Object Detection through Edge Behavior Modeling

Adin Ramirez-Rivera adin@khu.ac.kr
Mahbub Murshed mmurshed@khu.ac.kr
Oksam Chae oschae@khu.ac.kr

Kyung Hee University
1 Seocheon-dong, Giheung-gu, Yongin-si, Gyeonggi-do 446-701, South Korea

Abstract

The detection of moving objects depends on the accuracy of the model used to represent the background. Common pixel-based and naive edge-based approaches have many drawbacks in dynamic environments, e.g.*, false detections with noise. We propose a novel background model that encodes the background as edges, building a statistical distribution per segment that represents the edge behavior. We build the background distributions using a kernel-based approach; the moving objects are detected as the edges that deviate from the distributions. The method does adaptive thresholding to the edges, which maintains their shape and boosts the detection accuracy. Sets of gradient distributions are incorporated into the model, to determine edges that lie within the distributions, but are moving edges. The number of distributions is handled dynamically, allowing them to increase and decrease accordingly to the situation. The experiments show that the proposed method improves the detection rates, due to its robustness against illumination changes.*

1. Introduction

Object detection is a common problem in computer vision; objects are segmented out from the background using different techniques. One of the most used techniques is background subtraction. It models the background, and then checks whether the elements in the frame are in the model; the elements that are not part of the model are called foreground. Furthermore, there are many approaches to do this task [7]. Background subtraction presents several challenges: (1) objects change its motion, moving objects stop and become part of the background while background objects start its march and become part of the foreground; (2) illumination changes in the scene, due to weather variations or artificial sources; (3) camera jitter, due to environment [11]; (4) and dynamic background, which presents a challenge to be detected correctly as background. Toyama *et al*. [10] presents a generalization of these problems.

Background subtraction methods use both statistical and non-statistical techniques to create and maintain a background model. Changes in the motion of objects have several challenges, being the statistical approaches a good solution. One instance of these approaches uses a set of distributions—Mixture of Gaussians (MoG)—to model each pixel intensity. The MoG method [8] is capable of adapting to slow illumination and objects motion changes. Nevertheless, it is not able to adapt to dynamic backgrounds. MoG updates each distribution with a fixed learning rate and number of Gaussians to maintain the background. Some authors use MoG with other features besides intensity, like texture information [9], or edges [5].

Most background modeling methods build the background model considering and evaluating a set of features per pixel, frequently being intensity. This leads to a high computational cost and important information loss due to ignore the spatial information. Some authors make use of neighborhood and context information, like Jain *et al*. [5]. They use edge maps to build a Gaussian Mixture Model, setting a fixed neighborhood to solve the edge movement problem. The per-pixel operations are sensitive to edge shape changes and movement, creating spurious pixels in the result that looks like noise.

Other edge-based methods have been proposed in the literature. Kim and Hwang [6] proposed an inter frame difference method based on the intersection of edges in two consecutive frames, to extract objects in video sequences for multimedia encoding purposes. In a similar way, Dailey *et al*. [4] proposed a method for surveillance, which is an inter frame difference approach that uses the gradient information. The authors subtract three consecutive frames to obtain the moving edges, and then extract the bounding boxes of the moving objects. However, the lack of a background model makes these methods prone to error and very sensitive to changes in the scene. Fig. 1 shows the edges in two consecutive frames, the color edges show the variation due to illumination and background movement. Therefore, robust background modeling techniques are needed to over-

(a) (b) (c)

Figure 1: Edge movement in consecutive frames. (a) and (b) show edges of two consecutive frames. (c) Shows the common edges (*black edges*), the edges of the map in (a) (*blue edges*) and the edges of the map in (b) (*red edges*). The edges present shape and position variations in consecutive frames.

come the position and shape (of edges) variation problem.

Our work models the scene using the behavior of edges; to overcome the problem with the variation of edges, we model the edges using distributions mined from their changes. Their shape and position variations are learned through a kernel-density approach; these distributions allow us to recover each edge variation range and shape deformation boundaries. Through the use of the distributions, we can match the background edges accurately, overcoming limitations of other methods, such as, fixed thresholds for each edge that incorporate errors because not every edge will present the same type nor the same amount of variations. The distributions allow us to recover the foreground edges, as those that do not lie within the distributions boundaries. However, the moving objects that lie within the distribution boundaries should be differentiated from the background model. To overcome this limitation, we introduce a set of gradient distributions for each edge, this improves the accuracy through a discrimination process based on the gradients.

The proposed approach can (1) cope with dynamic backgrounds and illumination changes, due to its robust background model; (2) solve the inherit problem of edges shape and position variation, by the kernel distribution encoding scheme; (3) adaptively threshold the distributions; (4) use complex modeling techniques because edges reduce the amount of elements to be modeled, in comparison with pixel-based methods that need to model each pixel in the image. Moreover, the proposed method produce moving edges, which can be used in tracking and recognition methods.

The present paper is divided as follows. Section 2 explains the proposed method; it describes the background modeling, and the object detection through the use of the model. Section 3 presents a comparison of our method with other methods and evaluates our results. Section 4 presents the conclusion of the paper.

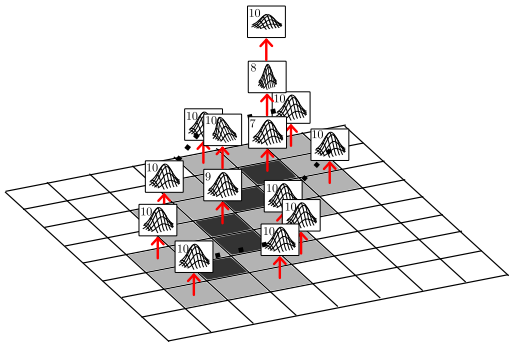

Figure 2: A distribution with the set of MGG on it, showing different gradient distributions with their frequency score for each pixel in the edge distribution.

2. Proposed Method

2.1. Background Modeling

The algorithm is based on the use of edges. However, most edge-based methods use edges in a pixel-based manner; we create a hybrid method in which the distributions of edges are modeled as segments, to make use of spatial relations among the pixels of an edge, and the matching is done pixel wise, to allow edges break because in the extraction process edges tend to merge together.

The background model consists of two parts: a statistical map (SM) and a set of mixtures of Gaussians of gradients (MGG). Fig. 2 depicts the background model, the gray levels denotes the accumulation of the distributions in the SM, while the MGGs are shown coming from each position in the distribution. The model is updated online; this same mechanism makes the algorithm able to cope with moving objects while it learns the background. The SM models the behavior and frequency of edges, and the MGG models the changes in the gradient due to illumination variations in the scene. The former is built from a set of quantized edges from the image. The edges are extracted using Canny edge detector [3], and then the edge map is binarized based on the presence of edges. For each edge, its frequency is accumulated over time and space using a kernel estimator; the distributions of edges are calculated through

$$E_t = \frac{1}{\sqrt{2\pi}h} \sum_{e \in Q_t} \sum_{x \in e} \sum_{p \in \mathcal{N}(x)} e^{-\frac{(p-x)^2}{2h^2}}, \qquad (1)$$

where E_t is the set of distributions for the frame t, Q_t is the binary edge map of the frame t, e is an edge from Q_t, x is the position of a pixel in the edge e, p is a position in the neighborhood of x ($\mathcal{N}(x)$), and h is the width of the kernel estimator. We choose to use a normal function, $N(0, \Sigma)$, to be our kernel estimator; it has the property of smooth the distributions while computes them. The smoothing of the edges overcomes the lack of a large amount of data to

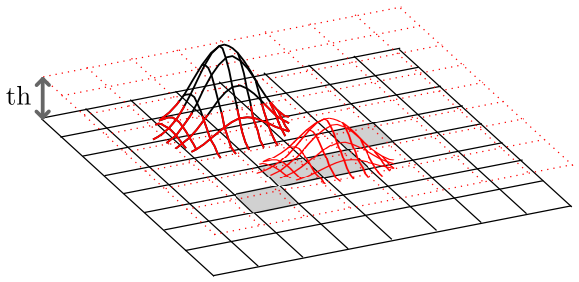

Figure 3: Statistical map threshold over the distributions height. The distributions below the threshold are not considered background.

model the variation of edges. In our experiments we use a kernel width of five pixels.

This procedure creates the distributions on the statistical map; the height of the distribution represents the frequency of edge, and the spreadness the edge shape variations. The SM is computed as the weighted mean

$$SM_t = (1-\alpha)SM_{t-1} + \alpha E_t, \tag{2}$$

where E_t is the set of distributions for frame t, and α is a learning rate. We defined it as $\alpha = 1/N$, where N is the number of frames an edge will be held in the model. In our experiments we use $N = 200$ frames.

Common approaches based on edges have the problem of using fixed thresholds to match the edges. However, this is not a good solution because edges change independently from each other. Hence, fixing the threshold for all the edges is not reasonable. The proposed method uses an adaptive thresholding method based on the statistical map. The SM is thresholded in two different ways. First, we separate the spatial stable edges from the noisy outliers by reducing the width of the distributions, leaving only those pixels within two and a half standard deviations from the mean of the distribution. To do this, the distribution is thinned and the peak is computed; from that peak the width of the distribution is computed and only those pixels within the range are kept, and the rest is set to zero. This procedure, ensures the preservation of the shape of the distribution, while removing the less probable position of edges from the distribution. Second, we only consider temporal stable edges, which are those distributions with a frequency above 60% of the number of frames for modeling. Figure 3 shows the threshold operation in the height; only the distributions over the threshold, th, are considered for the matching procedure against possible moving edges. The lower height distributions, as the one depicted in red, are kept and updated to preserve the scenes characteristics, but they are ignored for the matching procedure. Therefore, the edge that is showed in shades under the red distribution will be considered as moving edge. In other words, the background edges will be those edges that were present in more than 60% of the

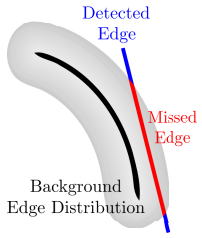

Figure 4: Overlapping miss classification problem. A moving edge is partially removed by a distribution; the red part of the edge is matched as background while the blue part is matched as foreground.

frames that the model holds ($th = 0.6N$). These two constraints ensure to have true background edge behavior in the statistical map; after this step, the SM can be used to verify if an edge is part of the background or not.

The edges are matched against the SM to remove the background from the scene, the edges that lie on a distribution are removed. However, foreground edges that lie over the background distribution will be mistakenly removed. Figure 4 shows this problem, where part of the moving edge is absorbed into the background. This problem is not exclusive of our statistical method, every edge-based method cannot distinguish between overlapping edges. However, an overlapping edge belonging to the foreground will present a different orientation with respect to the modeled background edge. Therefore, to avoid the overlapping problem, we incorporate gradient information to the distributions of edges to differentiate between them. The use of edges reduces the amount of data needed to model the scene, which allows us to easily incorporate more information without an overhead in the computing time.

Each distribution in the statistical map is modeled with gradient information also. Each pixel of the SM distributions holds a mixture of Gaussians of gradients (MGG) that is increased and pruned dynamically. Figure 2 shows a distribution with its set of MGG. For each new distribution pixel, one Gaussian is created to maintain the gradient information of the edge. Each Gaussian is represented with a mean, a standard deviation, and a frequency score, (μ, σ, f); the mean is set to the new gradient value, and the standard deviation and the frequency are set to an initial value. Each new gradient value, for each position of the distributions in the SM, is compared against the MGG; for each Gaussian in the set, the new value is checked against the distribution, if the value is within 2.5 standard deviations from the mean (such as the distribution marked by '✓' in Fig. 5 that matches the testing value), the corresponding Gaussian is updated through the weighted mean

$$\mu^t_{k,x,y} = (1-\alpha)\mu^{t-1}_{k,x,y} + \alpha g_{x,y} \tag{3}$$

$$\sigma^{t\,2}_{k,x,y} = (1-\alpha)(\sigma^{t-1}_{k,x,y})^2 + \alpha(g_{x,y} - \mu^t_{k,x,y})^2, \tag{4}$$

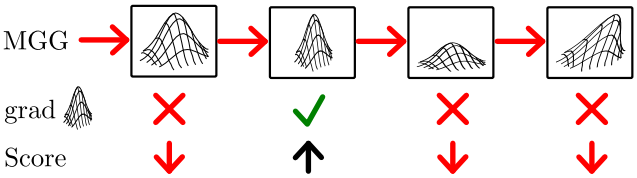

Figure 5: Update mechanism of a MGG, the Gaussian that matches the testing gradient value gets its values (frequency, mean and variance) updated, the others get their frequency decreased.

where $\mu^t_{k,x,y}$ is the kth mean in the position (x, y) of the distribution at frame t, $\sigma^t_{k,x,y}$ is the kth standard deviation in the position (x, y) of the distribution at frame t, $g_{x,y}$ is the gradient value in the position (x, y) from the current frame, and α is the learning rate from Eq. (2). Moreover, the frequency score of the matched Gaussian is reset to the initial value; this is done to keep each Gaussian for the same amount of time in the model. If the Gaussian was not match (such as the distributions marked with an 'X' in Fig. 5) its score is decreased and a new one is created; once the score reaches zero the Gaussian is removed from the MGG.

In other words, to allow the method to forget gradients that are no more in a given position, the frequency of each Gaussian is updated; when the Gaussian is not matched the frequency is decreased, and if it is matched its frequency score is restored to a predefined number of frames to keep it in the model. This mechanism allows us to keep the Gaussians that are updated periodically, because the edges are changing position in the neighborhood. For our experiments, we set the frequency to 30 frames.

2.2. Object Detection

From the incoming frame, its edges are extracted using Canny edge detector [3]. The foreground is extracted as those edges that deviate from the SM and the MGG. In other words, the moving edges will be those that do not lie in a distribution, and those that lie in a distribution but do not find a match in the MGG. The last one represents new objects that are in the same position with background edges. If those are new objects, that stopped, the model will absorb them into the background.

To reduce the noisy edges produced by random changes in the environment, we use an inter frame approach to verify the detected edges, because flickering edges will not appear in consecutive frames. Therefore, we examine the projection of the neighborhood of the N-th frame edge in frames $N - 1$ and $N + 1$ for its presence. If the edge is there then it is kept, otherwise it is removed.

To extract the moving object bounding box, we use a density verification approach. The image is divided into blocks (we use a nine by nine block size), then the amount of pixels in each block is computed, and the blocks are marked as foreground if the density is above some threshold. Then these blocks are merged into super blocks, made of a three by three blocks neighborhood. The super blocks are thresholded again using a higher threshold. The continuous regions in the blocks are kept if they appear in the super blocks' regions. This operation is similar to the hysteresis operation for edges, leaving only connected regions between the two thresholds, and using the high threshold regions as markers. Finally, for each region the bounding box is computed. In the detected regions, there are only moving edges, we can be sure that the regions we are recovering are moving objects. Nevertheless, if some spurious edges escape the filtering process, a final inter frame approach is used to increase the detection accuracy.

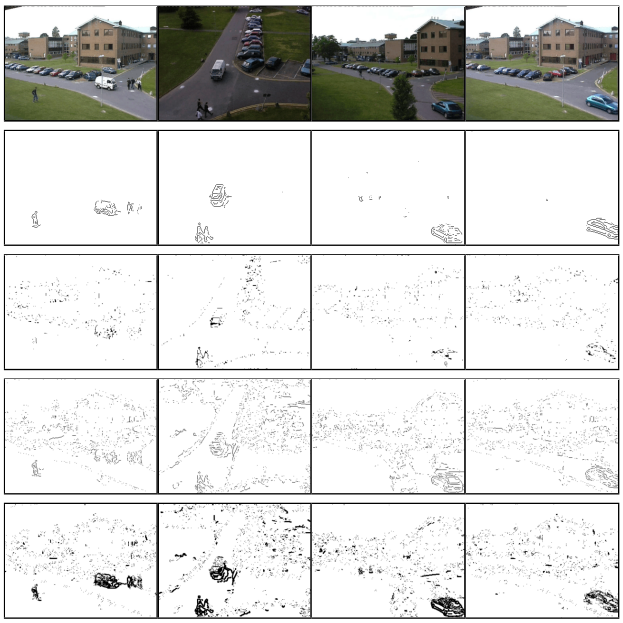

Figure 6: Results of the methods in four different data sets. From top to bottom: Original frame, proposed method, Dailey *et al*., Kim and Hwang, and Jain *et al*. From left to right: D1T1, D1T2, D2T1, and D2T2.

3. Results

We compared the proposed method in four data sets with dynamic environments. The data sets are from PETS 2001 [1], they show a parking lot scene from four different viewpoints and with many background variations, *e.g*., illumination, vegetation, reflects in the windows, clouds. The four data sets are named: D1T1, D1T2, D2T1, and D2T2. The ground truth was segmented by hand, and it is in the form of bounding boxes [2].

We compared against other three methods that are based on edges. Dailey *et al*. [4], Jain *et al*. [5], and Kim and Hwang [6]. To compare the algorithms we compute the precision, that is the percentage of detections that is foreground, recall, that is the percentage of foreground detected, and false positive rate, that is the percentage of background detected as foreground, defined by

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

$$FPR = \frac{FP}{FP + TN}, \tag{7}$$

where TP denotes the true positives, as the amount of pixels overlapping with the ground truth, FP denotes the false positives, as the amount of pixels that were detected but are not moving object, FN denotes the false negatives, as the amount of pixels that were detected as background but are moving object, and TN denotes the true negatives, as the amount of pixels that were detected as background that are background.

The results of the methods are shown in Fig. 6. The other methods present problems in the dynamic environment, being unable of adapt to the quick changes in the environment. Kim and Hwang's and Dailey *et al*.'s methods present problems due to the lack of a background model, their detection have many noisy edges and miss classification; this shows the need of a background model in dynamic environments. The mixture of Gaussians used by Jain *et al*., reduce the noisy edges; however, it does not cope with rapid changes as the proposed method does. The proposed method models the behavior of edges, incorporating the natural changes in the edges, and overcoming the limitations of other simpler models. Moreover, Dailey *et al*.'s and Jain *et al*.'s methods produce thick edges, which complicates the detection because edges merge together.

The high amount of noise generated by the other methods, difficult the object detection. Table 1 shows the quantitative measures of the methods; they are the average of detections in several frames. Despite the high recall of Dailey *et al*.'s, Kim and Hwang's and Jain *et al*.'s methods, their detection capabilities are not usable for detection purposes. The recall is high in those methods due to the noise, which is mistakenly classified as moving objects. However, in the other metrics the poor efficiency of the methods is revealed. The precision shows the percentage of the detection that is really moving; this metric show that the proposed method outperforms the other methods detection capabilities. The amount of noise and false detections is also revealed by the FPR metric; the other methods have a high value on this metric revealing the large percentage of false detection, while the proposed method have small values showing the robustness of the detections. In average, the proposed method out performs the other methods in dynamic environments.

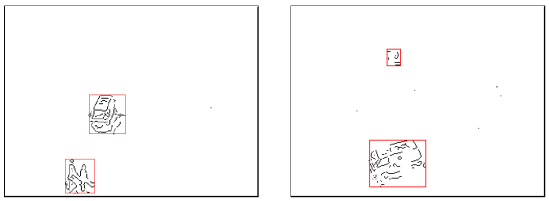

Figure 7: Bounding box generation for the results of the proposed method.

The bounding box extraction is shown in Fig. 7. The density approach used to extract the bounding boxes showed to be robust against noise. Moreover, it can handle objects of different sizes. The robustness of the bounding box extraction is a critical step in further processing steps, *e.g*., tracking.

4. Conclusion

In the present paper, we presented a novel object detection and background modeling algorithm based on edges. The background model consists of two parts, a distribution map that models the behavior of the edges, *i.e*., shape and position variations, and a set of mixture of Gaussians of gradients that increases the robustness of the method, and solves a problem ignored by previous methods, the edge overlapping problem. The edge distributions allow us to do adaptive thresholding, maintaining the shape of the edges, and adjusting the incoming edges to the model better than fixed-threshold methods. The MGG size is maintained dynamically, and it increases and decreases to adjust to the scene's requirements. Furthermore, the incorporation of the SM and the MGG enable the model creation in real time and in the presence of moving objects; avoiding the constraint of previous methods of have an ideal sequence to initiate the model.

This work shows the robustness of edges as an alternative to pixel based methods. The use of edges can increase the accuracy of algorithms in object detection and tracking, without an overhead in the computations, and obtain better results than current edge based methods. Moreover, additional information, *e.g*., curvature or color, can be added to the descriptor of edges to make them more robust, and increase their description capabilities of the scene.

References

[1] Performance evaluation of tracking and surveillance 2001. `ftp://ftp.pets.rdg.ac.uk/pub/PETS2001/`, Nov 2010. 4

Table 1: Quantitative measures for the evaluated methods, in four different sequences.

Methods	Recall				Precision				FPR			
	D1T1	D1T2	D2T1	D2T2	D1T1	D1T2	D2T1	D2T2	D1T1	D1T2	D2T1	D2T2
Proposed	0.7612	0.8220	0.6499	0.7997	0.7293	0.8167	0.6938	0.7004	0.0066	0.0070	0.0072	0.0063
Dailey *et al*.	0.9721	0.9284	0.9234	0.9399	0.0294	0.0504	0.0240	0.0225	0.7495	0.6679	0.7013	0.7187
Kim and Hwang	0.9980	0.9967	0.9891	0.9910	0.0258	0.0379	0.0210	0.0195	0.8880	0.9858	0.8888	0.8882
Jain *et al*.	0.9990	0.9975	0.9946	0.9956	0.0254	0.0378	0.0209	0.0191	0.9005	0.9908	0.8927	0.9089

[2] L. M. Brown, A. W. Senior, Y. li Tian, J. Connell, A. Hampapur, C. fe Shu, H. Merkl, and M. Lu. Performance evaluation of surveillance systems under varying conditions. In *In: Proceedings of IEEE PETS Workshop*, pages 1–8, 2005. 4

[3] J. F. Canny. A computational approach to edge detection. pages 184–203, 1987. 2, 4

[4] D. J. Dailey, F. W. Cathey, and S. Pumrin. An algorithm to estimate mean traffic speed using uncalibrated cameras. *IEEE Transactions on Intelligent Transportation Systems*, 1:98–107, 2000. 1, 5

[5] V. Jain, B. Kimia, and J. Mundy. Background modeling based on subpixel edges. In *Image Processing, 2007. ICIP 2007. IEEE International Conference on*, volume 6, pages VI –321–VI –324, sept. 2007. 1, 5

[6] C. Kim and J. Hwang. Fast and automatic video object segmentation and tracking for content-based applications, 2002. 1, 5

[7] M. Piccardi. Background subtraction techniques: a review. In *Systems, Man and Cybernetics, 2004 IEEE International Conference on*, volume 4, pages 3099–3104 vol.4, 10-13 2004. 1

[8] C. Stauffer and W. Grimson. Learning patterns of activity using real-time tracking. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(8):747–757, aug 2000. 1

[9] Y.-L. Tian, M. Lu, and A. Hampapur. Robust and efficient foreground analysis for real-time video surveillance. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 1182–1187 vol. 1, 20-25 2005. 1

[10] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers. Wallflower: Principles and practice of background maintenance. *Seventh International Conference on Computer Vision*, 1:255+, 1999. 1

[11] B. Zhong, H. Yao, S. Shan, X. Chen, and W. Gao. Hierarchical background subtraction using local pixel clustering. In *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, pages 1–4, 8-11 2008. 1