ISSN: 0067-2904

Construction of a Robust Background Model for Moving Object Detection in Video Sequence

Abdulamir A. Karim
Department of Computer Science, University of Technology, Baghdad, Iraq

Abstract

Background Subtraction (BGS) is one of the main techniques used for moving object detection which further utilized in video analysis, especially in video surveillance systems. Practically, acquiring a robust background (reference) image is a real challenge due to the dynamic change in the scene. Hence, a key point to BGS is background modeling, in which a model is built and repeatedly used to reconstruct the background image.

From N frames the proposed method store N pixels at location(x,y) in a buffer, then it classify pixel intensity values at that buffer using a proposed online clustering model based on the idea of relative run length, the cluster center with the highest frequency will be adopted as the background pixel value at location (x,y). For background updating, two approaches has been proposed to repeatedly update background image.

The experiment results show that the average Precision, Recall and F-measure for the proposed method was 0.89, 0.96 and 0.93 respectively. While the average time in seconds required to construct background pixel from a buffer of size 50, 100 and 150 pixel was 0.0022813 sec , 0.0510166 sec and 0.12240419 sec respectively.

Keywords: Background Modeling, Background Subtraction, Moving Object Detection.

بناء نموذج خلفية فعال لكشف هدف متحرك في سلسلة فيديو

عبدالأمير عبدالله كريم
قسم علوم الحاسوب، الجامعة التكنلوجية، بغداد، العراق

الخلاصة

تمثل طرح الخلفية أحد التقنيات الرئيسية المستخدمة لأكتشاف الأجسام المتحركة والتي تستخدم بشكل كبير في تحليل الفيديو ، خاصة في أنظمة المراقبة بالفيديو.عمليًا ، استحصال صورة الخلفية (الصورة المرجعية) المتين يعد تحديًا حقيقيًا بسبب التغيير الديناميكي في المشهد. وبالتالي ، فإن النقطة الأساسية في عملية طرح الخلفية هي نمذجة الخلفية ، حيث يتم بناء نموذج واستخدامه بشكل متكرر لإعادة بناء صورة الخلفية.

الطريقة المقترحة تخزن N من البكسل الموجوده في الموقع (x، y) ل N من الأطارات في مخزن مؤقت ، ثم تقوم بتصنيف قيم كثافة البكسل في ذلك المخزن المؤقت باستخدام نموذج عنقدة مقترح قائم على فكرة طول التشغيل النسبي ، سيتم اعتماد مركز العنقود ذا التكرار الأعلى كقيمة بكسل الخلفية في الموقع (x، y). لتحديث الخلفية ، تم اقتراح نهجين لتحديث صورة الخلفية بشكل متكرر.

Email: ameer.aldelphi@yahoo.com

أظهرت النتائج التجريبية أن متوسط الدقة ، الاستعادة و مقياس-F للطريقة المقترحة كان 0.89 ، 0.96 و 0.93 على التوالي. في حين أن متوسط الوقت بالثواني المطلوب لإنشاء بكسل خلفية من مخزن مؤقت بحجم 50،100و 150 بكسل كان 0.0022813 ثانية ، 0.0510166 ثانية و 0.12240419 ثانية على التوالي.

1. Introduction

In video processing systems object detection and tracking are considered as the most important processes, and in order to detect and segment an object, BGS can be employed, which is considered as a simple but very efficient method of object detection [1, 2]. BGS involve comparing the current frame with an already built background image, depending on a predefined threshold value, a decision is make to decide whether the detected object is a foreground or background, hence , a key for BGS is background modeling (reconstruction) [2].

Two methods can be used to obtain the reference image , the first conventional one, is by acquiring a background image directly from the scene, this method suffer from a number of effects such as illumination changes, objects move inside the scene , in other words it is not fit with the changing environment in a complex background [1]. Hence, a robust background reconstruction algorithm is required to model the background accurately especially when there is a frequent objects moves inside or outside the background, and that what lead to the use of background modeling (reconstruction) methods in which a reference image is created from the previous frames.

Background modeling is considered a research area till now, and it still face a number of challenges , from which : it is rather slow which means it cannot meet the real time requirements, Also it fail to construct an excellent background especially in busy environment [1]. The main task of background construction is to select the best candidate from a consecutive pixel intensity values in order to represent the background of each pixel coordinate within a predetermined period of time [1][2] .

Reconstructing a background image from a video sequence containing a complex scene is a critical task , which can be done in a number of methods [2] . The simplest method to background modeling is Time Average Background Image (TABI) in which a background image is obtained by averaging a number of consecutive frames, this method is efficient with simple situations with no moving objects, but it is not robust to complex scene which contain a number of moving objects, it combine foreground objects with background image [3, 4]. Then background reconstruction methods has been extended by modeling the pixel intensity as a Mixture of Gaussian (MoG) , this method can cope with slow changes in illumination, and long term changes in the scene, but on the other hand , it is computationally expensive , and it's parameters required carful estimation [3,2, 5]. Later , median filter has been employed to background modeling , in which the background pixel is proposed to be the median at each pixel coordinate of all the frames in the buffer, median method can applied to less complex scenes [3, 5]. The last and The most commonly used background modeling method is Pixel Intensity Classification (PIC) method , in this method for each pixel location (coordinate) in a consecutive frames , the pixel intensity values between the inter-frame is taken into consideration , then those pixels classified according to their intensity values, finally, the pixel intensity value with the higher frequency is chosen to represent the background pixel [2, 3].

In this work , a new background modeling method has been proposed, which considered as an improved PIC method, the proposed method can construct a robust and accurate background image, it also overcome two of the difficulties which face PIC method.

2. Moving Object Detection

Moving object detection is the process of extracting the regions of variation (foreground regions) out of background image in a video sequence. The pixels of moving objects (foreground) is taken into consideration due to the subsequent processing such as object tracking, object classification and recognition [6]. There have been many methods for moving object detection, the most common are presented in the following subsections:

2.1. Optical flow: Is calculates the optical flow field of the image, then do bunch process to the distribution of the optical flow of the image (i.e. video frame). The essential idea of optical flow is that feature of the frame are tracked over a period of time to compute the relative velocity of the moving objects in the scene. It's ultimate objective is to determine the motion between consecutive frames which are captured at time **t** and **t +α t** at each position [7].

Optical flow is a complex algorithm, it required a huge computation and it is poor for real time, also it is very sensitive to noise, besides, a special hardware is needed for this method, thus, it is practically very poor [4,6, 7] .

2.2. Frame difference: in this approach moving objects are detected by calculating the pixel-by-pixel difference of two consecutive frames in a video sequence. The below equation is used to obtain the above difference:

$$| f_t(i,j) - f_{t-1}(i,j) | > T \quad (1)$$

Where f_t (i,j) , f_{t-1}(i,j) are the pixel values at location (i,j) for two consecutive frames at time **t , t-1** respectively, **T** is the threshold value [7] .This approach has been developed to accommodate three-frame differencing so that to improve the results that has been obtained by two-frames difference. Frame difference method is a simple algorithm , less complex in it's computation with high stability , and it also adaptive to dynamic change in video frames , besides it can detect moving objects in real time even thought there were a dynamic changes in the environment. On the other hand the object obtained by this method cannot be an integrated i.e. complete region of object cannot be completely extracted, it can extract the boundary of the object only). In addition, it is extra sensitive to threshold value, and it need a supportive algorithm that can be used to detect stationary objects [4,6, 7].

2.3 Background Subtraction: this method subtract the current frame from reference frame (background image) which contains non moving object. The background image can to be captured directly from the scene, or else, it has to be modeled. The subtraction is done in pixel-by-pixel manner, so that each pixel of reference frame is subtracted from its corresponding current frame pixels, and then thresholded [4,6, 7]. The below equation is used to obtain the above difference:

$$| f_t(i,j) - R(i,j) | > T \quad (2)$$

Where f_t (i,j) is the current frame (at time t) pixel values at location (i,j), R(i,j) is the pixel values of the reference frame at location (i,j) , **T** is the threshold value. When the difference is greater than T, that's mean this pixel is belong to an object (foreground), otherwise it's belong to background.

This method almost can obtain an integrated (complete) object, besides, it is the easiest and most efficient method among the two above ones. Although this method is accurate and fast, it is oversensitive to variations in illumination, which largely effecting it's performance. Besides, it is not robust to environments with many moving object, especially if those objects moves slowly [4,6, 7].

3. Background Modeling

A key task for BGS is to compare between an input frame and a background image (reference image), this comparison is always done using subtraction operation. Results of subtraction leads to the process of object detection which determine which area of the inputted frame can be classified as foreground (with respect to a predetermined threshold value), the ultimate result of BGS operation is a binary image [8] ,hence, in order to perform a good BGS , a background image has to be obtained. Two main methods can be used to obtain background images, the first, is used in idealistic environment, when there were no objects move inside or outside the scene, and this can be happen only in indoor environment, in this case a background image can be acquired directly from the scene, but this situation cannot practically happen, because background can always be changed. In outdoor environment like highways or busy streets, because of the changes in light conditions along different times of the day, also changes in weather conditions like rain , fog, strong wind, all that can modify the reference image [9],hence, there was a tendency to use another method, in which the background image has to be modeled, and this approach is the one which has been followed in this work. From the numerous available methods which can be used to model the background, PIC method is adopted and a new algorithms which considered as an improved of PIC method are proposed.

The main idea behind all PIC methods is that, it assume the pixel intensities of the background are frequently appears in the consecutive frames with the higher probability, hence, when a classification to the pixel intensity is performed to those pixels between inter-frame, the intensity value with the higher frequency can be selected to represent the background pixels [2,3, 10] **.**

PIC methods suffer from several difficulties , which must be overcome . Firstly, a predetermined threshold must be fixed in order to divide the pixel intensity intervals, which strongly influences the real time requirement of any PIC algorithms. Secondly, another threshold has to be specified in order to merge the approximating intensity intervals. Thirdly, in spite of it is a time consuming method, it is very difficult to perceive which of the approximating intensity intervals should be merge, and which of them should not [2]. The proposed method overcome the second and third difficulties, because whenever an interval is determined, there is no need to merge it with another interval , and that will positively effect on the processing time overhead.

4. Performance Evaluation

Each of the background modeling methods have its own characteristics (i.e. strength and weakness). Evaluation measures helps to identify those characteristics in order to emphasis it's strength point and overcome its weakness points [8]. In the below paragraphs, we first illustrate the main challenges of BGS, then we identify it's requirements, finally a number of evaluation measures will be introduced.

4.1 challenges of Background Subtraction

There have been a number of challenges that any robust background modeling must cope with. Among those challenges, the below situations have to be handled [4,5,6,8, 11]:

- Dynamic (non stationary) background.
- Illumination changes: in outdoor environment the light intensity already changes during day.
- Sudden change of light: like switching the lights on/off in indoor environment.
- Motion changes: object being introduced or removed from scene.
- Slow moving object: such as cars moves in a very busy street.
- High frequently moving objects.
- Season changes.
- Video noise: video signals are generally contaminated by noise. A good BGS model has to face such degraded signals affected by various types of noise like compression artifacts or sensor noise.
- Camera jitter: camera shake
- Shadows: the overlapping between shadows and foreground is strongly hinder the separation and classification operation, i.e. it cause shadow pixels to classified as foreground.
- Camouflage: intentionally or not, some objects may slightly differ from the background, making accurate classification so difficult, such as a person wearing dark clothes standing near a grey car.

4.2. Requirements of Background Subtraction

Beside the above challenges, there have been a number of requirements, any background modeling method should fulfilled [7,8, 12] :

- In spite of the difficulties of background modeling due to the unpredictability and complexity of scene, BGS methods have to be computationally economical , hence, it can accommodate the real time requirement (i.e. react quickly to changes in background).
- It must adapt the low memory space requirement.
- It must control the moving objects that immerse into the scene at advance time, i.e. those objects which remain in the scene for a while, does those objects considered as foreground or background.
- BGS methods should avoid detecting non-static tiny background objects like rain, moving trees leafs, in other words, some part of the scene can contains moving object, but those objects should be regarded as background.

4.3 Performance Measures

In general, human beings are the better evaluator of every vision system, however, it is not constantly possible for human being to evaluate those vision systems in a quantitative manner. Hence, in order to evaluate any proposed method, it is necessary to provide a quantitative evaluation measures [5]. In this work, the performance of the constructed background has been judge depending on the results of BGS, in which we consider a binary classification for the detected object and the background (i.e. segmentation by thresholding) [8]. The accuracy of this classification is implemented in pixel level using three quantitative measure which are Recall , Precision and F-measure. Note that the result of those measures should be high for better detection of moving foreground objects.

Recall measure can be obtained by: [8]:

$$Recall = \frac{\text{no. of correctly classified foreground pixels}}{\text{no. of foreground pixels in ground truth}} \tag{1}$$

which can more analytically described by: [5]

$$Recall = \frac{\text{True posative}}{\text{True posative+ Fasl negative}} \tag{2}$$

While Precision measure can be obtained by: [8]

$$Precision = \frac{\text{no. of correctly classified foreground pixels}}{\text{no. of pixels classified as foreground}} \tag{3}$$

which can more analytically described by: [5] $$Precision = \frac{\text{True posative}}{\text{True posative+ Fasl possative}} \tag{4}$$

And F-measure can be obtained by: [5, 8]

$$F - measure = \frac{2 * Precision * Recall}{Precision + Recall} \tag{5}$$

Here *True positive* is the number of pixels correctly labeled as object (foreground) class, *False positive* is the number of pixels that are incorrectly labeled as object class, while *False negative* is the number of pixels which are not labeled as object class but should have been [5].

In this work, in order to establish a good performance measures, the ground truth pixels are labeled manually.

5. Proposed method

In this work an improved PIC algorithm has been proposed in which a background model is fast enough so that it can react immediately (*in real time*) to the changing background, it can also overcome two of the PIC drawbacks (which has been mentioned in section 3).

The proposed method assume that the video under consideration is a grey scale video , since adding color in background modeling leads to increase the complexity of the background estimation process, whilst it does not add any positive effect to that process.

In Algorithms 1 and 2 below, N frames has been examined from the input video sequentially and N pixels at location(x,y) stored in a buffer, then the pixels in the buffer sorted in ascending manner. The pixels in the buffer are read sequentially, if any consecutive pixels in buffer are close together, they are regarded in the same cluster, else a new cluster has been created, and the new pixel added to it. At the end, the cluster center with the highest frequency will be adopted as the background pixel value at location (x,y).

Algorithm 1: **background modeling algorithm**

Input : gray scale video

Output : the background model (reference Frame)

```
Step 1: read N frame sequence from the input video sequentially.
Step2: maintain an independent clustering model for each pixel (x,y)
    2.1 classify pixel intensity value at location (x,y) for N frames based on online clustering using algorithm 2.
Step 3 : calculate cluster centers to the cluster with the highest frequency
    3.1 search for the cluster with the highest frequency .
3.2 obtained cluster summation of the cluster with the highest frequency
3.3 calculate candidate cluster average
Candidate cluster average = $C_{high}(x,y) / m_{high}(x,y)$ // where high represent the cluster number with the highest frequency
step 4: select the background intensity value :
  4.1 adopt cluster average of the cluster with the highest frequency as a background pixel value at location ( x, y )
Step 5: repeat step 2-4 so that the entire pixel of N frames will be classified.
```

Algorithm 2: **clustering algorithm based on relative run length**

Input : a buffer contains **N** pixels

Output : clusters and frequencies of clusters of that buffer

```
Step 1: parameter initialization.
        i = 1 \\ set pixel sequence to 1
        n = 1 \\ set cluster number to 1
        $C_n(x,y) = f_1(x,y)$ \\ set cluster one summation to $f_1(x,y)$
        $m_n(x,y) = 1$ \\ set cluster one frequency to 1
T= [ 7 10 ]        \\ set threshold value to the range from 7 to 10 according to the video illumination changes
step 2: store pixels in buffer
store N pixels at location (x,y) into buffer
```

```
sort the pixels in the buffer in ascending  manner  so that the buffer
                buffer = { $f_1(x,y)$, $f_2(x,y)$ , ... ,  $f_N(x,y)$ }
Step3 :  repeat steps ( 3.1, 3.2 )  until i = N
    3.1   if | $f_i(x,y) - f_{i+1}(x,y)$ | < T     \\ if any consecutive pixels in buffer are close together
         Then
            $C_n(x,y) = C_n(x,y) + f_{i+1}(x,y)$ \\ add new pixel value to cluster summation
            $m_n(x,y) = m_n(x,y) + 1$          \\ increment cluster frequency by 1
          else                 \\ create a new cluster
            n = n + 1          \\ increment cluster number  by 1
            $C_n(x,y) = f_{i+1}(x,y)$ \\ set new cluster summation to the value $f_{i+1}(x,y)$
            $m_n(x,y) = 1$         \\ set new cluster frequency to 1
    3.2   i = i+1;             \\ increment pixel sequence by 1
 Step 4 : return cluster summation  { $C_1(x,y)$, $C_2(x,y)$ , ... ,  $C_N(x,y)$ }
          return cluster frequency   { $m_1(x,y)$, $m_2(x,y)$ , ... ,  $m_N(x,y)$ }
end.
```

5.1 Background updating

Due to the dynamic change in the scene, which strongly effected on the background image therefore, background image has to be updated repeatedly and/or selectively. In this work it has been proposed to use two approaches for updating the background which are:

1. In the first approach, it has been proposed to update the background in a pre-specified period of time (i.e. at regular time, usually each (5-10) minutes). Hence, the last image sequence in that period of time which length is about 100 frame as usual will be utilized to reconstruct the background image, which will be used as a reference image in the next (5-10) minutes. This approach can be used in indoor environment, also it is applicable in outdoor environment when the illumination changes slowly or in stationary background.
2. In the second approach the difference between the current frame and the reference image is *regularly* calculated , and if it is found that the number of altered pixel is exceed a predefined threshold (usually 50 %), that means the background is strongly changed, and hence, a reference image has to be reconstructed one more time. This approach is applicable in indoor environment when the lights turned on/off severely, or in outdoor environment in dynamic background (background with numerous motion changes).

5.2 Object detection

In order to evaluate the accuracy of the resulted background, first the objects in the video stream should be extracted, then a number of evaluation measure are employed to evaluate the results.

For object detection purpose, the manual thresholding technique has been adopted, where the value of the threshold is determined empirically. To do so, we have consider the reconstructed background (reference image) as B(i,j) while the frame that contains the foreground object (target) as F(i,j). In order to detect the object in the target frame, the absolute difference between the target frame and background image has to be calculated, then the result is thresholded. Equation (6) below is used to classify the target frame pixels as either a moving object (labeled as 1) or background pixel (labeled as 0) , as follow :A

$$F_t(i,j) = \begin{cases} 1 & \text{if } |F_t(i,j) - B(i,j)| > T \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

This operation is repeated for all pixel location (i,j) for the video frames under consideration. Threshold value selection is of high significance, if the selected value is too small, then the background pixel can be incorrectly classified as moving object (foreground), if the selected value is too large , then some foreground pixels can be incorrectly classified as background pixels.

Theoretically , the threshold value is considered in the range (0 – 255) , while in the considered experiment , the adopted manually selected threshold value is fixed in the range (40-50).

Algorithm 3 below, read frames sequentially from the inputted video, then a background subtraction operation is performed between the current frame and background image (reference frame) , then depending on a comparison between the background subtraction value and a predefined threshold, a decision is made to label the pixel under consideration as an object or background pixel.

Algorithm 3: **object detection algorithm**

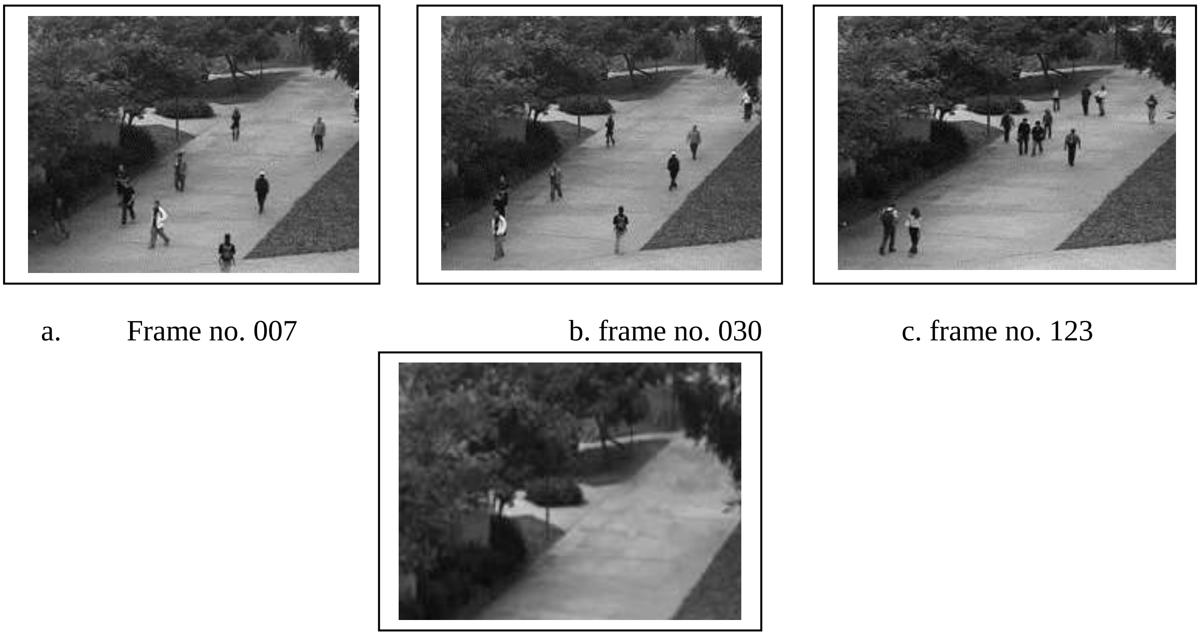
Input : gray scale video , background image (reference frame), threshold value
Output : video with detected object (labeled as 1)

```
Step 1:  Read  frames  sequentially  from the inputted video.
Step2:  For  each frame      // starting with the first frame and ending with the last.
        2.1 for each pixel F(i,j)   in frame  //  i = 1,..., frame high
                                              //  j = 1,..., frame width
2.2     Calculate the absolute difference between  video pixel F(i,j) and background image pixels B(i,j)
2.3     If  the absolute difference  greater than the Threshold  value
            Set  F(i,j) to 1 // labeled as an object
     Else  Set  F(i,j) to 0 // labeled as an background
2.4 end for  //  repeat steps ( 2.1-2.3)  so that the entire pixel of  the frame  under consideration will be classified.
Step 3 : end for   // repeat step 2  so that the entire frames of the inputted video  will be manipulated.
```

6. Experimental Results

The proposed algorithms has been implemented in C# programming languages, under visual studio 2015 environment, it is run on PC with COREi3 2.13 GHz processor, having RAM of 4 GB and windows 7 operating system.

In order to illustrate the effectiveness of the proposed method, three types of video stream has been used to test it, which are pedestrian walking in park video from UCSD-Anomaly data set, pedestrian walking in campus video from Virat data set and vehicles moving in highway video obtained from internet. Figures-(1, 2, 3) below illustrate three background images constructed from three video clips selected from the above three datasets respectively. Figure-4 below shows the result of the background subtraction operation for three frames selected from the vehicles moving in highway video based on the background constructed by using the proposed method.



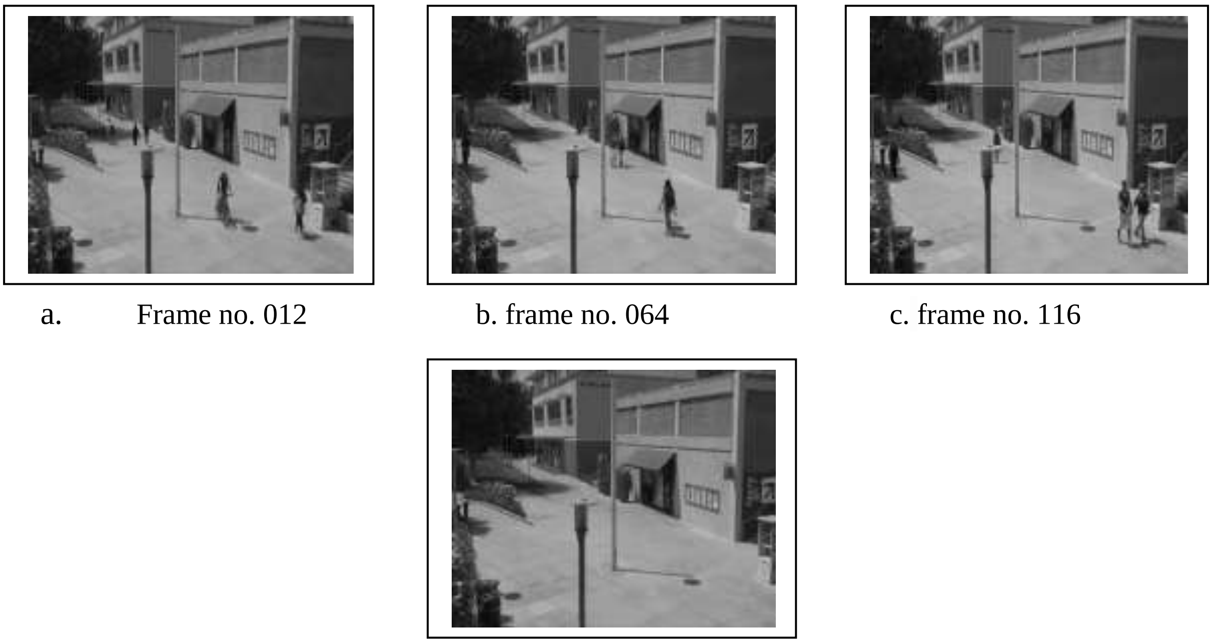
a. Frame no. 007

b. frame no. 030

c. frame no. 123

d.estimated background

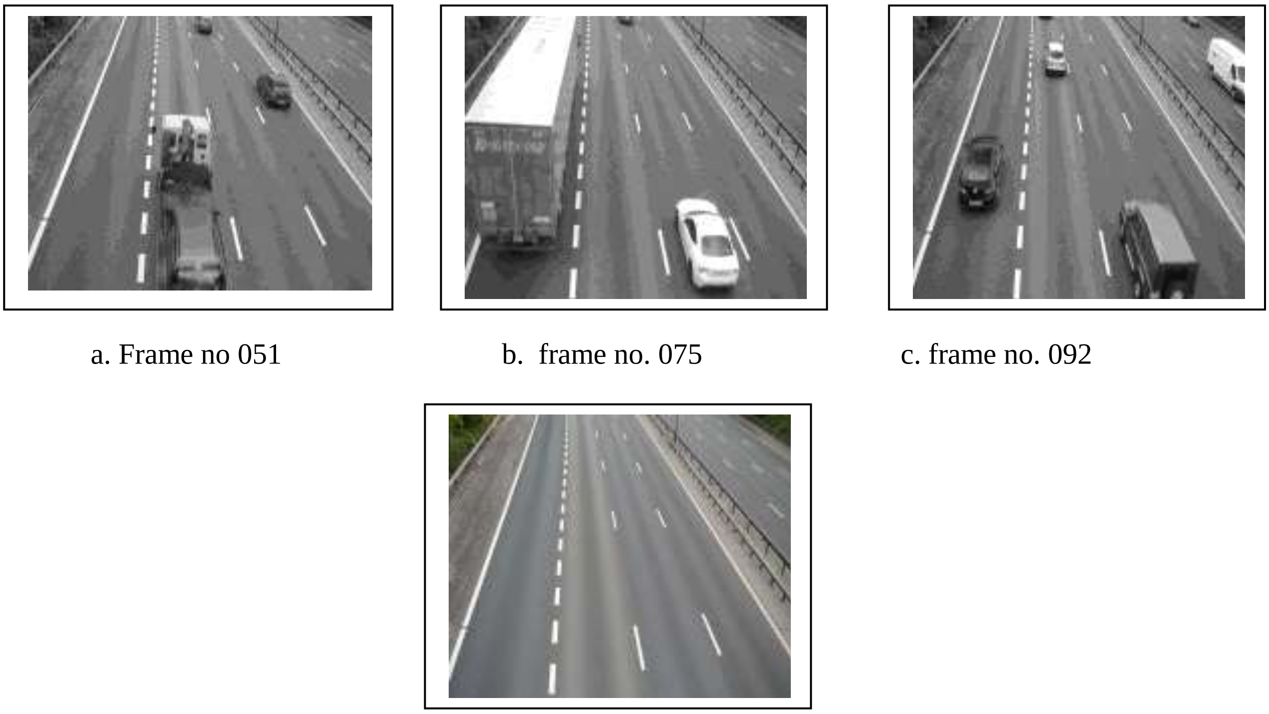
Figure 1- (a-c) frames from UCSD-Anomaly data set (d) its reconstructed background



a. Frame no. 012 b. frame no. 064 c. frame no. 116

d. estimated background

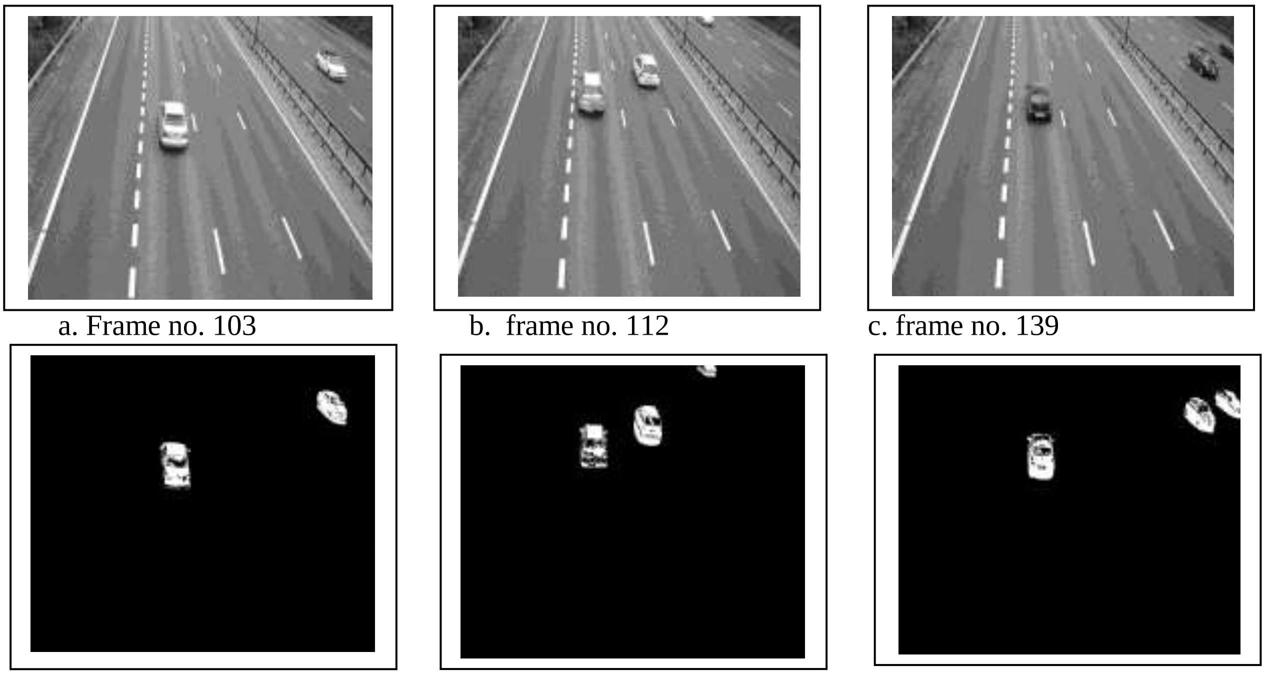
Figure 2- (a-c) frames from Virat data set (d) its reconstructed background.



a. Frame no 051 b. frame no. 075 c. frame no. 092

d. estimated background

Figure 3-(a-c) frames of vehicles moving in highway (d) its reconstructed background.



a. Frame no. 103 b. frame no. 112 c. frame no. 139

d. BGS of Frame no. 103 e. BGS of frame no. 112 f. BGS of frame no. 139

Figure 4- (a-c) frames of vehicles moving in highway (d-f) frames after background subtraction.

To validate the proposed method, the results obtained by it are compared with those obtained by conventional TABI, Median and PIC using k-mean clustering.

In term of accuracy precision, recall and F-measure performance measures has been used to measure the accuracy of the constructed background image. By applying those three measures on the foreground (moving object) image resulted from subtracting a selected frame from the constructed background image and the ground truth image for foreground which has been labeled manually. It is observed from Table-(1, 2 and 3) that higher Precision , Recall and F-measure has been obtained for the proposed method as compared with traditional background modeling methods.

Table 1- the average Recall, Precision and F-measure for UCSD-Anomaly video sequence.

method Measures	Precision	Recall	F-measure
TABI	0. 63	0.92	0.75
Median	0. 67	0.84	0.76
PCI	0.85	0.88	0.86
Proposed	0.87	0.94	0.90

Table 2- the average Recall, Precision and F-measure for Virat video sequence.

method Measures	Precision	Recall	F-measure
TABI	0. 65	0.91	0.76
Median	0. 73	0.91	0.81
PCI	0.84	0.97	0.90
Proposed	0.90	0.96	0.93

Table 3- the average Recall, Precision and F-measure for vehicles moving in highway video sequence.

method ╲Measures	Precision	Recall	F-measure
TABI	0. 69	0.91	0.78
Median	0. 72	0.98	0.83
PCI	0.87	0.96	0.91
Proposed	0.91	0.97	0.94

From computation time point of view, it is concluded from Table-4 that the proposed method is not the faster method, but experimental results show that the proposed method is fast enough so that it still cope with *real time* requirement.

Table 4- the average no of seconds (time) required to construct **one** BG pixels from a buffer of size 50,100,150 respectively by Applying a number of Traditional Methods (TABI, Median, PCI) and the proposed method.

method ╲no.of frames	50 frame	100 frame	150 frame
TABI	0.000176	0.000337	0.0010606
Median	0.000561	0.008812	0.0632377
PCI	0.009947	0.0825507	0.1923907
Proposed	0.0022813	0.0510166	0.12240419

7. Conclusions

Based on performance measures results which is obtained from the experiment, it is found that proposed method outperform the other conventional background construction methods in term of it's accurately construction the background which lead to good object detection when applying background subtraction method.

The proposed method can be considered to be not consume a lot of computational time, although it is not faster than other conventional methods, but it can attain the real time requirements, of background update.

Although the proposed method can overcome the second and third difficulties which the ordinary PIC suffer from (as mentioned in section 3), it still has the ability to construct the background image accurately and rapidly.

References

1. Cai, X., Ali, F. H. and Stipidis, E. **2008**. Rubust Online Video Background Reconstruction Using Optical Flow and Pixel Intensity Dsitribution, IEEE publication in the ICC 2008 proceedings, PP: 526-530.
2. Cai, L. and Jiang, Y. **2012**. An Effective Background Reconstruction Method for Video Object Detection, Third International Conference on Networking and Distributed Computing, IEEE, PP: 161-165.
3. Xiao, M., Han, C. and Kang, X. **2006**. A Background Reconstruction for Dynamic Scenes, 9th International Conference on Information Fusion.
4. Hou, Z. and Han, C. **2004**. A Background Reconstruction Algorithm Based on Pixel Intensity Classification in Remote Video Surveillance System, Preceding of the Seventh International Conference on Information Fusion , Fairborn. United State, PP: 754-759.
5. Subudhi, B. N., Ghosh, S. and Ghosh, A. **2013**. Change Detection for Moving Object Segmentation with Robust Background Construct Under Wronskian Framework, *SPRINGER*.

6. Haifeng, S. and Chao, X. **2013**. Moving Object Detection Based on Background Subtraction of Block Update, 6th International Conference on Intelligent Networks and Intelligent Systems, IEEE, PP: 51-54.

7. Mohanty, A. and Shantaiya, S. **2015**. a Survey on Moving Object Detection using Background Method in Video, Proceedings on National Conference on Knowledge, Innovation in Technology and Engineering (NCKITE).

8. Brutzer, S., Hoferlin, B. and Heidemann, G. **2011**. Evaluation of Background Subtraction Techniques for Video Surveillance, in Proceeding of IEEE Conference in Computer Vision and Pattern Recognition (CVPR), pp: 1937-1944.

9. Zainuddin, N. A., Mustafah, Y. M., Shafie, A. A., Rashidan, M. A. and Aziz, N. N., **2014**. Adaptive Background Reconstruction for Street Surveillance, 5th International Conference on Computer and Communication Engineering, IEEE, pp: 232-235.

10. Mei, X. and Lei, Z. **2008**. Using Modified Basic Sequential Clustering for Background Reconstruction. *Information Technology Journal*, **7**(7): 1037-1042.

11. Niranjil K. A. and Sureshkumar C. **2015**. Background Subtraction in Dynamic Environment Based on Modified Adaptive GMM with TDD for Moving Object Detection, *the Korean Institute of Electrical Engineers*, **10**(1): 372-378.

12. Devi, R. B., Chanu Y. J. and Singh, K. M. **2016**. a Survey on Different Background Subtraction Method for Moving Object Detection. *International Journal For Research in Emerging Science and Technology*, **3**(10).

13. Patil, P. A. and Deshpande, P. A. **2015**. Moving Object Extraction Based on Background Reconstruction. *International Journal of Innovative Research in Computer and Communication Engineering*, **3**(4): 2725-2731.