A Framework for False Positive Suppression in Video Segmentation

Min Han Tun, Geoff West and Tele Tan
Curtin University of Technology, Western Australia
minhantun@postgrad.curtin.edu.au
{g.west,t.tan}@curtin.edu.au

Abstract

Object detection in video surveillance is typically done through background subtraction or temporal differencing. While these techniques perform very well under scenes where there are minimal light changes, they begin to fail when the scene contains rapid illumination changes. The effects of this is most profound in indoor environments. Under these conditions, the background modeling techniques produce large numbers of false positives. This paper proposes a sequential approach to suppressing these false positives. Both frame and regions level spatial scales are considered to detect sudden light changes and make use of Gabor filter responses and edge maps to identify and remove false positives.

1. Introduction

With the rise in demand for intelligent surveillance and monitoring environments, the need for better performing systems capable of operating under diverse conditions has become a real necessity. The computer vision community has developed a wide variety of object detection and tracking systems to meet these needs [3, 4, 7, 8]. A fundamental task in this vision approach is the detection and identification of moving objects. The most widely used approach to detecting objects is to perform some form of background subtraction. There are many background subtraction algorithms mentioned in recent literature and most follow a common scheme of modeling the background with some form of image feature, typically pixel intensity.

It is well known that background modeling techniques suffer from two problems. The first being the stationary background problem: foreground objects get merged into the background when they become motionless. The second problem arises when sudden light changes occur in the environment. This paper is aimed at attempting to reduce the effects of this light change. Due to the inability of background modeling techniques to reflect these changes in the background model quickly, false positives are introduced. The introduction of false positives in the detection means that the accuracy of the detection is reduced and interferes with object segmentation and tracking.

It is very difficult, if not impossible, to use just one image feature to suppress these false positives. This paper proposes a sequential pattern recognition approach to suppressing false positives. A number of different image processing techniques are sequentially applied to the video frame to suppress the light changes, as shown in Figure 1. This gives us much flexibility in that different image features can easily be used to differentiate between foreground and background and to deal with different environmental conditions using available methods. The idea of sequentially adding image processing algorithms is based on the assumption that as we progress through the processing stages, the amount of data to be processed will reduce. Hence, we can add more complex algorithms while maintaining a reasonable amount of computational time.

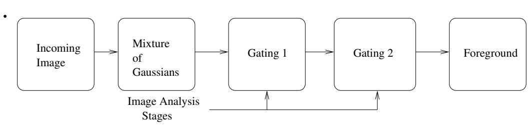

Figure 1. Sequential Image Analysis

This paper is targeted towards suppressing false positive in indoor video. Gaussian Mixture Models [8] are used as an initial step towards background modeling and object detection. Then a light change detection algorithm is used to identify frames that contain light changes. Once these frames are identified, a number of filters to eliminate false positives are then applied to these frames. Section 2 gives an over view of related work done in this area. Although there are many techniques available, this paper concentrates on using Gabor filters to use edge and texture information. The algorithms used to suppress light changes are discussed in section 3. Section 4 contains experimental data from this research and the paper is concluded in section 5.

2. Related Work

Many attempts at dealing with light changes have been developed over the years. Intensity normalization is an earliest example of an attempt at illumination independent background subtraction and is still in use today. Matsushita et al [5] uses time dependent intrinsic images with intensity normalization to suppress effects of lighting changes. An estimation of intrinsic images is done and a database of the results are constructed using Principal Component Analysis. Each incoming image is then compared against this database to get the corresponding illumination image. This illumination component is then removed from the raw image to achieve illumination independence.

Wallflower is one of the most effective systems when dealing with sudden light changes [10]. This system maintains a number of background models with different illumination conditions. When too many changing pixels are detected, it checks all background models and takes the one that produces the least foreground pixels to be the current background model. This reduces the amount of false positives produced since the background model closely models the current lighting conditions.

Homomorphic filtering [6] works well on scenes with Lambertian surfaces. In this technique, each image is considered a combination of illumination and reflectance. The relative location of illumination and reflectance can be identified in the frequency domain. A log function is applied to the pixels to make the illumination and reflective components additive. Illumination is assumed to change slowly over space and a low pass filter is applied to remove illumination.

Xie et al [13] uses order consistency to reduce the effects of light changes. They suggest that if we consider the gray values of two pixels, corresponding to two points in the background scene in the image as illumination change over time, the ordering of pixel values are preserved over a range of conditions. This order consistency is not observed for object pixels. Xie et al use this observation coupled with a statistical model to build a change detection system under sudden illumination change.

Tian et al [9] integrates texture analysis into a Gaussian Mixture Model of [8] to deal with fast light changes. The idea here is that the texture of the background and the foreground produced by sudden light change is relatively the same. The foreground regions where texture measure is greater than a predefined threshold is considered to be background and removed.

Greenhill et al [2] improves on the Median Value model allowing the threshold to adapt to changing lighting conditions. A history of previous average illumination for a certain number of frames is maintained. This historical data is then used to calculate the worst case approximation of disparity between current and historical median values. Once a correction factor is applied to this disparity value it is added to the threshold allowing the threshold to adapt to changing illumination. This model performs well on scenes with global illumination changes but still cannot handle spatially non-uniform illumination change.

Much research has also gone into using edge information to perform image segmentation and object detection [12]. These methods are typically based on building an edge map and finding the difference between the frames to obtain the moving part of the image. Edge based methods alone are particularly sensitive to image noise level. Fan et al [1] proposed a edge based region growing technique to segment images producing better results than using just edges alone.

Of relevance to this paper, Toyama et al [10] stated that background models should consider changes in different spatial scales. It is difficult to detect sudden illumination change at pixel level and more often than not it will require a region or frame level decision.

3. False Positive Suppression

The first step towards object detection is to model the background and remove it from the scene. Then we use an algorithm to detect the frames that contain sudden light changes. Once these frames are identified, a number of image processing algorithms are applied to identify and remove the false positives in those frames. The following Figure illustrates the steps taken in this work.

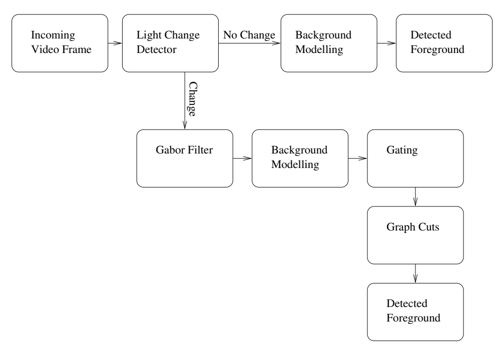

Figure 2. False Positive Suppression Steps

3.1. Adaptive Background Modeling

Not only frame and region levels need to be considered to efficiently suppress light changes but also a robust adaptive pixel level model is required to build a strong background model. There is a wide variety of background modeling techniques and among them Stauffer and Grimson's [8] algorithm is perhaps the most widely used background modeling algorithm. This is the model used in this paper.

A mixture of *K* Gaussian is used to model the background. The probability of a pixel *X*, at time *T* is defined as:

$$P(X_t) = \sum_{i=1}^{K} \omega_{i,j} * \eta(X_t, \mu_{i,t}, \Sigma_{i,t}) \tag{1}$$

where

$$\eta(X, \mu, \Sigma) = \frac{1}{(2\pi)^{\frac{n}{2}} |\Sigma|^{\frac{1}{2}}} e^{\frac{1}{2}(X_i - \mu_t)^\gamma \Sigma^{-1}(X_i, -\mu_t)} \tag{2}$$

and

$$\omega_{i,t} = (1-\alpha)\omega_{i,t-1} + \alpha(M_{k,t}) \tag{3}$$

where μ is the mean, α is the learning rate. After the Gaussian are ordered by ω/α the first *B* distributions satisfying the criteria of

$$B = \sum_{k=1}^{b} \omega_k > T \tag{4}$$

are taken to be the current background. This method has proved to be very effective in frames with slow or no light changes. Once lighting in the environment begins to change rapidly or when the change is spatially non-uniform, this method begins to produce large areas of false positives.

3.2. Gabor Filters and Gating

Once the frames with false positives are identified, Gabor filters are applied to the frame. Gabor filters are very good feature extractors, typically in texture analysis. A Gabor function can be thought of as a sinusoidal plane of a particular frequency and orientation modulated by a Gaussian envelope. The Gabor function used in this paper is described in [11] and can be written as:

$$e^{\{-[(x-x_0)^2+(y-y_0)^2]/(2\sigma^2)]\}} \sin \omega(x \cos\theta - y \sin\theta) + \varrho \tag{5}$$

where x_0 and y_0 specify the center of Gaussian. For the sinusoidal plane wave ω is the frequency and ϱ is the phase of the plane wave. Gabor functions have a tendency to produce responses to uniform illumination (DC component). The DC component is removed by subtracting the mean pixel from each pixel value of the filters. Gabor filters are created for one frequency and eight orientations. The filter sizes are set as 32x32 window and the sigma value set as 0.5. Figure 3 shows the intensity plot of the Gabor function for all eight orientations.

The reasoning behind the choice of Gabor filters is that they can simultaneously produce an edge map of the image as well as a set of Gabor responses based on the image's texture properties allowing the use of both edge and texture properties in identification and removal of false positives.

An artificial grid of five cells by five cells is placed on the background image as well as the foreground Image.

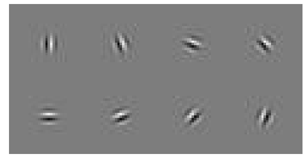

Figure 3. Intensity plot of Gabor function for all eight orientations

The reason for having this grid is that Mixture of Gaussian produces a large blobs containing both foreground and background in instances of light change. The grid helps to reduce the size of the blob and eliminate background portions of the blob. The Gabor responses in each cell are then measured for the foreground image and compared to the responses in the same region of the background image. The cells with the same responses are considered to be false positives and removed from the foreground.

Figure 4 (a) is the result of using Mixture of Gaussian model on Gabor edge maps and (b) is the result of removing the background edges with the gating process which shows significant reduction in false positives.

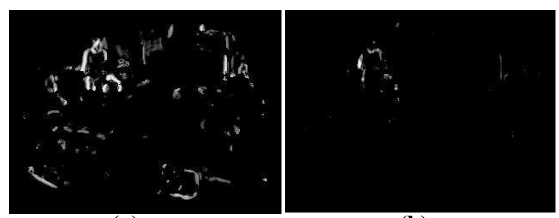

Figure 4. Edge Maps: (a) Result of using Mixture of Gaussian on Gabor output (b) result of running through the gating function on (a)

3.3. Graph Cuts

One side effect of placing an artificial grid is that the blobs that are left are square rather than having the shape of the true foreground object. In order to regain the silhouette of the image, a Graph Cut algorithm [14] is applied. A cut on the graph is then performed to partition the graph into two disjoint sets *S* and *T* such that the source *s* is in the set *S* and the sink *t* in the set *T* [14].

Once the directed graph that represents the image is created, with each pixel as a node in the graph, an edge is created between each of the eight neighbors of the nodes and edge weights set. Edge weights are calculated based on the color of the neighboring pixels. If the colors are similar the nodes are considered to be highly correlated and if the color difference is wide then they are less likely to be of the same object. The color of each node is first converted into its gray value and the difference between the gray values then becomes the cost of each edge. The edge weight function can be defined as:

$$cost = e^{(-1.0*(G_i - G_n)^2/2\sigma^2)} \tag{6}$$

where G_i and G_n are the gray values of the current node and its neighbor respectively. σ is the mean intensity of the image.

The edge map is then scanned to find leading and tailing edge pairs. The magnitude of both edges are then added together to form ranking criteria. The top ten ranking edge pairs are taken as belonging to the object and a foreground seed is placed as the mid point between the leading and tailing edge. Background seeds are then defined on either side of the leading and tailing edge. Figure 5 (b) shows the image with foreground and background seeds placed.

The connection between foreground seed and the source node of the graph is given a high weight indicating that this pixel and the pixels that are highly correlated to it are foreground and zero is set as the weight for the edge between the sink and foreground seed node. For the background seed nodes, the reverse settings are made. Once the seed positions and edge weights are defined, the Graph Cut [14] algorithm is used to segment the image regaining objects with proper boundaries.

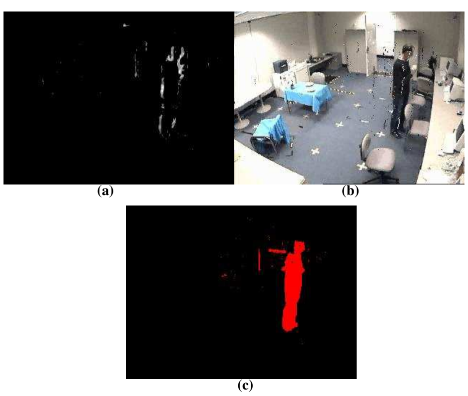

Figure 5. Seed placement for graph cuts. (a) Edge image with false positives cleaned. (b) Seed image the black dots are background seeds and the white are foreground seeds. (c) The result of applying graph cuts.

3.4. Light Change Detection

This paper uses a light change detection algorithm to identify the frames that contain light changes. Since these false positives are caused by light change we can safely say the the frames with rapidly changing illumination will contain false positives. This light change detection is based on the idea that the average illumination of the frames where light changes quickly will be far greater than the ones where there is no change. The average illumination is calculated as:

$$I_{avg} = \frac{\sqrt{(R_f - R_b)^2 + (G_f - G_b)^2 + (B_f - B_b)^2}}{total} \tag{7}$$

where R_f,G_f,B_f and R_b,G_b,B_b represents the RGB channel data of current and previous frames respectively and *total* is the total number of pixels in the $M * N$ frame.

The steps described in Section 3.2 and Section 3.3 are applied only to those frames with light change. This improves the speed of the system. The idea is that since Mixture of Gaussians are proven to perform well under constant or slowly changing light, it is not necessary to have false positive suppression.

4. Experimental Results

Several video sequences containing spatially uniform and non-uniform light changes are captured to test the accuracy of this system. The results obtained from these sequences are shown in the next section.

The captured videos are processed with Stauffer and Grimson [8] Mixture of Gaussian (MoG) model as well as our sequential approach. The ratio between the amount of pixels detected and the total pixels is used as a comparison metric. The sequences are first manually segmented to establish a ground truth. A number of frames picked out from this video are presented in Figure 6.

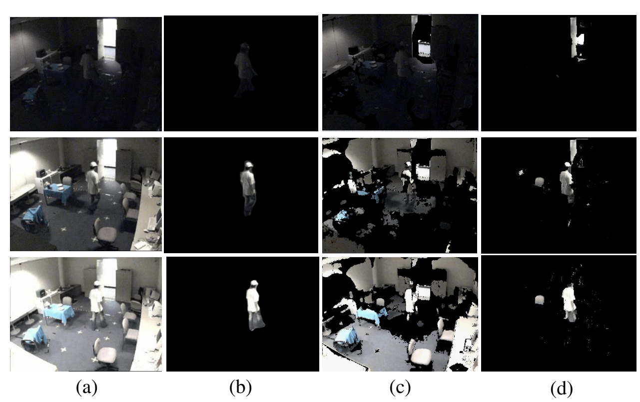

Figure 6. Comparison of detected results. (a) unprocessed image (b) ground truth (c) result of MoG (d) sequential approach

A comparison of these results over the several videos are shown in Table 1. As can be seen from the table, using sequential suppression can result in detection with significantly less false positives while still detecting the foreground object. From Figure 6, it can be seen that some under-segmentation occurs with this method. On Average about 3% of the object is undersegmented. This is due to two things: when the light goes off in the room the edges of the object become weak and it becomes hard to find proper seed points. The second reason because graph cuts segment the object into regions with similar colour. Consider a fair skined person wearing a black shirt. If foreground seeds are placed on the shirt alone graph cuts will produce only the shirt and not the head or hands since their colors are different. Table 1 shows results from different videos captured in the lab as well as those in a real kitchen.[1] The kitchen videos cannot be published due to privacy issues. Video 1 and 2 are taken in a lab environment with sudden changes of light(fluorescent light). The system was developed against these videos. Videos 3 and 4 are from a ceiling mounted camera capturing the daily activities of a kitchen in a typical suburban home.

Table 1. False positive pixels detected.

Videos	No of Frames	MoG (%)	Sequential Approach (%)
1	3000	51.97	2.11
2	2700	39.7	3.5
3	3500	53.8	2.34
4	1500	60.32	3.31

5. Conclusions and Future Work

As can be seen from Section 4 this method performs very well in detecting objects under rapid and spatially non-uniform light changes. This paper has described a framework to reduce false positives in a sequential manner. This system works by finding foreground edges, places seeds between these edges and then uses graph cuts to segment the foreground object. However, sometimes edges of false positives can be stronger than the actual foreground and they get marked as foreground. This can potentially be mitigated by incorporating temporal coherence in adjacent frames.

There is a need to further investigate ways of mitigating under-segmentation as mentioned in Section 4. The speed performance of this system may be improved by moving the Gabor Filters from spatial domain implementation to frequency domain implementation. This system can be further extended to incorporate other methods and image properties such as that used in Tian et al [9].

[1] The videos are available at http://www.cs.curtin.edu.au/∼13088015/.

References

[1] S. L. X. Fan, Z. Man, and R. Samur. Edged based region growing - a new image segmentation method. In *Proceedings of the 24th ACM SIGGRAPH International Conference on Virutal Reality and Its Applications in Industry*, pages 302–305, 2004.

[2] S. Greenhill, S. Venkatesh, and G. West. Adaptive model for foreground extraction in adverse lighting conditions. In *8th Pacific Rim International Conference on Artificial Intelligence*, volume 3157/2004. SpringerLink, 2004.

[3] I. Haritaoglu, D. Harwood, and L. S. Davis. W4: Real-time surveillance of people and their activities. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 22, pages 809–830, Aug 2000.

[4] K. Kim, T. H. Chalidabhongse, D. Harwood, and L. Davis. Background modelling and subtraction by codebook construction. In *International Conference on Image Processing*, volume 5, pages 3061–3064, Oct 2004.

[5] Y. Matsushita, K. Nishino, and M. Sakauchi. Illumination normalization with time-dependent intrinsic images for video surveillance. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 26, pages 1336–1347, Oct 2004.

[6] R. J. Radke, S. Andra, O. Al-Kofahi, and B. Roysam. Image change detection algorithms: a systematic survey. In *IEEE Transactions on Image Processing*, volume 14, pages 294–307, March 2005.

[7] C. Rother, V. Kolmogorov, and A. Blake. ”grab cut” - interactive foreground extraction using iterated graph cuts. In *ACM Transactions on Graphics*, volume 23, pages 309–314, 2003.

[8] C. Stauffer and W. E. L. Grimson. Adaptive background mixture models for real-time tracking. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 246–252, June 1999.

[9] Y.-L. Tian, M. Lu, and A. Hampapur. Robust and efficient foreground analysis for real-time video surveillance. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2005.

[10] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers. Wallflower: Principals and practice of background maintenance. In *IEEE Conference on Computer Vision*, 1999.

[11] M. R. Turner. Texture discrimination by gabor functions. *Biological Cybernetics*, 55(2-3):71–82, 1986.

[12] Z. Xiaoyan, Z. Rongchun, and M. Zhiqiang. A new method for moving object extraction automatically. In *IEEE International Symposium on Communications and Information Technology*, volume 2, pages 1460–1463, 2005.

[13] B. Xie, V. Ramesh, and T. Boult. Sudden illumination change detection using order consistency. *Image and Vision Computing*, 22:117–125, 2004.

[14] V. K. Yuri Boykov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. In *IEEE Transactions on pattern analysis and machine intelligence*, volume 26, 9, pages 1124–1137, 2004.