surveillance system can detect and track people, accidents, and objects. Besides, it can be also adapted under dangerous situations that threaten human security and abnormal behaviour. In fact, a smart surveillance system relies on video analysis, detection, and tracking of moving objects. It is important to know that the detection of movement in the video is the first stage to consider after the subtraction of the background, and before starting the analysis of any system.[1]

In fact, to segment moving objects in a sequence of images, a BS method is used. It is started by designing the background model of the scene. Then, a comparison to the current frame is fulfilled before updating the background model.Thus, a flexible, accurate, and reliable background model is required to obtain good tracking of the moving object. Indeed, the background template must: (a) be precise in its texture and react to changes over time. (B) Flexible under different lighting conditions. (C) Rapid real-time rendering[2][1]. A number of researches focus on developing background subtraction techniques,, [3][4] , .citing as examples the statistical models [5][6], fuzzy models, and neural models. Unfortunately, the majority of the developed methods lack precision in detecting the intriguing target of video streams.

Over the past years, several background modelling methods have been proposed to identify foreground objects in a video clip.These methods differ according to their strength and weakness in discovering objects of interest. Accordingly, The background subtraction library was created by andrews sobral in 2012, The library is a collection of background subtraction algorithms. Where the library contained 29 algorithms compiled in the C ++ programming language. Several other methods have been added and developed. Currently, the library is containing about 43 of the most popular and widely used algorithms in the field of video analysis and smart monitoring. [7], [8].

Several other research papers deal with the issue of evaluating background subtraction methods under different scenarios. The biggest challenge faced by background subtraction methods is analyzed and studied in [9],[10].

In [9] the analyses of the advantages and disadvantages of background modeling methods, and comparison of their performance in terms of quality and computational cost, using chang.net dataset, and another dataset with different environmental conditions is done. The challenges faced during the use of BS techniques are well identified and considered in [11]. In [12] a comparison of 29 methods in bgslibrary from the large group of BG, using the BMC dataset is done. The evaluation of the methods' strength and performance in terms of processor and memory is also considered. The authors in [13] evaluated nine methods of background subtraction under the influence of two types of regular and Gaussian noise. In this paper, all the algorithms in the background subtraction library were comprehensively evaluated in real-time. Among the evaluation methods, emphasis was placed on. Firstly, segmentation performance in demonstrating the ability of the algorithms to extract moving objects, by choosing the most commonly used set of measurements in the literature. Second, the execution time differs from one algorithm to another, some of the themes have a fast response time while others take a long time to respond. Third, the majority of algorithms have a constant value in memory consumption over time, and some of them have an increasing value over time. Fourth, the processor is used for a good, comprehensive, real-time evaluation. A BMC (background modelling challenge) dataset was selected using the synthetic video type with the presence of noise. Results are presented in tables, columns, and images for intro masks.

The BMC (Background Modelling Challenge) dataset is a set of evaluation data based on real and synthetic videos that present many of the challenges that researchers need in evaluating subtraction methods in the background video[14].

The rest of this article is divided as follows: Section 2 presents a state of the art on background methods, in which most of the techniques in the background subtraction library are studied. Section 3, evaluates the criteria used in this article are explored and highlighted and all steps of subtraction analysis are provided in essential details of the experiments. All algorithms simulated. A table of images of the foreground masks, and a table of metrics used in the evaluation, are presented. This section also details the results obtained in the form of bars and curves, which are discussed and analyzed in Section 4. Finally, the conclusion and future work are done in Section 5.

## 2. STATE OF THE ART

### 2.1. Basic Background Model

The main operating of the background subtraction process is to create a background template without any moving objects to compare it to serial frames. The basic background is modeled using the average , the median or the histogram analysis over time[15], Several studies are developing methods to achieve this goal,where they dimonstrate that each method has its drawbacks and advantages. First, the background can be set manually using static image free of moving object. Then, the difference between the current image and static image is calculated. This method is named Static Frame Difference , it is simple to implement , however, it fails in extracting the foreground during high light changes or when the moving object stop suddenly [12]. [12]. Frame Difference is another method to create a background template. It is relatively simple to analyze the difference between adjacent frames in the sequence of images in the video . The illumination does not affect the extraction of the moving object. However, it is impossible to extract complete information when the target is moving, or, when the background color is similar to the target color, where, the information in the moving target is considered as background information [16], the arithmetic mean of pixels between two consecutive images is used to configure and maintain the background model in [17]. The background model (B) can be obtained as [15].

$$B = \frac{1}{l}\sum_{t=1}^{l} Z_t \quad (1)$$

where l is the length containing gray scale images Zi. The background must be maintained with the following serial process.

$$B_t = (1-\alpha)B_{t-1} + \alpha Z_t \quad \alpha \in [0,1] \subset \mathbb{R} \quad t \in \{1, l\} \subset \mathbb{Z} \quad (2)$$

This method is called Adaptive background learning .The feature of this algorithm is adaptive maintenance during scene changes. However, some pixels of the moving object are updated with the background [15].

### 2.2. Statistical Models Based On Gaussian Distributions

Several static models of background extraction based on Gaussian distributions are develepod such as Pfinder [18], ElgammalKDE 2000 [19], Stauffer and Grimson GMM[6], ZivkovicGMM2004 2006 [20] [21], Lopez-RubioAE. Where, Pfinder, Stauffer and Grimson are the most popular and adapted models.

Pfinder[18] used to model the color of each object through the Gauss distribution of each pixel of the image. A class $a$ is calculated for each background and foreground point by the following equations.

$$P_a(x_t) = \eta(x_t, \mu_{a,t}(x), \textstyle\sum_{a,t}(x)) \quad (3)$$

$$\eta(X, \mu, \textstyle\sum) = (2\pi)^{-D/2} \det(\textstyle\sum)^{-1/2} \quad (4)$$

where, Probability is defined by $P_a(x_t)$ , $x$ is the pixel, $a_t(x)$ is the class. If the probability limited to background class is greater than threshold $P_b(x_t) > T$ The pixel belongs to the background.

The Stauffer and Grimson model[6] is based on a parametric probabilistic background. For scene modeling, it uses a mixture of $K$ Gaussian distributions , k is chosen between 3 and 5. The probability of pixel color is given by.

$$P(x_t) = \textstyle\sum_{i=1}^{K} w_{i;t}\, \eta\left(x_t, \mu_{i,t}(x), \textstyle\sum_{i,t}(x)\right) \quad (5)$$

Where $x_t$ is the pixel, number distribution Gaussian is K, $w_{\mathrm{i,t}}$ is the probability of k Gaussian at time t, $\mu_{i,t}$ is the average, $\sum_{i,t}$ is the covariance matrix , $\eta$ is the probability distribution function of the Gaussian k.

The authors on [20] and [21]improved the Model [6] where they used mixture distribution with unstable components. The number of components is $K_t^x$ for a pixel $x_t$ $1 \leq K_t^x \leq K$. Only the required number of pixels is used at every moment, that is why this method achieves a better processing time than[6] and gives better results. the function of Weight update is.

$$w_{i,t} = w_{i,t-1} + \alpha(M_{i,t} - w_{i,t-1}) - \alpha cT \quad (6)$$

where $cT$ is a constant

### 2.3. Fuzzy Models

Sugeno Integral method is used to distinguish between the input image and the background model the measurement of the color and texture[23], Choquet Integral is adapted in [24] [25]. It is based on color, edge and texture features. This method shows better results compared to the existing techniques. Fuzzy Running Average is used to extract the background model in [26] . Where, 2-fuzzy method in the multimedia background is adapted in [27]–[29].

### 2.4. Neural And Neuro-Fuzzy Methods

The Neural Network method is used to determine whether the pixel belongs to the foreground or to the background. In fact, a background segmentation approach using a multi-layer neural network containing 124 neurons is adapted in [30] . This method is based on probabilistic neural networks (PNN). A self-organizing network method for background subtraction (SOBS) is used in [31]. Whereas, a combined SOBS with a fuzzy function is adapted in the learning stage of the background in [32] [12].

## 3 . EXPERIMENTS

### 3.1. Detection Performance And Evaluation

In the present work, all algorithms in the subtraction library were evaluated in the background by segmentation performance, execution time, and And memory consumption and processor . Firstly, assessing segmentation performance is to demonstrate the ability of algorithms to extract moving objects, by choosing the most commonly used set of measurements in the literature. Recall. Precision .F_mesasure. Accuracy. TPR. FPR. In fact, these measurements are calculated by TP,TN,FP,FN. We get by the following equations where TP (True Positives) is the set of pixels in the front, which are captured as belonging to the front after the simulations. FP (False Positives ) is the set of background pixels that are captured as belonging to the foreground. TN( True Negatives): is the set of background pixels that are captured as belonging to the background. FN(False Negatives) is the set of foreground pixels that are captured as belonging to the background.

$Recall = \frac{TP}{TP+FN}$, $Precision = \frac{TP}{TP+FP}$, $\mathrm{F-measure} = \frac{2*Recall*Precision}{Recall+Precision}$, $Accuracy(AC) = \frac{TP}{TP+FN+FP}$

$\mathrm{FalsePositiveRate}(FPR) = \frac{FP}{FP+TN}$, $\mathrm{FalseNegativeRate}(FNR) = \frac{FN}{TP+FN}$, $\mathrm{True\ Positive\ Rate}(TPR) = \frac{TP}{TP+FN}$

The results of the measurements are shown in Table 1. It is important to note that the simulated algorithm provides a visual note of the extracted images which are represented by the Foreground masks. On the other hand, metrics help algorithms to evaluate and make a comparison between them . The Foreground masks are shown in Figure 1.

Each algorithm has a certain execution time, some of which are executed at a short speed and some that take a long time. With a certain amount of memory consumption, where the majority of algorithms have a constant value in memory consumption over time, and some have an increasing value over time, such as (T2FMRF_UM, T2FMRF_UV). the core processor on which the analyses were performed is i5 , 2.40 GHZ, RAM(8 GO).The library is compiled in C ++ programming language, based on opencv. Moreover, visual studio 2017 is used to simulate these algorithms. Then, comparisons are performed through MATLAB2017.

### 3.2. The Dataset Used

For a good, comprehensive, real-time evaluation. a BMC (background modelling challenge) dataset was selected using the synthetic video type with the presence of noise. The video number selected from the dataset is 211.

Table 1. The rating scales used for BMC data analysis, synthetic video 211, and use case in cloudy with noise.

| Methods | RC | PC | TPR | FPR | FNR | FM | AC |
|---|---|---|---|---|---|---|---|
| FrameDifference; | 0.6845 | 0.4092 | 0.6845 | 0.0257 | 0.3155 | 0.5122 | 0.3443 |
| StaticFrameDifference | 0.9264 | 0.1580 | 0.9264 | 0.1285 | 0.0736 | 0.2699 | 0.1560 |
| WeightedMovingMean | 0.6043 | 0.8619 | 0.6043 | 0.0025 | 0.3957 | 0.7105 | 0.5509 |
| WeightedMovingVariance | 0.6937 | 0.8293 | 0.6937 | 0.0037 | 0.3063 | 0.7555 | 0.6071 |
| MixtureOfGaussianV1 | 0.6909 | 0.9385 | 0.6909 | 0.0012 | 0.3091 | 0.7958 | 0.6609 |
| MixtureOfGaussianV2 | 0.6668 | 0.5596 | 0.6668 | 0.0137 | 0.3332 | 0.6085 | 0.4373 |
| AdaptiveBackgroundLearning | 0.9278 | 0.5376 | 0.9278 | 0.0208 | 0.0722 | 0.6808 | 0.5160 |
| AdaptiveSelectiveBackgroundLearning | 0.9047 | 0.7330 | 0.9047 | 0.0086 | 0.0953 | 0.8098 | 0.6804 |
| GMG | 0.9553 | 0.6100 | 0.9553 | 0.0159 | 0.0447 | 0.7446 | 0.5931 |
| KNN | 0.9201 | 0.6888 | 0.9201 | 0.0108 | 0.0799 | 0.7878 | 0.6499 |
| DPAdaptiveMedian | 0.4136 | 0.9379 | 0.4136 | 7.1289e-04 | 0.5864 | 0.5741 | 0.4026 |
| DPGrimsonGMM | 0.9193 | 0.6732 | 0.9193 | 0.0116 | 0.0807 | 0.7772 | 0.6356 |
| DPZivkovicAGMM | 0.8864 | 0.8327 | 0.8864 | 0.0046 | 0.1136 | 0.8587 | 0.7524 |
| DPMean | 0.4800 | 0.8611 | 0.4800 | 0.0020 | 0.5200 | 0.6164 | 0.4455 |
| DPWrenGA | 0.8549 | 0.8714 | 0.8549 | 0.0033 | 0.1451 | 0.8631 | 0.7591 |
| DPPratiMediod | 0.8525 | 0.9419 | 0.8525 | 0.0014 | 0.1475 | 0.8950 | 0.8099 |
| DPEigenbackground | 0.9159 | 0.7320 | 0.9159 | 0.0087 | 0.0841 | 0.8137 | 0.6859 |
| DPTexture | 0.3768 | 0.9395 | 0.3768 | 6.3151e-04 | 0.6232 | 0.5379 | 0.3679 |
| T2FGMM_UM | 0.0392 | 0.9812 | 0.0392 | 1.9531e-05 | 0.9608 | 0.0753 | 0.0391 |
| T2FGMM_UV | 0.6205 | 0.0430 | 0.6205 | 0.3597 | 0.3795 | 0.0803 | 0.0419 |
| T2FMRF_UM | 0 | NaN | 0 | 0 | 1 | NaN | 0 |
| T2FMRF_UV | 0.3872 | 0.5659 | 0.3872 | 0.0077 | 0.6128 | 0.4598 | 0.2985 |
| FuzzySugenoIntegral | 0.8364 | 0.8690 | 0.8364 | 0.0033 | 0.1636 | 0.8524 | 0.7427 |
| FuzzyChoquetIntegral | 0.8695 | 0.8451 | 0.8695 | 0.0041 | 0.1305 | 0.8571 | 0.7500 |
| MultiLayer | 0.9222 | 0.9041 | 0.9222 | 0.0025 | 0.0778 | 0.9130 | 0.8400 |
| PixelBasedAdaptiveSegmenter | 0.9423 | 0.6804 | 0.9423 | 0.0115 | 0.0577 | 0.7902 | 0.6532 |
| LBSimpleGaussian | 0.9424 | 0.0624 | 0.9424 | 0.3687 | 0.0576 | 0.1170 | 0.0621 |
| LBFuzzyGaussian | 0.9378 | 0.2239 | 0.9378 | 0.0846 | 0.0622 | 0.3615 | 0.2206 |
| LBMixtureOfGaussians | 0.9189 | 0.6897 | 0.9189 | 0.0108 | 0.0811 | 0.7880 | 0.6502 |
| LBAdaptiveSOM | 0.9198 | 0.6943 | 0.9198 | 0.0105 | 0.0802 | 0.7913 | 0.6547 |
| LBFuzzyAdaptiveSOM | 0.9119 | 0.7404 | 0.9119 | 0.0083 | 0.0881 | 0.8172 | 0.6910 |
| LBP_MRF | 0.9398 | 0.7640 | 0.9398 | 0.0076 | 0.0602 | 0.8429 | 0.7284 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| VuMeter | 0.7281 | 0.7764 | 0.7281 | 0.0055 | 0.2719 | 0.7515 | 0.6019 |
| KDE | 0.9401 | 0.1613 | 0.9401 | 0.1272 | 0.0599 | 0.2753 | 0.1596 |
| IndependentMultimodal | 0.9140 | 0.6821 | 0.9140 | 0.0111 | 0.0860 | 0.7812 | 0.6410 |
| MultiCue | 0.9769 | 0.7029 | 0.9769 | 0.0107 | 0.0231 | 0.8175 | 0.6913 |
| SigmaDelta | 0.9226 | 0.6455 | 0.9226 | 0.0132 | 0.0774 | 0.7596 | 0.6124 |
| SuBSENSE | 0.9291 | 0.8976 | 0.9291 | 0.0028 | 0.0709 | 0.9131 | 0.8400 |
| LOBSTER | 0.9079 | 0.9053 | 0.9079 | 0.0025 | 0.0921 | 0.9066 | 0.8292 |
| PAWCS | 0.9232 | 0.9008 | 0.9232 | 0.0026 | 0.0768 | 0.9118 | 0.8380 |
| TwoPoints | 0.8539 | 0.3270 | 0.8539 | 0.0457 | 0.1461 | 0.4729 | 0.3097 |
| ViBe | 0.8475 | 0.9273 | 0.8475 | 0.0017 | 0.1525 | 0.8856 | 0.7947 |
| CodeBook; | 0.9299 | 0.1123 | 0.9299 | 0.1912 | 0.0701 | 0.2005 | 0.1114 |

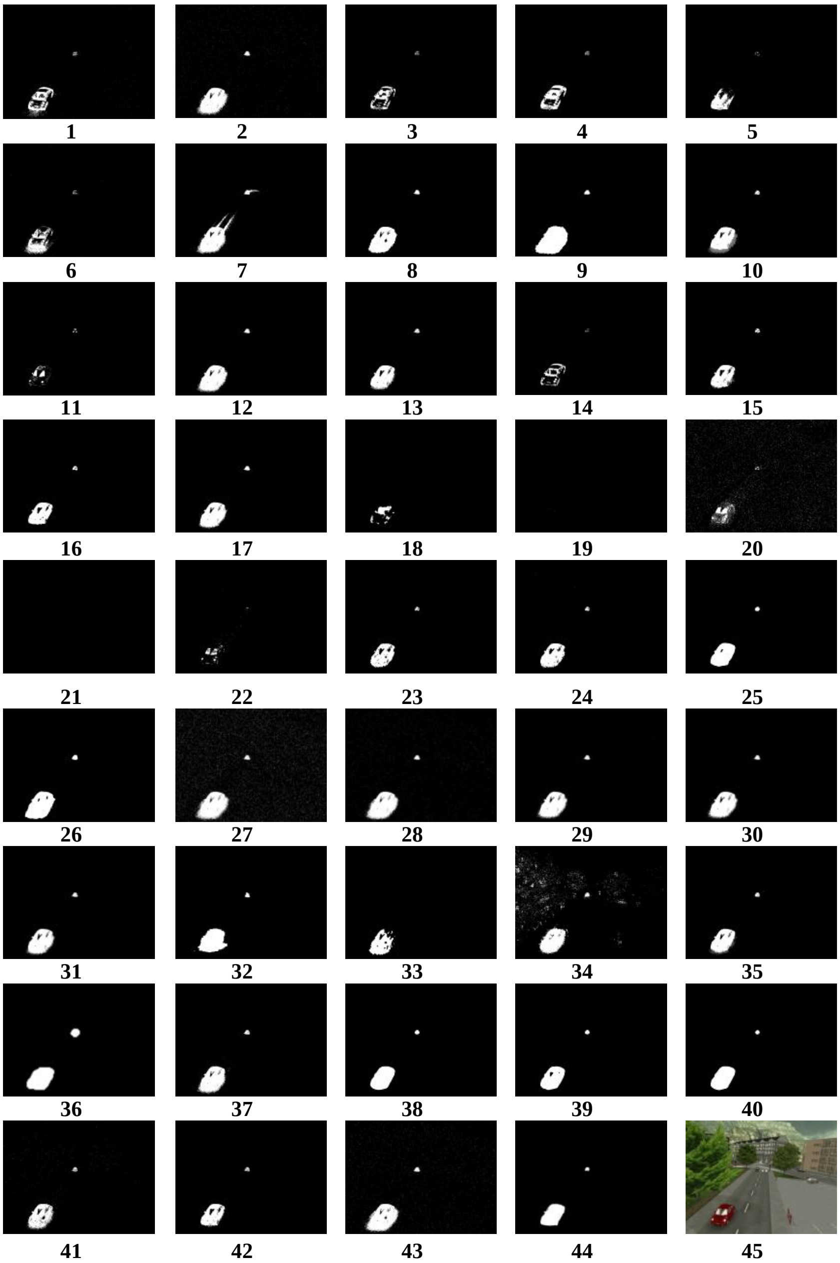


Figure 1. Foreground masks obtained from Street Video 211 from BMC dataset.

In this case , cloudy with noise are the variables parameters. They are measured and compared as in the Figure.1. where each sub-figure presents a part of tests as follows:

1= FrameDifference; 2= StaticFrameDifference; 3=WeightedMoving Mean; 4= WeightedMovingVariance; 5=MixtureOfGaussianV1;6=MixtureOfGaussianV2;7=AdaptiveBackgroundLearning; 8=AdaptiveSelectiveBackgroundLearning; 9=GMG; 10=KNN; 11= DPAdaptiveMedian; 12=DPGrimsonGMM. 13=DPZivkovicAGMM. 14=DPMean. 15=DPWrenGA. 16=DPPratiMediod. 17=DPEigenbackground. 18=DPTexture. 19=T2FGMM_UM. 20=T2FGMM_UV. 21=T2FMRF_UM. 22=T2FMRF_UV. 23=FuzzySugenoIntegral. 24=FuzzyChoquetIntegral. 25=MultiLayer. 26=PixelBasedAdaptiveSegmenter.27=LBSimpleGaussian; 28=LBFuzzyGaussian. 29=LBMixtureOfGaussians. 30=LBAdaptiveSOM. 31=LBFuzzyAdaptiveSOM. 32=LBP_MRF. 33=VuMeter. 34=KDE. 35=IndependentMultimodal. 36=MultiCue. 37=SigmaDelta. 38=SuBSENSE. 39=LOBSTER. 40=PAWCS. 41=TwoPoints. 42=ViBe. 43=CodeBook. 44= ground truth; 45= Original frame

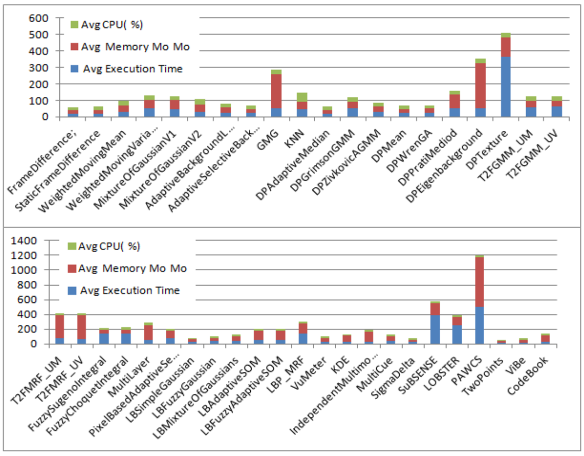


Figure 2. Average value of time, memory consumption, and processor

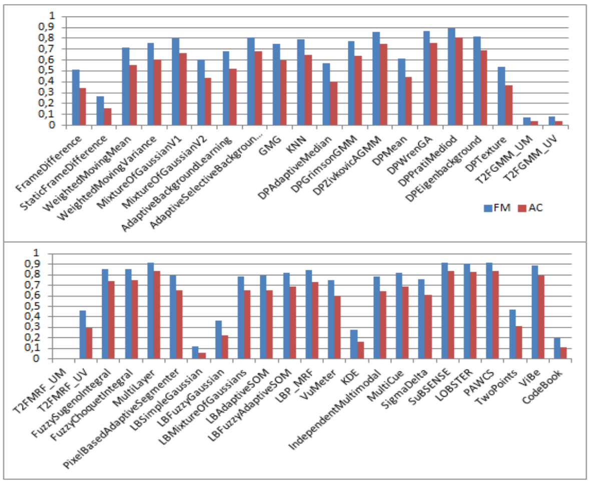


Figure 4. The columns represent the F-measure and Accuracy value, taken from the Street Video211, for the BMC data set

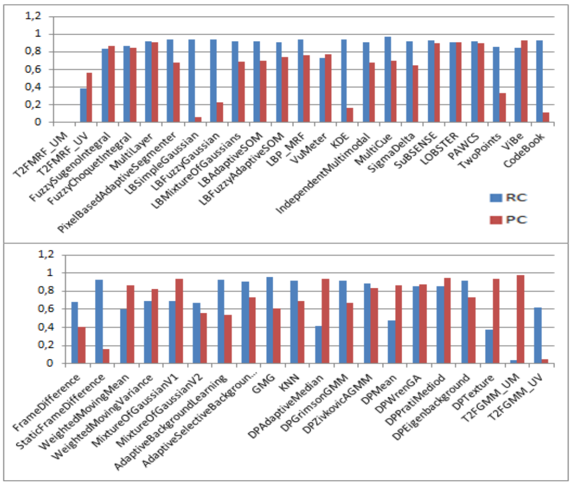


Figure 3. The columns represent the Precision and Recall value, taken from the Street Video211, for the bmc data set.

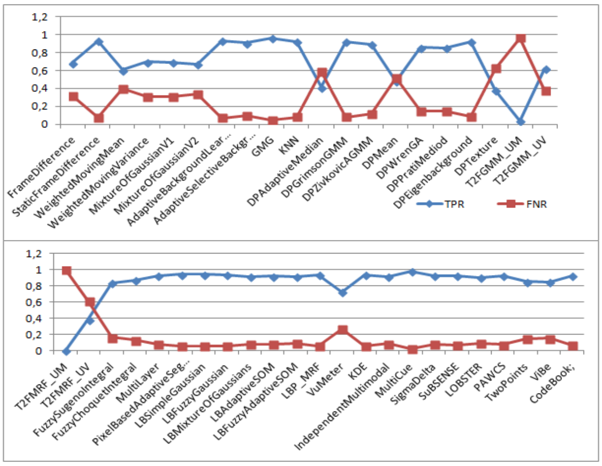


Figure 5. the Curves represent False Negative Rate, and True positive rate of the synthetic street video from the BMCdata set

## 4. RESULTS AND DISCUSSION

The evaluation results for this video are done in Table 1. It can be noticed that this type takes very high scaling scores F-measure. So , when the scale value is aqual to 1 better result may be obtained. In the Table 1 some of the algorithms are considered using scale value of 0.9 , all such as: SuBSENSE, LOBSTER, PAWCS . Although these methods are difficult to implement. Moreover , they take a long time to response (about 6 to 8 minutes) leading to a reduction of the effectiveness in extracting objects in motion at the ideal time. Also, some other methods implemented and done as in Figure.2. took lowest values in the evaluation measures and ease to implemnt using this kind of video. As a result, they are effictive and give acceptable results, where we can listed them as follows: CodeBook, LBSimpleGaussian, StaticFrameDifference. In other hand, some of the given methods are capable to achieve good and consistent results between the values of the measurements of the Table 1, with the execution time and memory of the Figure 2. These methods may be done as : AdaptiveSelectiveBackgroundLearning, MultiCue, DPWrenGA, DPZivkovicAGMM. The foreground masks extracted from the BMC video (Frame 597) of the Figure 1, are well interpreted in Table 1, and can be depicted as shown in Figures 3, 4, and 5.

## 5. CONCLUSION AND FUTURE WORK

In this work, a comprehensive evaluation of the performance of the BGSLibrary's algorithms was performed. In fact, hash performance, execution time, memory and processor consumption are considered to fullfil this goal, where, a noise-containing synthetic video from the BMC dataset was used. The experimental results showed that the synthetic video from the bmc dataset had good results ,where, a higher true positive rate and lower false negative rate, and an increase in the F scale value to 0.91 are obtained. The MultiCue, DPWrenGA, PixelBasedAdaptiveSegmenter , DPZivkovicAGMM, AdaptiveSelectiveBackgroundLearning algorithms are successfully deal with shadows and noise . Indeed, they achieve good results consistent with high benchmark values and low execution time.

Future work will focus on validating these methods under different challenges such as snow, fog, wind, crowd, shadows, nois. The design and the update of a background model is one of the biggest challenge in artificial vision. So, the developement of a system capable of detecting a background model correctly and completely with high performance and ease algorithm, then, adapt it to the external dynamic changes and acquired shadows through real time implementation will be considered.

References

[1] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati, “Detecting moving objects, ghosts, and shadows in video streams,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 10, pp. 1337–1342, Oct. 2003, doi: 10.1109/TPAMI.2003.1233909.

[2] Z. Tang, Z. Miao, and Y. Wan, “Background Subtraction Using Running Gaussian Average and Frame Difference,” in *Entertainment Computing – ICEC 2007*, Berlin, Heidelberg, 2007, pp. 411–414, doi: 10.1007/978-3-540-74873-1_50.

[3] N. J. B. McFarlane and C. P. Schofield, “Segmentation and tracking of piglets in images,” *Mach. Vis. Appl.*, vol. 8, no. 3, pp. 187–193, May 1995, doi: 10.1007/BF01215814.

[4] S. Calderara, R. Melli, A. Prati, and R. Cucchiara, “Reliable background suppression for complex scenes,” in *Proceedings of the 4th ACM international workshop on Video surveillance and sensor networks*, Santa Barbara, California, USA, Oct. 2006, pp. 211–214, doi: 10.1145/1178782.1178814.

[5] Y. Benezeth, P. M. Jodoin, B. Emile, H. Laurent, and C. Rosenberger, “Review and Evaluation of Commonly-Implemented Background Subtraction Algorithms,” in *2008 19th International Conference on Pattern Recognition*, Dec. 2008, pp. 1–4, doi: 10.1109/ICPR.2008.4760998.

[6] C. Stauffer and W. Grimson, “Adaptive Background Mixture Models for Real-Time Tracking,” in *Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149)(CVPR)*, 1999, vol. 02, p. 2246, doi: 10.1109/CVPR.1999.784637.

[7] A. Sobral, *A background subtraction library. Contribute to andrewssobral/bgslibrary development by creating an account on GitHub*. 2019.

[8] A. Sobral, “BGSLibrary: An OpenCV C++ Background Subtraction Library,” p. 6.

[9] Y. Xu, J. Dong, B. Zhang, and D. Xu, “Background modeling methods in video analysis: A review and comparative evaluation,” *CAAI Trans. Intell. Technol.*, vol. 1, no. 1, pp. 43–60, Jan. 2016, doi: 10.1016/j.trit.2016.03.005.

[10] S. Brutzer, B. Höferlin, and G. Heidemann, “Evaluation of background subtraction techniques for video surveillance,” in *CVPR 2011*, Jun. 2011, pp. 1937–1944, doi: 10.1109/CVPR.2011.5995508.

[11] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers, “Wallflower: principles and practice of background maintenance,” in *Proceedings of the Seventh IEEE International Conference on Computer Vision*, Sep. 1999, vol. 1, pp. 255–261 vol.1, doi: 10.1109/ICCV.1999.791228.

[12] A. Sobral and A. Vacavant, “A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos,” *Comput. Vis. Image Underst.*, vol. 122, pp. 4–21, May 2014, doi: 10.1016/j.cviu.2013.12.005.

[13] F. J. López-Rubio, E. López-Rubio, M. A. Molina-Cabello, R. M. Luque-Baena, E. J. Palomo, and E. Domínguez, “The effect of noise on foreground detection algorithms,” *Artif. Intell. Rev.*, vol. 49, no. 3, pp. 407–438, Mar. 2018, doi: 10.1007/s10462-016-9525-3.

[14] A. Vacavant, T. Chateau, A. Wilhelm, and L. Lequièvre, "A Benchmark Dataset for Outdoor Foreground/Background Extraction," in *Computer Vision - ACCV 2012 Workshops*, Berlin, Heidelberg, 2013, pp. 291–300, doi: 10.1007/978-3-642-37410-4_25.
[15] T. Bouwmans, "Recent Advanced Statistical Background Modeling for Foreground Detection - A Systematic Survey," *Recent Pat. Comput. Sci.*, vol. 4, no. 3, pp. 147–176, Sep. 2011.
[16] L. Shang, F. You, C. Han, X. Wang, and S. Zhao, "Optimization of Three - Frame Difference Method and Improvement of Pedestrian Detection Code Book," *J. Phys. Conf. Ser.*, vol. 1302, p. 022014, Aug. 2019, doi: 10.1088/1742-6596/1302/2/022014.
[17] A. H. S. Lai and N. H. C. Yung, "A fast and accurate scoreboard algorithm for estimating stationary backgrounds in an image sequence," in *1998 IEEE International Symposium on Circuits and Systems (ISCAS)*, May 1998, vol. 4, pp. 241–244 vol.4, doi: 10.1109/ISCAS.1998.698804.
[18] C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland, "Pfinder: real-time tracking of the human body," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 780–785, Jul. 1997, doi: 10.1109/34.598236.
[19] A. Elgammal, D. Harwood, and L. Davis, "Non-parametric Model for Background Subtraction," in *Computer Vision — ECCV 2000*, Berlin, Heidelberg, 2000, pp. 751–767, doi: 10.1007/3-540-45053-X_48.
[20] Z. Zivkovic, "Improved adaptive Gaussian mixture model for background subtraction," in *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, Aug. 2004, vol. 2, pp. 28-31 Vol.2, doi: 10.1109/ICPR.2004.1333992.
[21] Z. Zivkovic and F. van der Heijden, "Efficient adaptive density estimation per image pixel for the task of background subtraction," *Pattern Recognit. Lett.*, vol. 27, no. 7, pp. 773–780, May 2006, doi: 10.1016/j.patrec.2005.11.005.
[22] E. López-Rubio and R. M. Luque-Baena, "Stochastic approximation for background modelling," *Comput. Vis. Image Underst.*, vol. 115, no. 6, pp. 735–749, Jun. 2011, doi: 10.1016/j.cviu.2011.01.007.
[23] H. zhang and D. xu, "Fusing Color and Texture Features for Background Model," in *Fuzzy Systems and Knowledge Discovery*, Berlin, Heidelberg, 2006, pp. 887–893, doi: 10.1007/11881599_110.
[24] G. Choquet, "Theory of capacities," *Ann. Inst. Fourier*, vol. 5, pp. 131–295, 1954, doi: 10.5802/aif.53.
[25] M. M. Azab, H. A. Shedeed, and A. S. Hussein, "A new technique for background modeling and subtraction for motion detection in real-time videos," in *2010 IEEE International Conference on Image Processing*, Sep. 2010, pp. 3453–3456, doi: 10.1109/ICIP.2010.5653748.
[26] M. H. Sigari, N. Mozayani, and H. R. Pourreza, *Fuzzy Running Average and Fuzzy Background Subtraction: Concepts and Application*. .
[27] T. Bouwmans and F. El Baf, "Modeling of Dynamic Backgrounds by Type-2 Fuzzy Gaussians Mixture Models," *MASAUM J. Basic Appl. Sci.*, vol. 1, no. 2, pp. 265–276, Feb. 2010.
[28] F. El Baf, T. Bouwmans, and B. Vachon, "Type-2 Fuzzy Mixture of Gaussians Model: Application to Background Modeling," in *Advances in Visual Computing*, Berlin, Heidelberg, 2008, pp. 772–781, doi: 10.1007/978-3-540-89639-5_74.
[29] F. E. Baf, T. Bouwmans, and B. Vachon, "Fuzzy statistical modeling of dynamic backgrounds for moving object detection in infrared videos," in *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, Jun. 2009, pp. 60–65, doi: 10.1109/CVPRW.2009.5204109.
[30] D. Culibrk, O. Marques, D. Socek, H. Kalva, and B. Furht, "Neural Network Approach to Background Modeling for Video Object Segmentation," *IEEE Trans. Neural Netw.*, vol. 18, no. 6, pp. 1614–1627, Nov. 2007, doi: 10.1109/TNN.2007.896861.
[31] L. Maddalena and A. Petrosino, "A Self-Organizing Approach to Background Subtraction for Visual Surveillance Applications," *IEEE Trans. Image Process.*, vol. 17, no. 7, pp. 1168–1177, Jul. 2008, doi: 10.1109/TIP.2008.924285.
[32] L. Maddalena and A. Petrosino, "A fuzzy spatial coherence-based approach to background/foreground separation for moving object detection," *Neural Comput. Appl.*, vol. 19, no. 2, pp. 179–186, Mar. 2010, doi: 10.1007/s00521-009-0285-8.
[33] M. Hofmann, P. Tiefenbacher, and G. Rigoll, "Background segmentation with feedback: The Pixel-Based Adaptive Segmenter," in *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, Jun. 2012, pp. 38–43, doi: 10.1109/CVPRW.2012.6238925.