# Robust Online Video Background Reconstruction Using Optical Flow and Pixel Intensity Distribution

Xiaodong Cai, F.H.Ali, E.Stipidis
School of Science and Technology
University of Sussex
Brighton, East Sussex, BN1 9QT, UK
F.H.Ali@sussex.ac.uk

*Abstract*—**Obtaining a dynamically updated background reference image is an important and challenging task for video applications using background subtraction. This paper proposes a novel algorithm for online video background reconstruction. Firstly, multiple candidates of background values at each pixel are obtained by locating subintervals of stable intensity in a processing period. Then criteria based on pixel intensity distribution and local optical flows are employed to decide the most likely candidate to represent the background. For the methods utilizing the distributions of intensity values, the decision of determining the background value at a pixel position is based on the observation that the appearance time and sub-period frequency of the background is higher than non-background. An enhanced method using neighborhood optical flow information is adopted for more precise decision with slightly additional computation by identifying the events of covering and revealing of a pixel position. The experimental results show that the proposed algorithm outperforms existing adaptive mixture Gaussian background model and provides robust, efficient background image reconstruction in complex and busy environment.**

***Keywords-video; background subtraction; optical flow; Gaussian background model***

## I. Introduction

Background subtraction is a simple but effective solution for video object detection and segmentation (VOS) [1]–[4]. This technique involves comparing the observed frame with an existing reference background image. The reference image can be obtained using a pre-stored image [3], which is simple but can easily suffer from light condition and other changes due to the none adaptive mechanism. Moreover, it is difficult or impossible to control the areas being monitored, such as emptying the foreground objects in online situations. Alternatively, reference background image can be obtained by methods like adaptive learning procedure [5]. However, these solutions are not efficient due to the slow learning and fail to achieve sensitive detection in complex and busy environments.

The Adaptive Mixture Gaussian Background Model (AMGBM) [1], [4], [6] has attracted a lot of attention, especially for outdoor and long-term applications. It assumes that an initial model can be obtained by using a short training period. However, it is not a straightforward task to achieve such an initialization owing to two reasons: Firstly, in busy environments, the exposure of a certain background position could be very short and frequent. Secondly, the learning rate of AMGBM initialization is slow, for example, if it is assumed that the background is present at least 70% of the time and the learning rate is initialized to 0.002 [6], then it takes about 356 frames for the pixel values to be included as part of the background. The situation can be worse in busy environments. Therefore, model initialization is a critical step for robust and efficient background modeling.

To address the above problems, a novel integrated algorithm is proposed for automatic, efficient and robust online background reconstruction using optical flow estimation and pixel intensity distribution analysis. Performance of the proposed algorithm is studied practically and compared with the AMGBM.

The remainder of this paper is organized as follows. In section II, the diversity of pixel intensity distribution is analyzed. Following that, the proposed algorithm is described in III. Then the experimental results are presented and discussed in IV. Finally, this work is concluded in section V.

## II. Analysis of Pixel Intensity Distribution

The target of the online background construction is to determine the best candidate from a sequence of intensity values to represent the background of each pixel position within a processing period. In practical applications, due to the dynamic and complex environment, the background of a certain position is irregularly covered and exposed by turns. Therefore, locating the time instance when the background is exposed is critical. We assumes that a background pixel is constructed in the condition that it is revealed at least a short period of time continually. Also, the camera is approximately stationary, only small background motion is allowed.

To investigate the distribution of pixel intensity in a position, let $\{f_1, f_2 ... f_n\}$ represents a sequence with $n$ frames. Taking the intensity of a pixel over the entire sequence to form a time-intensity data sequence $P$, then

$$P = \{a_1, a_2 ... a_n\} \tag{1}$$

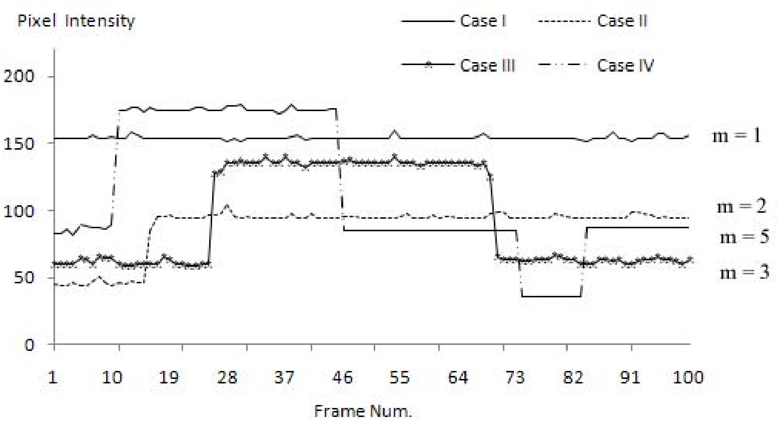


Figure 1. Typical cases of pixel intensity distribution in a training period

where $a_i$, ($i \in [1, n]$), is the intensity of a pixel at frame $i$ and $a_i \in [0, 255]$ in this work. Though the distribution of this data sequence is diverse, it was found in our studies that, for practical applications, it can be classified into four typical regular distributions as shown in Fig. 1. Each distribution can be divided into $m$ time intervals over the sequence:

$$P = \{S_1, S_2 \ldots S_m\} \tag{2}$$

where $S_k$ ($k \in [1,m]$) is the $k^{th}$ time interval starting from time $t$ given by:

$$S_k = \{a_t, \ldots a_{t+(N_k-1)}\} \tag{3}$$

where $N_k$ is the number of pixel intensity values in $S_k$, also the following conditions hold:

$$n = \sum_{k=1}^{m} N_k \tag{4}$$

$$N_k \geq T_a \tag{5}$$

where $T_a$ is the minimum number of frames to form a valid interval. The choice of $T_a$ depends on the video objects' motion complexity of the visual scene. The interval $S_k$ is also constrained by:

$$\forall a_x, a_y \in S_k \qquad |a_x - a_y| \leq T_k \tag{6}$$

where $T_k$ is the maximum difference of values within the interval $S_k$ and can be chosen experimentally. Then an Interval Intensity $V_k$ is calculated for the $k^{th}$ interval as follows:

$$V_k = \arg\max_h H_k(a_h) | a_h \in S_k \tag{7}$$

where $H_k(a_h)$ represents the density of value $a_h$ in $S_k$ by histogram calculation. From equation (2) and (7), the analytical form of the data sequence becomes:

$$P = \{S_1(V_1), \ldots S_m(V_m)\} \tag{8}$$

where $S_k(V_k)$ represents that all the pixel intensity values in interval $S_k$ is replaced using $V_k$. From the discussion above, the pixel intensity distribution function $f(t)$ is derived mathematically as follows:

$$f(t) = \begin{cases} V_1 & t \in (1, N_1) \\ V_2 & t \in (N_1 + 1, N_1 + N_2) \\ \cdot & \\ \cdot & \\ V_m & t \in \left( \sum_{k=1}^{m-1} N_k + 1, \sum_{k=1}^{m} N_k \right) \end{cases} \tag{9}$$

Based on equation (9), the practical meanings are analyzed for the four typical cases in Figure 1 and other irregular situations not presented in the figure.

1) **Case I**: $m$ =1. The fluctuation of the pixel intensity is within the threshold $T_k$ in equation (6) through the entire sequence. This indicates two possible situations: a background pixel is never covered or an object never moves during the investigated period. It is worth of noting that, for online background reconstruction, if a "semantic" object is not moving, then it will be considered as part of the background unless the history of this pixel is "memorized" or a long term background model is used.

2) **Case II**: $m$ = 2 and $V_1 \neq V_2$. It indicates that only one significant change of the pixel intensity takes place during the training period. Possible situations could be that a background point is being covered since the transition moment by a moving object (referred to as *covering event*), or, a still object starts moving at the transition moment resulting in the exposure of a background pixel (referred to as *revealing event*). The background value can be found before a covering event or after a revealing event. Therefore, identification of these two events becomes essential.

3) **Case III**: $m$ =3. If $V_1 = V_3 \neq V_2$, the most likely situation is that a background pixel is covered during t ∈ $(N_1, N_1 + N_2 - 1)$ and then revealed after this period. This is particularly true when $N_2$ is a small value. However, when $V_1 \neq V_3 \neq V_2$, the situation gets complex because any one of $V_1$, $V_2$ and $V_3$ can be considered as background.

4) **Case IV**: $m \geq 4$. ($m$ = 5 in Figure 1). Additionally, at least two elements in the group $\{V_1, V_2, ..., V_m\}$ are the same. It demonstrates a normal situation in many practical systems: a background position is revealed many times during the training procedure. Multiple moving objects (or different parts of a moving object) cover and pass the same position in different time durations. This is especially true in a busy environment, such as highway and train stations.

5) **Case V**: For cases not discussed above, $m$ is bigger ($m\geq$ 6) and none of the elements in the group $\{V_1, V_2, ..., V_m\}$ are the same. These situations rarely can be seen in practical systems. Assuming that the maximum delay tolerance is 100 frames (about 4 seconds in real-time applications) for online reconstruction and $m$=6, this suggests an extremely busy environment or objects moving with high speed. In this case, reconstruction is hard to achieve.

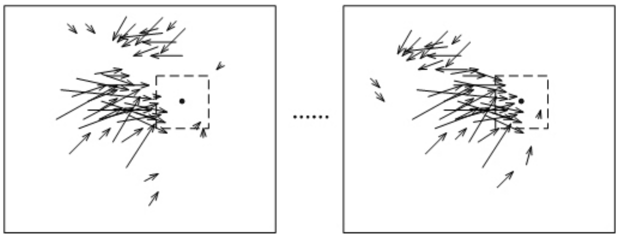
According to the analysis above, a novel algorithm is proposed with three criteria for background pixel intensity determination. Details are given in next section.

## III. ALGORITHM DESCRIPTION

Base on the discussion on section II, there is no argument on solving the problem in Case I. To address the problem as analyzed in Case II and the first situation in Case III, we first consider to distinguish the events of revealing and covering. It was found in [7] that, local optical flow can be used to indicate the movement of a pixel's neighborhood. By identifying the optical flow features of both events then the question is answered. A *covering event* is shown in Fig. 2. When the direction of the flow fields in the neighborhood of a pixel is towards the pixel itself, it indicates that a background point is being covered. Fig. 3 demonstrates a *revealing event*. When the direction of the flow fields in the neighborhood of a pixel is away from the pixel itself, it indicates that a background value is being revealed.

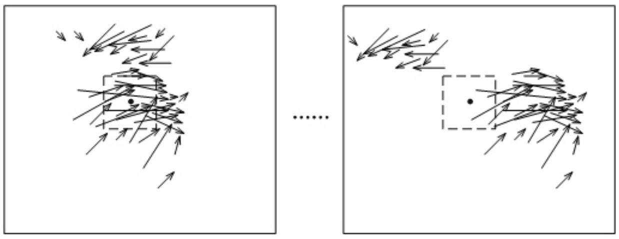
Base on above consideration, $T_a$ frames are taken before and after the time $t$ for optical flow estimation. Firstly, optical flow for each consecutive pair of images in position $(x,y)$ are calculated by using a Lucas-Kanade optical flow estimator [8]. As a result, $T_a$ vectors are obtained. For the $i^{th}$ vector, assuming that it's head and tail are located in$(x_{ih}, y_{ih})$ and $(x_{it}, y_{it})$, then an *approaching flow* $f_a(t)$ of a pixel at time $t$ is defined as:

$$f_a(t) = \sum_{i=(t-T_a)}^{t} (\sqrt{(x-x_{it})^2+(y-y_{it})^2} - \sqrt{(x-x_{ih})^2+(y-y_{ih})^2}) \quad (10)$$



(a) time $t$ (b) time $(t+n)$

Figure 2. neighborhood optical flow of pixel in covering event



(a) time $t$ (b) time $(t+n)$

Figure 3. neighborhood optical flow of pixel in revealing event

Similarly, a *revealing flow* $f_r(t)$ of a pixel at time $t$ is calculated in equation (11).

$$f_r(t) = \sum_{i=t}^{t+T_a} (\sqrt{(x-x_{ih})^2+(y-y_{ih})^2} - \sqrt{(x-x_{it})^2+(y-y_{it})^2}) \quad (11)$$

The event is determined by the net flow $f_n$:

$$f_n = f_a(t) - f_r(t) \quad (12)$$

if $f_n \geq Th_e$, where $Th_e$ is a positive threshold, then a *covering event* is detected, if $f_n$ is negative, and $|f_n| \leq Th_e$, then a *revealing event* is identified. In the case of *covering event*, the *Interval Intensity* presented just before the pixel is covered is considered to be background value. For *revealing event*, background value of current position is defined as the Interval Intensity obtained after the pixel is revealed. This method is called *optical-flow rule* in our algorithm.

However, optical flow estimation is highly computationally expensive. It is not practical to use the above method in situations where pixel intensity change significantly and more frequently such as Case IV and V. In addition, in the second situation in Case III, background pixel can not be determined using local optical flow.

For the situations which are not specified in Case II or the first situation of Case III, a closer look is taken at practical systems like surveillance. It is found that monitored objects normally occupy comparative small areas in the scene. Distributions of most pixel intensity are not classified into Case II and the first situation of Case III. This indicates that computational expensive motion estimation can be avoided for most pixel positions. What is more, the number of intervals presenting background pixel is higher than that of other pixel values. Therefore, for the situations not in Case II nor the first situation of Case III, the *Interval Intensity* having the highest density is consider to be the background value. This is referred to as *interval rule*.

Finally, it is also adopted that, in the situations where any two *Interval Intensity* values are not equal, or the *m* is a big number, such as Case V, the background value is determined by the observation that background pixel is stay longer than other objects in the same position. This is referred to as *time rule* in our algorithm. A completed background reconstruction algorithm is given in Table 1 where $V_b$ stands for the background value.

TABLE I. PROPOSED BACKGROUND RECONSTRUCTION ALGORITHM

```
1. for $k = 1$ to maximum number of pixels in the frame
    2. for $i = 1$ to $n$: Read pixel intensity sequence
    3. define the distribution function $f(t)$
    4. switch (Case) {
       Case I:
           $V_b \leftarrow V_1$; break;
       Case II:
           $V_b \leftarrow$ optical-flow rule; break;
       Case III:
           if ($V_1 = V_3$)
               $V_b \leftarrow$ interval rule; break;
           else
               $V_b \leftarrow$ time rule; break;
        default:
          if $H(V_i) \geqslant 2$, $V_i \in \{V_1, V_2, \ldots V_m\}$
               $V_b \leftarrow$ interval rule; break;
          else
               $V_b \leftarrow$ time rule; break;
5. end
}
```

## IV. EXPERIMENTAL RESULTS AND ANALYSIS

In this section, experimental results obtained demonstrate the performance of the proposed algorithm and compared with the AGMBM [6].

Two video sequences were selected in our experiment. The "highway" sequence describes a very common scene in outdoor surveillance. Some original frames are shown in Fig. 4. There are a few fast-moving cars. However, because the camera lens covers a wide range of view in the scene, the observed car speed in the far end is quite slow and it is much higher when cars close to the camera. The "busy street" sequence in Fig. 7 presents a very busy scene where many types of video objects are present with different directions, speeds and motions. For both sequences, $T_a$=5 and $T_k$=15, they are chosen experimentally.

The first experiment is carried out for the "highway". Background frames are reconstructed based on the proposed algorithm and the AMGBM algorithm as shown in Fig. 5 and Fig. 6, respectively. Both algorithms are analyzed in the frames numbered 60th, 90th, and 105th. For the proposed algorithm, one frame is dropped from every three frames to reduce the computational complexity. Our experiments show that this frame dropping has little influence on the performance of the proposed algorithm but the computation complexity is significantly reduced by about 1/3.

In the 60th frame in Fig. 5 and Fig. 6, both background images are not very well reconstructed due to the fact that information given by the first 60 frames is insufficient. Both algorithms produce a false detection in the far end areas where slow moving cars are present. However, it can be clearly observed that the proposed algorithm still outperforms the AMGBM, the area of false detection specified within the white rectangle in Fig. 5(a) is much less than that in 6(a).

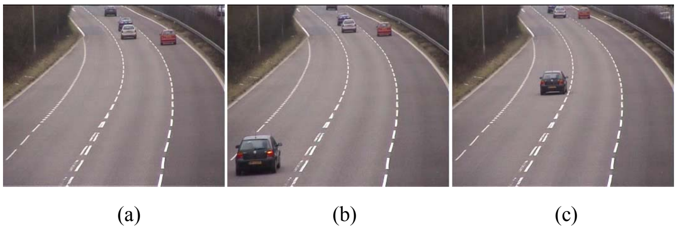


Figure 4. Video sequence "highway" (a) 1st frame (b) 25th frame (c) 90th frame

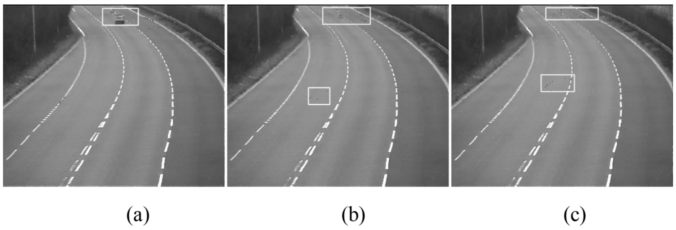


Figure 5. Background image reconstructed from "highway" using the proposed algorithm (a) 60th frame (b) 90th frame (c) 105th frame

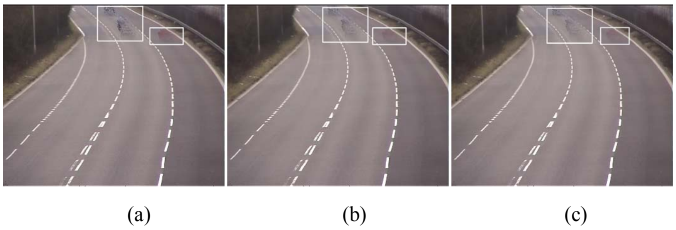


Figure 6. Background image reconstructed from "highway" using the AMGBM algorithm. (a) 60th frame (b) 90th frame (c) 105th frame

Improved performance in both cases can be observed as soon as the algorithms receive more frame information. When the proposed algorithm runs up to the 90th frame, Fig. 5(b) clearly shows that most exposed pixel positions have been merged into the background picture, only very tiny areas exist in the far end. Further improvement is immediately observed in the 105th frame in Fig. 5(c), only few pixels are not classified as background.

The results of the AMGBM algorithm are illustrated in Fig. 6. Although the reconstructed background is improved, big blurred areas are still present in the far end and this situation was continually observed up to 250th frame. This is due to the slow adaptation of the model discussed earlier.

The second set of experiment was carried out on "busy street" where more activities and different types of motions are present. The results of the proposed algorithm and the AMGBM are demonstrated in Fig. 8 and Fig. 9 respectively. Not surprisingly, similar results are obtained from both sequences. In "busy street", the proposed algorithm achieved a clear reconstructed background frame using about 75 frames, but the performance AMGBM is poorer as soon as slow moving objects are present. The Fig. 9(c) shows that, the AMGBM gives false detection to a slow moving car and integrates it into the background in the right-top corner.

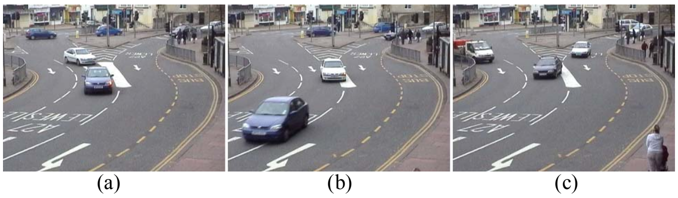


Figure 7. Video sequence "busy street". (a) 1st frame (b) 23rdframe (c) 90thframe

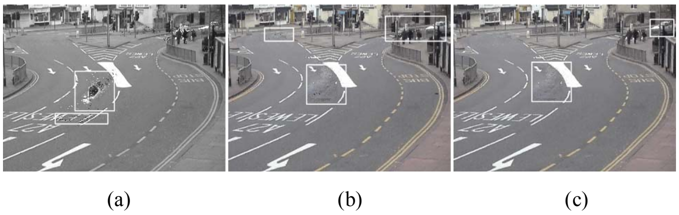


Figure 8. Background image reconstructed from "busy street" using the proposed algorithm. (a) 48th frame (b) 60thframe (c) 75thframe

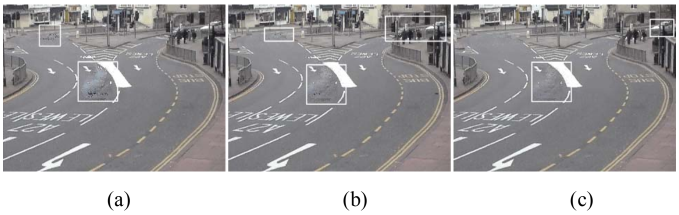


Figure 9. Background image reconstructed from "busy street" using the AMGBM algorithm. (a) 48th frame (b) 60thframe (c) 75thframe

## V. CONCLUSION

A novel algorithm for video background reconstruction is presented and implemented in this paper. The proposed algorithm is shown to be stable and outperforms the AMGBM in different scenes and situations in real videos. In most cases, the algorithm can achieve clear background reconstruction results within 100 frames (about 4 second in real-time applications). Methods based on intensity distribution are more efficient but have less accuracy in some cases while methods utilizing local optical flow can achieve more precise results with slightly additional computation. The integrated algorithm designed provides robust and efficient performance. It is worthy of pointing out that, the proposed algorithm can be easily extended to be a long-term background model. In that case, a history memory window could be integrated with this algorithm.

## REFERENCES

[1] P. KaewTraKulPong and B. R, "An Improved adaptive background mixture model for real-time tracking with shadow setection," in Proceedings of the 2nd European Workshop on Advanced Video-based Surveillance Systems, Kingston upon Thames, 2001.

[2] A. M. McIvor, "Background subtraction techniques," unpublished, http://www.mcs.csueastbay.edu/tebo/Classes/6825/, Retrieved April 2006.

[3] J. Pan, C.-W. Lin, C. Gu, and M.-T. Sun, "A robust spatiotemporal video object segmentation scheme with prestored background information," in Proceedings of the IEEE International Symposium on Circuits and Systems, Arizona, USA, vol. 3, pp. 803-806, May 2002.

[4] C. Stauffer and W. E. L. Grimson, "Learning patterns of activity using real-time tracking," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 8, pp. 747–757, 2000.

[5] Y. Zhang, Z. Liang, Z. Hou, H. Wang, and M. Tan, "An adaptive mixture gaussian background model with online background reconstruction and adjustable foreground mergence time for motion segmentation," in Proceedings of the IEEE International Conference on Industrial Technology, pp. 23– 27, Dec. 2005.

[6] C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Collins, CO, USA, vol. II, pp. 246–252. June 1999.

[7] D. Gutchess, M. Trajkovic, E. Cohen-Solal, D. Lyons, and A. Jain, "A background model initialization slgorithm for video surveillance," in Proceedings of the Eighth International Conference on Computer Vision, Vancouvier, Canada, pp. 733–740. July 2001.

[8] L. B. D and K. T, "An Iterative Image Registration Technique with an Application to Stereo Vision," in Proceedings of Imaging Understanding Workshop, pp. 121–130, 1981.