

# A Novel Robust Statistical Method for Background Initialization and Visual Surveillance

Hanzi Wang<sup>1</sup> and David Suter<sup>1</sup>

<sup>1</sup> Department of Electrical and Computer Systems Engineering  
Monash University, Clayton 3800, Victoria, Australia  
{Hanzi.wang, D.suter}@eng.monash.edu.au

**Abstract.** In many visual tracking and surveillance systems, it is important to initialize a background model using a training video sequence which may include foreground objects. In such a case, robust statistical methods are required to handle random occurrences of foreground objects (i.e., outliers), as well as general image noise. The robust statistical method *Median* has been employed for initializing the background model. However, the Median can tolerate up to only 50% outliers, which cannot satisfy the requirements of some complicated environments. In this paper, we propose a novel robust method for the background initialization. The proposed method can tolerate more than 50% of foreground pixels and noise. We give quantitative evaluations on a number of video sequences and compare our proposed method with five other methods. Experiments show that our method can achieve very promising results in background initialization: including applications in video segmentation, visual tracking and surveillance.

## 1 Introduction

Visual tracking and surveillance has gained a wide range of applications including monitoring freeways [1], recognizing human action [2, 3], motion segmentation [4], etc. Background subtraction, which detects changes from a background model, is a crucial step in these applications. To extract foreground objects, one usually needs to model the background scene using a short training video sequence.

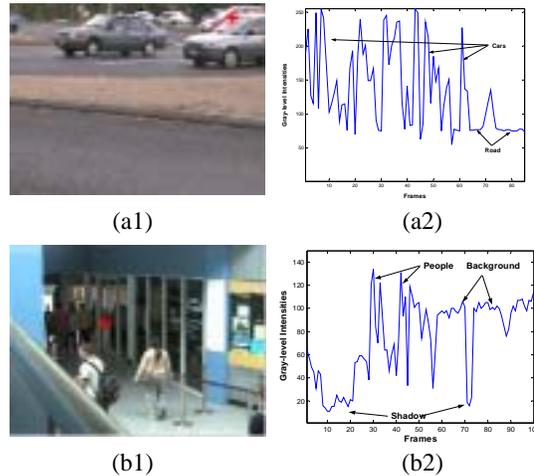
There are a number of methods (for example, [2, 3, 5-8]) that have been proposed for modeling background scene in recent years. Simple methods represent background features by an average of either grey-level or color samples at each pixel over a training time. Pfinder [3] is one of the examples. It assumes that the values of the pixels, over a time window at a particular image location, are Gaussian distributed. Such kind of methods does not address scenes with dynamic backgrounds, or where foreground objects are present in the training stage. Some methods have been proposed to model dynamic background scenes, for example, Mixture of Gaussians (MOG) [5, 8, 9]. In MOG, the background features are characterized by a mixture of several Gaussians. Each Gaussian represents a distribution per pixel. Thus, MOG can efficiently model dynamic background scenes. However, when the background involves a wide distribution in color/intensity, modeling the background with a mixture

of a small number of Gaussian distributions is not efficient, when foreground objects are included in the training frames, MOG does not work well and it will misclassify [6].

Among the above-mentioned methods, almost all of the methods require that the training sequence is *free of any foreground objects*. In practical cases, for example, in a busy road or in a public area, it is hard to control the environments. Such a requirement can not be always satisfied. We must initialize the background model in a way that is robust to the presence of foreground objects in the background training data. In contrast to background model representation and model maintenance, only a few studies of background model initialization have been made (e.g., [1, 4, 10, 11]).

For example, the authors of [11] proposed a Smoothness Detector (SD) Method. They assumed that a background value always has the longest stable value. They employed a moving window along time at each pixel to search for the stable intervals. However, we find one problem of the method is that when the data include multi-modal distributions (i.e., some modes from foreground objects and some modes from background as shown in Fig. 2 and Fig. 3), and when the modes from foreground objects tend to be relatively stable, this method can not differentiate these modes from those from the background.

In order to decide the window length ( $L$ ) and the intensity flicker of the window ( $T_f$ ) for each pixel, the authors of [11] also proposed an Adaptive Smoothness Detector (ASD) method. Because the ASD method tries different  $L$  and  $T_f$  at each iteration until the solution is found, the computational cost of the ASD method is greatly increased.



**Fig. 1.** Two examples that background is visible less than 50 percent of the training time: (a1) and (b1) show one frame of each training sequence; (a2) and (b2) show the intensity distributions over time at one pixel (marked by red star) of the sequence.

Motivated by [11], a Local Image Flow (LIF) algorithm [10] was proposed. Two steps are used: in the first step, all stable sub-intervals in a training sequence are located for each pixel. In the second step, the method locates the sub-interval with the greatest average likelihood using local motion information, and produces background

value by computing the mean value over the chosen sub-interval. Optical flow is computed for each consecutive pair of images and used to estimate the likelihood. While this potentially adds valuable information, most optical flow computation methods themselves are computationally complex and very sensitive to noise.

In [1], the authors used the Median intensity value over observations at each pixel, to initialize the background for a traffic monitoring system. The underlying assumption is that the background at each pixel can be seen for more than 50 percent of time in the training sequence. However, the requirement that background appear more than 50% of time in a video sequence may not be always satisfied. Fig. 1 illustrates two such examples. In Fig. 1, we can see that the background value at the marked pixel (with red star) is visible less than 50 percent of the training time. The noise is either from the moving foreground objects or the shadows of the foreground objects.

A robust method which can tolerate more than 50% of noise is possible [12]. Examples include RANdom Sample Consensus (RANSAC) [13], Adaptive-Scale Sample Consensus (ASSC) [14], etc. To overcome the problems inherent in methods based on the Median, we propose a new robust method for background initialization. The major advantage of the proposed method is that it can tolerate over 50% of noise (including foreground pixels) in the data. The essential idea of the proposed method has been previously published in [15] which was restricted to only the background initialization problem. This paper also provides applications of the proposed method to video segmentation, visual tracking and surveillance.

This paper is organized as follows: in Sect. 2, we propose a new robust method for background initialization. In Sect. 3, experiments showing the advantages of, and applications of, our method are provided. We conclude in Sect. 4.

## 2 The Proposed Method for Background Initialization

### 2.1 Assumptions

In our method, we make some assumptions which are similar to those in [10, 11]:

1. The background at each pixel should be revealed at least for a short interval during the training period.
2. A background value tends to be relatively stable and constant.
3. A foreground object can remain stationary for a short interval in the training sequence. However, the interval should be no longer than the interval from the revealed static background.
4. The background scene remains relatively stable.

Stability is one characteristic of essentially stationary backgrounds. The foreground value at a pixel is assumed to have no less variance in grey-level intensity than a background value.

### 2.2 The Proposed Method

We employ a two-step framework:

(1) locate all non-overlapping stable subsequences of pixel values; (2) choose the most reliable subsequence (from which we use the mean value of either the grey-level intensities or the color intensities over that subsequence as the model background value).

In the first step, we use a sliding window with a minimum length  $L_w$  to locate all stable sub-intervals  $\{l_k\}$  (similar to [10, 11]). For a test sequence of  $N$  frames, we have  $N$  observations at each pixel  $\{x_i | i = 1, \dots, N\}$ . Let  $x_{l_k(t)}$  be a pixel value of the  $k$ th subsequence  $l_k$  at time  $t$ . The  $k$ th stable subsequence candidate should satisfy:

$$\forall (t-1, t) \in l_k, \begin{cases} |x_{l_k(t)} - x_{l_k(t-1)}| \leq T_f \\ |x_{l_k(t)} - \bar{x}_{l_k(t-1)}| \leq T_f \end{cases} \quad (1)$$

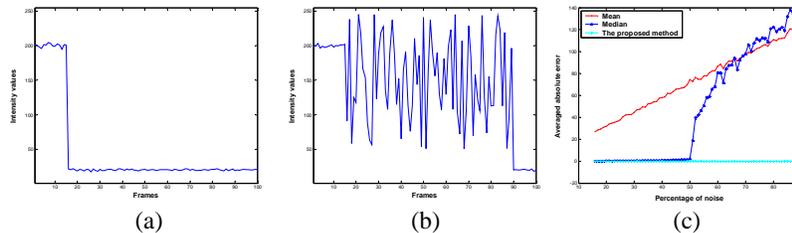
where  $\bar{x}_{l_k(t-1)}$  is the mean value from the beginning of the subsequence  $l_k$  to time  $t-1$ .

If we cannot find any candidate subsequence with a minimum length  $L_w$  we use the longest stable candidate subsequence. We experimentally set  $L_w$  to 5 and  $T_f$  to 10, for all test sequences. Note: even after this step, the chosen subsequences can contain pixels from foreground, background, shadows, highlights, etc. (e.g., see Fig. 1 b).

The second step is a crucial step, because in this step, a reliable subsequence, which is most likely to arise from the background, will be chosen. Our definition of reliability is motivated by RANSAC [13]. We build in to our objective function the notions of consensus and of scale estimation. We consider both the number ( $n$ ) of data points “agreeing” with a model (contained in the candidate interval), and the distribution of these data (e.g. standard variance  $S$ ):  $n$  should be large, and  $S$  should be small. We define our objective function as finding the most stable interval from the non-overlapping sub-intervals  $\{l_k\}$  by:

$$\hat{l}_k = \arg \max_k (n_{l_k} / S_{l_k}) \quad (2)$$

where  $n_{l_k}$  and  $S_{l_k}$  are respectively the number of values (length) of, and the standard variance of, the observations in the  $k$ th subsequence  $l_k$ .



**Fig. 2.** Estimating background value from noisy data: (a) and (b) illustrate two cases of the distributions of the simulated data; (c) the results obtained by the three methods.

To illustrate the robustness of the proposed method we generate synthetic data to simulate the observations over time at a pixel. One hundred data values (i.e., 100 frames) were generated. The first fifteen data values (i.e., a relatively stationary foreground object pixel) have intensity value of 200 and standard variance of 2. From the sixteenth to the  $i$ 'th data, we simulate random noise (such as foreground objects in

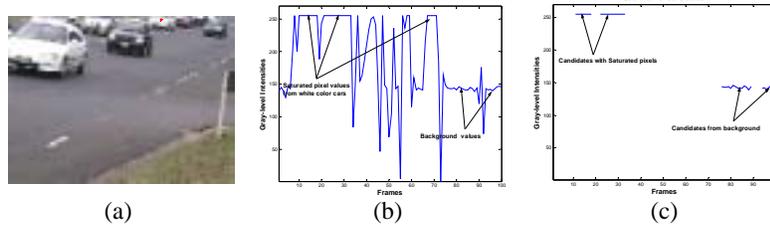
transit at that pixel) with intensity values ranging from 50 to 250. We simulate a background value in the sub-interval from the  $(i+1)$ 'th data to the 100th data, with unit variance. We increase  $i$  value from 16 to 90 with step 1 each time. We repeat the experiment ten times and output the average value.

Fig. 2 (c) shows the results of finding backgrounds by three statistics: Mean, Median, and the proposed method. We see that the Mean is not robust to noise at all. The Median can only tolerate noise occupying less than 50 percent of the data. In contrast, the proposed method is much more robust.

However, we note that equation (2) might be erroneous when  $s_k$  is very small.

This can happen when some pixels of a short subinterval have saturated colors. The saturated pixel values are clipped within the range from 0 to 255 and sequences containing these saturated pixels have a very small (or zero) standard variance [16]. For this case, the assumption (1) in Sect. 2.1 is violated. When we detect such a case happen, we use the following equation instead of equation (2):

$$\hat{l}_k = \arg \max_k (n_k) \quad (3)$$



**Fig. 3.** One example showing that the intensities of the saturated pixels are clipped: (a) one frame of the test sequence. We investigate the grey-level intensity distribution of the observations at one pixel, which was marked with a red star. In (b), we can see that there are some saturated pixels corresponding to white colored cars. The sub-interval candidates obtained in the first step are shown in (c). The two candidates corresponding to saturated pixels have a standard variance of zero. In such case, we should use equation (3) instead of equation (2).

Fig.3 shows an example where the intensities of some saturated pixels are clipped.

### 3 Experiments

#### 3.1 Background Model Construction Test

The test sequences are recorded by a Canon MV750i digital video camera. We stored the sequences at a resolution of 160x120, and a sample rate of five frames per second. We have deliberately chosen different background including both indoor and outdoor scenes (including within these scenes, foreground objects, shadows, highlights, and illumination changes to simulate true situations that a visual surveillance system may meet in practice).

		Training sequences	Mean	Finder	Median	SD	ASD	Proposed Method
R1	S1							
	S2							
R2	S1							
	S2							
TS	S1							
	S2							
SC	S1							
	S2							
PS	S1							
	S2							

**Fig. 4.** Ten video sub-sequences of five test videos. The third column shows one frame of each training subsequence; the remaining columns show the difference between the background and the background estimate obtained by the competing methods. The results obtained by the proposed method are shown in the last column.

**Road1 (R1):** Heavy traffic in daytime (some shadows on the road).

**Road2 (R2):** Vehicles passed by a crossing road in the evening. Some parts of the road were highlighted when vehicles (with lights on) got close to those parts.

**Train Station (TS):** A gate of a train station. Many people exited or entered the station through that gate.

**Sport Center (SC):** In an indoor sport center, people walked through a corridor. Shadows of people were cast on the glass wall and the floor of the corridor. Also some illumination changes happened when people exited the back door and covered the light outside.

**Pharmore Shop (PS):** A pharmacy shop, which is located inside a big shopping center. People walked in front of the shop. The illumination of the background scene sometimes changed because of the reflected sunlight outside the shopping center.

We compare the proposed method with five other methods. All of the methods perform at pixel-level for background initialization (in contrast to methods that use region level analysis). To test each method, we choose two sub-sequences (S1 and S2) which include a number of frames ranging from 30 to 100 in each sub-sequence, from each test sequence. To evaluate the performance of each method, we employ three criteria, similar to those used in [10]: a) the Average gray-level Error (AE); b) the Number of Error pixels (NE); and c) the Number of Clustered error pixels (NC). We use the Mean value of Total error (MT) of the ten sub-sequences over each criterion as the overall measurement for each method.

We generate a Reference Frame (RF) for each test sequence by using the mean value of selected frames that are free of foreground objects. An error pixel is one whose grey-level value differs from the value of the reference pixel by a threshold 20. We define a clustered error pixel when the 4-connected neighbors of that error pixel consist of more than 4 error pixels.

**Table 1.** Experimental results by different methods on test sequences.

		R1		R2		TS		SC		PS		MT
		S1	S2	S1	S2	S1	S2	S1	S2	S1	S2	
Mean	AE	9.61	10.27	5.79	9.10	5.81	14.12	11.69	8.75	26.48	25.12	12.67
	NE	2994	2369	1630	1320	1323	4992	3537	3102	10253	9799	4132
	NC	2965	2273	1571	1231	1211	4811	3436	3023	10031	9677	4023
Pfinder	AE	9.14	9.17	6.25	6.01	5.50	20.89	10.08	9.92	12.99	38.82	12.88
	NE	2790	2127	1917	411	1125	7805	3402	1822	3969	12690	3806
	NC	2752	2016	1866	312	1042	7605	3347	1699	3746	12573	3696
Median	AE	5.14	4.58	2.69	3.45	2.89	4.15	6.63	3.14	9.51	8.49	5.07
	NE	352	159	276	142	40	353	1349	271	2559	2092	759
	NC	282	127	239	114	28	296	1301	247	2347	1947	693
SD	AE	7.99	5.94	2.83	5.58	2.96	3.50	6.10	2.77	7.85	5.43	5.10
	NE	2097	976	515	872	226	399	1304	217	1400	921	893
	NC	2018	840	487	741	153	228	1195	181	961	603	741
ASD	AE	5.59	6.01	2.43	3.58	2.66	2.81	7.62	2.94	6.47	4.64	4.48
	NE	588	252	114	55	44	56	892	123	598	559	328
	NC	443	152	82	11	22	0	819	15	420	306	227
The proposed method	AE	4.33	4.32	2.05	3.00	2.54	2.77	2.81	2.46	6.27	4.36	3.49
	NE	70	10	57	37	21	63	76	51	541	484	141
	NC	23	0	23	4	7	5	28	15	296	238	64

Fig. 4 shows one frame of each test subsequences and the resulting error pixels (corresponding to the white color pixels), obtained by the five other methods and the proposed method. A quantitative comparison is given in Table 1. From these results, we can see that the Mean and the Pfinder methods are the most inaccurate in background initialization. The Mean takes all observations at each pixel in the test subsequence into account. The Pfinder, using a temporal smoothing technique, gives larger weight value to recent observations. When the observations contain pixels from other than background, these two methods break down.

Compared with the Mean and the Pfinder, the Median method achieves a much better result because of its robustness to noise (from foreground objects, shadows, etc.). However, when the test subsequence includes too many foreground objects, or if the background value is visible for less than 50 percent of the test subsequence (more noticeable, in the S1 of Sport Center sequence, and in the S1 and S2 of the Pharmore Shop sequence), the Median method fails to estimate the background.

SD obtained more accurate results than the Median in the SC and PS sequences, but less accurate results in the R1, R2, and TS sequences. ASD achieves better results than the SD method in all test sequences because it uses different window length  $L$  and  $T_f$  at each pixel location. However, the cost is about 30-50 times slower than SD in computational time.

Among the six methods, the proposed method achieves the most accurate results and it also is about three times faster than SD, and about 100 times faster than ASD.

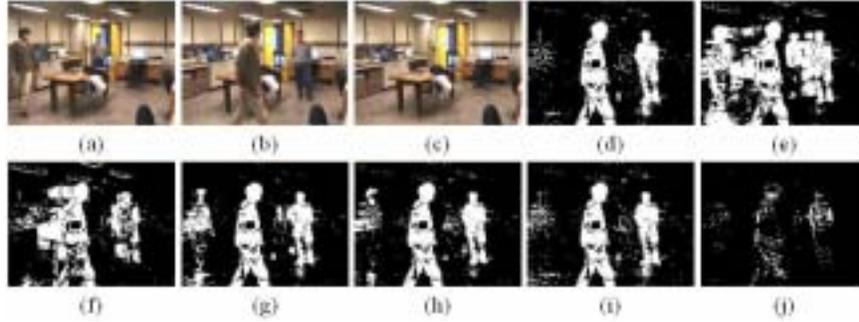
### 3.2 Applications

The proposed method can be applied in a wide range of practical computer vision tasks such as video segmentation, vehicle surveillance, tracking, etc. Fig. 5 and Fig. 6 show the application to segmentation and tracking.

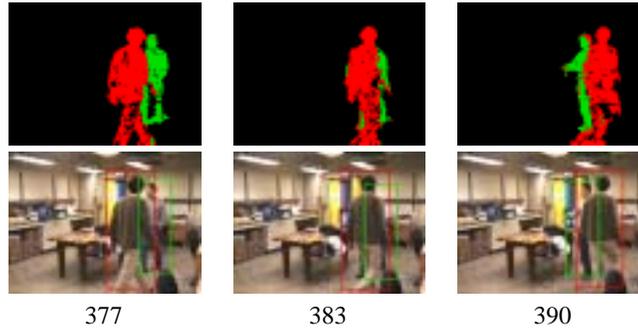
In Fig. 5, we use an image sequence from <http://www.ecse.rpi.edu/~cvrl/humanbody/>. The sequence shows an office with several people walking around. Almost every frame of the sequence includes people. The ground truth background image is not available. We use frames 310 to 359 as training images, which include two people walking around (Fig. 5 (a) and (b) show frame 310 and frame 359 of the image sequence). We initialize the background using the six methods. Because the MOG method is frequently used in many vision tasks, in this experiment, we also include MOG.

From Fig. 5, we can see that the proposed method outputs an accurate initialized background, and thus, it effectively extracts foreground objects (i.e., in this case, the two people). Because the Mean and Pfinder methods can not tolerate outliers at all, they totally broke down when the training frames include foreground objects. Although the Median method is robust to noise and outliers, it breaks down when data involves more than 50 percent. Thus, we can clearly see there is a ghost in the detected foreground. SD and ASD work better than the Mean, Pfinder, and Median. However, we can still see a ghost in the detected foreground in the result of SD. Although ASD produced a result close to that of the proposed method, the result of ASD is less accurate and the computational time is much higher. The result of MOG tends to give less false positive but more false negative pixels. This is because MOG blindly treats the pixels of the persons as background modes in the training stage.

In Fig. 6, we use the background initialized by the proposed method as the background model to segment/track people inside the office. The segmentation and tracking results (on frames 377, 383, 390) are shown in Fig. 6. We see the proposed method provides a good background initialization image for the tracking/segmentation system even when every frame in the training stage contains foreground objects.



**Fig. 5.** (a) and (b) are frames 310 and 359. (c) The initialized background image by the proposed method; (d) The detected foreground pixels by the proposed method; (f) to (j) are respectively the foregrounds obtained by Mean, Pfunder, Median, SD, ASD, and MOG.



**Fig. 6.** Segmentation and tracking results (sample frames 377, 383 and 390).

## 4 Conclusion

In this paper, a new robust method is proposed for the task of background initialization. The proposed method is very robust to outliers and can be used in many places where foreground objects can not be avoided in the training stage. One of the main strength of the proposed method is that it is highly robust to noise and outliers in data. The method is a great improvement over the traditional Median method.

We have evaluated our method on various environments including outdoor and indoor, daytime and nighttime, different illumination conditions. Comparisons with several other methods on background initialization show that our method can achieve very promising results even when the background is revealed much less than half of time in the training sequences. Furthermore, we show the method can be successfully used in video segmentation, tracking and surveillance.

## Acknowledgements

This work is supported by ARC grant DP0452416. This work was carried out within the Monash University Institute for Vision Systems Engineering.

## References

1. Gloyer, B., et al. Video-based Freeway Monitoring System Using Recursive Vehicle Tracking. in Proc. of IS&T-SPIE Symposium on Electronic Imaging: Image and Video Processing. 1995. p. 173-180.
2. Haritaoglu, I., D. Harwood, and L.S. Davis. W4: Real-Time Surveillance of People and Their Activities. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2000. 22(8): p. 809-830.
3. Wren, C.R., et al., Pfinder: real-time tracking of the human body. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 1997. 19(7): p. 780-785.
4. Cristani, M., M. Bicego, and V. Murino. Multi-level background initialization using Hidden Markov Models. in First ACM SIGMM international workshop on Video surveillance. 2003. p. 11 - 20.
5. Stauffer, C. and W.E.L. Grimson. Adaptive Background Mixture Models for Real-time Tracking. in Proc. IEEE Int'l Conf. on Computer Vision and Pattern Recognition. 1999. p. 246-252.
6. Toyama, K., et al. Wallflower: Principles and Practice of Background Maintenance. in 7th International Conference on Computer Vision. 1999. p. 255-261.
7. Harville, M. A Framework for High-Level Feedback to Adaptive, Per-Pixel, Mixture-of-Gaussian Background Models. in 7th European Conference on Computer Vision. 2002. p. 543-560.
8. Friedman, N. and S. Russell. Image Segmentation in Video Sequences: A Probabilistic Approach. in Proc. Thirteenth Conf. on Uncertainty in Artificial Intelligence. 1997. p. 175-181.
9. Harville, M., G. Gordon, and J. Woodfill. Foreground Segmentation Using Adaptive Mixture Models in Color and Depth. in IEEE Workshop on Detection and Recognition of Events in Video. 2001. p. 3-11.
10. Gutchess, D., et al. A Background Model Initialization Algorithm for Video Surveillance. in IEEE Int'l Conference on Computer Vision. 2001. p. 733-740.
11. Long, W. and Y.H. Yang. Stationary Background Generation: An Alternative to the Difference of Two Images. *Pattern Recognition*, 1990. 23(12): p. 1351-1359.
12. Stewart, C.V., Robust Parameter Estimation in Computer Vision. *SIAM Review*, 1999. 41(3): p. 513-537.
13. Fischler, M.A. and R.C. Rolles. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Commun. ACM*, 1981. 24(6): p. 381-395.
14. Wang, H. and D. Suter. Robust Adaptive-Scale Parametric Model Estimation for Computer Vision. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2004. 26(11): p. 1459-1474.
15. Wang, H. and D. Suter. Background Initialization with A New Robust Statistical Approach. *IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*. 2005. p. 153-159.
16. Horprasert, T., D. Harwood, and L.S. Davis. A Statistical Approach for Real-Time Robust Background Subtraction and Shadow Detection. in *ICCV'99 Frame-Rate Workshop*. 1999.