Scene Modeling using Edge Segment Distributions

Jaemyun Kim, Adin Ramirez Rivera, Minsun Park, and Oksam Chae
Computer Engineering, Kyung Hee University, Giheung-gu, Yongin-si, Gyeonggi-do, South Korea
{sense21c, adin, romana2ms, oschae}@khu.ac.kr

Abstract— *Most background modeling methods fail in the presence of dynamic backgrounds, because the model cannot handle sudden changes. A solution to this problem is the use of intensity-robust features: such as edges. In this paper, we propose a robust statistical edge-segment-based method for background modeling for non-ideal sequences. The proposed method learns the structure of the scene using the edge-segments' behavior. Moreover, the use of segments gives us local information of the scene, which help us predict the objects and background precisely. Additionally, the proposed method uses adaptive thresholding to detect the moving objects. Whereby, this approach increases the accuracy over previous methods, which use fixed thresholds. Experiments show that the proposed method produces reliable results in dynamic backgrounds, where other approaches fail to detect the moving objects.*

Keywords: Background modeling, motion detection, object detection, edge-segments

1. Introduction

The background model used in the moving object detection, using a fixed camera, plays a critical role in determining the performance of the moving object detection, as the background is subtracted from each frame. For a robust moving object detection, the background model should be able to absorb the shape and illumination variation of the background, and overcome the presence of moving objects and noise. Several features have been used to model the background and to detect the foreground [1]–[4]. However, we classify the background modeling methods, according to the feature used, into two groups: *pixel-based* and *edge-based* methods.

The pixel-based methods model the background using the intensity of each pixel, and several techniques [5]–[10] exist to create a model of each pixel. They produce, however, ghosts in the background model (*i.e.*, moving objects appear in the background model) compromising the method detection capabilities, which cause false detected foreground. Although statistical techniques have been used to overcome the ghost effect, the methods are susceptible to sudden illumination variations. Traditional pixel-based methods present two problems: multi-modal distributions in dynamic environments, and sensitivity to illumination changes and noise.

On the other hand, edge-based methods rely on edges, a feature that is less sensitive to intensity changes, solving one limitation of the pixel-based methods. Edges, however, have position and shape changes. Nevertheless, the use of edges allows these methods to use more robust and expressive models, because they work with fewer pixels. Moreover, most methods [11]–[17] use the edges pixel-wise, which misses the shape and neighborhood information of the edge, and such approach creates other problems (*e.g.*, scatter edges and false positives). A solution for these problems is treating each edge as segment by considering all pixels together rather than individually. For instance, Jain *et al*. [18] proposed a method that models the background based on a sub-pixel edge map, representing the position and orientation of the edge using a mixture of Gaussians model. The objects are extracted when no match is found with the background model. The method, however, is prone to incorporate ghosts in the background model due to moving objects, producing false detections. Also, Hossain *et al*. [19] made use of an edge-segment-based approach and flexible matching to detect moving objects. A set of list of edges holds the changes in the scene, and the object is detected from the elimination of edges on those lists. The edges, however, are matched using a fixed threshold, treating all the search regions of the edges in the same way, when they present different variations due to their dynamic nature. Then, a watershed-based iterative algorithm is employed to segment the moving object region from the extracted moving edges. Nevertheless, they require ideal frames to create the background model.

Most edge-based background modeling methods assume that a sequence free of moving objects is available to create the model. This assumption is not true in real scenarios, in which it is impossible to have a clear background—*e.g.*, a parking lot, a crowded street, or freeways. Therefore, a method that can build a background model in the presence of moving objects is needed. Moreover, they do not overcome the ghost effect due to moving objects. Additionally, these methods are sensitive to edges' shape and position changes. Despite that edges have different changes and that should be treated individually, edge-based methods use a fixed matching scheme to verify all the edges.

Consequently, we present a novel statistical background modeling method based on edge-segments, in which we model the structure of the scene by learning the edges' behavior. Thus, this behavior is encoded as statistical distribu-

(a) (b) (c)

Fig. 1: Edges have changes in shape and position. (a) Shows a sample scene with waving trees, and (b) its edges. (c) Shows the accumulation of 100 frames reveals the variation of the edges. The building's edges have small variations, while the trees' edges present high variations.

tions of edge-segments among frames. Thereby, the proposed method can work without an ideal sequence, overcoming edges shape and position changes, and background intensity changes. Additionally, we introduce an adaptive threshold mechanism that exploits the behavior of different edges, which reduces the false detections.

2. Background Modeling using Edges

Recent research is focusing on the use of edges as consistent features, due to their robustness against illumination and noise. Nevertheless, edges have some problems: shape and position changes. Figure 1 illustrates these problems, as it shows a sample scene with waving trees and its corresponding edges, and illustrates how much an edge could move from one frame to another (Fig. 1c). Moreover, each edge has different variations (the edges of the buildings exhibit little variation in comparison to the edges of the trees) and even edges from similar objects reveal diverse behavior (the tree in the middle manifest more motion than the trees in the corner of the picture). Therefore, a simple pixel by pixel matching approach would fail most of the time. Another problem arises when the moving object edges are close to the background edges, case in which the moving edges are confused with background. Additionally, recent methods verify all the edges based on fixed thresholds, which encumbers the task of distinguishing them from each other.

On the other hand, our method copes with these problems thanks to the edge-segment statistical approach. To represent the objects and the background, we use edge-segments that are less sensitive to illumination changes. Consequently, to avoid edge shape and position changes, we learn the behavior of these edge-segments through a set of kernel-based statistical distributions. Note that the use of segments introduces more information, which makes the score generation more reliable than a edge pixel by pixel verification. And so, we apply an adaptive threshold to each segment distribution, which sets a different threshold automatically according to each edge's observed variations. Moreover, the edge's variations allow us to determine the position where we

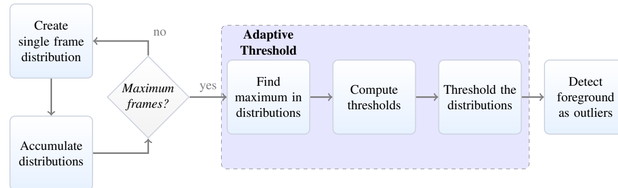

Fig. 2: A flow diagram of the proposed method.

will split foreground edges that merge with the background. Thus, we increase the detection accuracy of our method.

3. Proposed Method

Our statistical model attempts to predict the edge's behavior, *i.e.*, its shape and position changes, and encode it into a set of parameters. Therefore, when a new edge comes to the scene, we test it against the previous observed edge's behavior, and determine whether it fits the previous edges or it is a new one. Moreover, we use an adaptive comparison framework for the edges (*i.e.*, the threshold for the matching score and the search window) that increases the accuracy of the detection. Additionally, the statistical model allows us to suppress the contribution of the moving objects to the background model, leaving only the background edges contribution in the model. In summary, the background modeling method is divided in four parts (as shown in Fig. 2): (1) First we create the frame statistical model. It is a kernel-density distribution from the edge maps. (2) Then, the frame statistical distributions are accumulated using temporal information. (3) And the accumulation is adptively thresholded, allowing us to use non-ideal frames to learn the background. (4) Finally, the moving edges are extracted as outliers from the statistical model. Furthermore, we present an abstraction of the method in Fig. 3.

3.1 Statistical Modeling

To estimate the edges' behavior, first, we extract each image's edges using Canny edge detector [20], and represent the extracted edges set from frame t with a binary edge map, $\mathbb{E}_b^t$. Then, we create the *statistical map*, SM, that is the set of all the distributions, from a set of frames through

$$\mathrm{SM} = \sum_{t=t_0}^{t_f} \sum_{e \in \mathbb{E}_b^t} K(e), \tag{1}$$

where the range of frames is from the initial frame t_0 to the final frame t_f, $\mathbb{E}_b^t$ is the binary edge map at frame t, e is an edge from the edge map $\mathbb{E}_b^t$, and $K(\cdot)$ is a kernel function estimator.

We create a weighted distribution in each edge by placing a kernel function estimator, $K(\cdot)$, over each edge pixel. The kernel is a function chosen to model the statistical behavior of the edges, and it helps to compensate the small number of data—as the edges are sparse and change too much. Furthermore, the kernel width will give the minimum

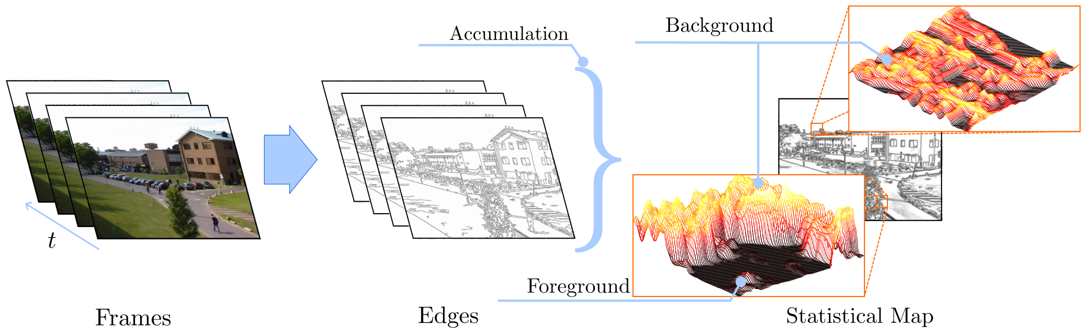

Fig. 3: An abstraction of the proposed method.

variation when the edges present no variation at all. Hence, in our experiments, we chose a Gaussian kernel to model the edge's behavior, as it should be in a long sequence, and it is given by

$$K(e) = \frac{1}{\sqrt{2\pi}h} \sum_{p \in \mathcal{N}(e)} \exp\left\{-\frac{(p-e)^2}{2h^2}\right\} \tag{2}$$

where e is a pixel's position from the edge map, p is the pixel's position that belongs to neighborhood $\mathcal{N}(e)$ of the edge e, and h is the width of the kernel (Note that the width of the neighborhood is the same as the width of the kernel).

3.2 Adaptive Threshold

Note that the distributions are different from two points of view: *accumulation* and *motion*. The accumulation of edges, among frames, reveals their variation and frequency (rate of the edge's occurrence in consecutive frames). Moreover, the frequency indicates which distributions represent background and which ones foreground (as shown in Fig. 3). Thereby, the background and the foreground have a distinctive frequency. For instance, the moving objects, that appear and disappear from the scene, create small peaks in the distribution; while the background edges have a high distribution. Consequently, we can remove the spurious distributions based on the edge's frequency. On the other hand, the different motion in the edges creates wider or narrower distributions, *e.g.*, edges with a lot of movement create spread distributions, while edges with little movement create sharp distributions. The creation of *ad hoc* distributions for each edge allows us to define accurate search regions for the edge matching process, and define adaptive thresholds for each edge according to its characteristics. We threshold the distributions from these two points of view: by using the accumulation to remove foreground, and by using the motion (through the standard deviation of each distribution) to improve the accuracy of the detection.

3.2.1 Accumulation Threshold

To remove the distributions created by the moving objects, we assume that the moving objects will have an average speed v in pixels per frames. Moreover, we use the inverse of the speed, $1/v$, that gives us the number of frames that an object stays in the same place. Hence, we proposed a threshold, to remove such distributions, defined by

$$T = \frac{\max\left(K(\cdot)\right)}{v}, \tag{3}$$

where $\max\left(K(\cdot)\right)$ is the maximum value from the kernel function, and v is the moving objects minimum average speed. In our experiments, we assumed that the objects will present a minimum average speed of $v = 2/N$, where N is the number of frames used for learning (we used $N = 200$), that is, an object will not be stopped more than half of the total frames used for learning the model.

3.2.2 Motion Threshold

To threshold the distribution according to their motion we need to compute the cutting point that represents certain percentage of the distribution (given by $k\sigma$). First, we thin each distribution using Multi-Directional Non-Maximum Suppression [21] (which is the application of the non-maxima suppression algorithm at several directions and the combination of the results) to extract the center (maximum peak of) each distribution. Consequently, we can compute several moments of the distribution, and approximate the cutting point of the distribution through slices of the distribution that are orthogonal to the center of the distribution. First, we define the ratio of the probability of two given points by

$$\frac{G_i}{G_j} = \exp\left(\frac{x_j^2 - x_i^2}{2\sigma^2}\right), \tag{4}$$

where x_i and x_j are points in the distribution with probability G_i and G_j, respectively, and σ is the standard deviation of the distribution. This relation allows us to use the mean of the distribution (G_0 at position $x_0 = 0$), to define any point position (x_i) as the a function of the ratio of its probabilities, by

$$x_i = \sqrt{2\sigma^2 \ln(\frac{G_0}{G_i})}. \tag{5}$$

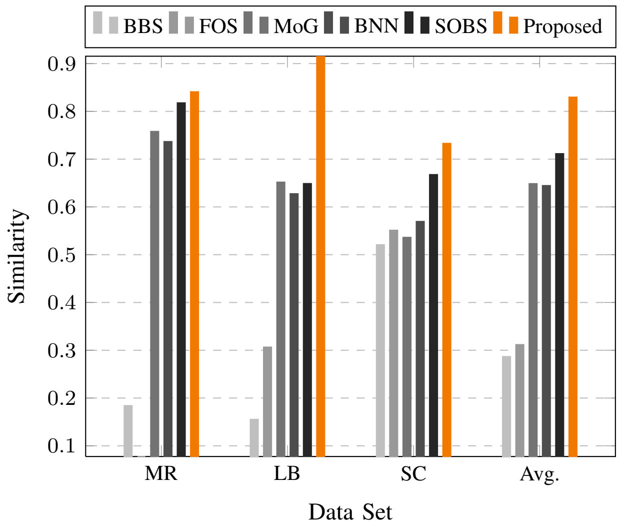

Fig. 4: Similarity measure on I2R database.

Thus, we can define the quantization step q between to points of the distribution, by

$$q = x_i - x_j \tag{6}$$
$$= \sqrt{2\sigma^2}\left[\sqrt{\ln\left(\frac{G_0}{G_i}\right)} - \sqrt{\ln\left(\frac{G_0}{G_j}\right)}\right]. \tag{7}$$

Moreover, we define the cutting point for a $k\sigma$ percentage of the distribution (in our experiments we use about 95% of the distribution by using $k = 2$) by

$$p_{\text{cut}} = \frac{k\sigma}{q} \tag{8}$$
$$= \frac{k}{\sqrt{2}\left[\sqrt{\ln\left(\frac{G_0}{G_i}\right)} - \sqrt{\ln\left(\frac{G_0}{G_j}\right)}\right]}. \tag{9}$$

Thereby, this point is the pixel position from the mean of the distribution that defines where to prune the distribution. Furthermore, we use several points from each distribution (as samples from the orthogonal slice of each mean point) to refine the approximation of this cutting point (by averaging the resultant cutting points). Consequently, we create a map with regions that represents the background.

3.3 Foreground Detection

We use the resultant distributions of the adaptive threshold operation as background model to detect the moving objects in the scene. In order to detect the moving objects, first, we obtain the edges in the incoming frames by using a Canny edge detector. Then, we compare the moving edges with the background model. Consequently, those edges that do not lie within a background distribution are consider moving objects.

Fig. 5: Segmentation of I2R data set [22] sequences. (Rows) Different sequences. (Left Column) A sample frame of each sequence. (Middle Column) Ground truth of each sequence. (Right Column) Proposed method segmentation result.

4. Experiments and Results

We test our method on different sequences [22] that present dynamic backgrounds and represent situations for video surveillance systems. For a detailed description of the sequences refer to [10]. The ground truth was segmented by hand. The sequences are the meeting room sequence (MR), the lobby sequence (LB), and the shopping center sequence (SC).

4.1 Evaluation

The quantitative metrics we use are *Precision*, *Recall*, and *Similarity*, defined by

$$\mathit{Recall} = \frac{tp}{tp + fn} \tag{10}$$
$$\mathit{Precision} = \frac{tp}{tp + fp} \tag{11}$$
$$\mathit{Similarity} = \frac{tp}{tp + fn + fp} \tag{12}$$

where tp is the total of true positives, fn is the total of false negatives, and fp is the total of false positives. Each indicates the number of items detected or not in the ground truth or the foreground detection accordingly to each method; in the region based methods it indicates the percentage of regions matched, the same is true for pixel and edge based methods. *Recall* gives the percentage of true positives detected. *Precision* gives the percentage of detected items that are true positives. And the *Similarity* is the weighted harmonic mean of Precision and Recall. The use of percentages let us to compare the methods' outputs, using different detection outputs, such as pixels and edges.

Table 1: Evaluation of the proposed method on I2R database.

Sequence	Precision	Recall
MR	0.899	0.929
LB	0.923	0.992
SC	0.759	0.956
Average	0.860	0.959

4.2 Results

We compared the proposed method in the I2R data set [22] against other five methods referred as *BBS*, *MoG*, *BNN* and *SBOS*, reported in [23], and *FOS*. Li *et al*. [10] use a Bayesian framework for background subtraction (BBS) that incorporates spectral, spatial and temporal features to classify background and foreground. The Mixture of Gaussians (MoG) [24] uses multiple weighted Gaussian distributions as a background model and an online update method for the parameters. The background neural network (BNN) [23] is a mixture of a probabilistic neural network and a winner-take-all neural network with temporal adaptation weights based on a Bayesian formulation. The *SOBS* [25] is a self-organizing approach through neural networks that present similarities with codebook methods. Kim and Hwang [14] present a fast object segmentation (FOS) algorithm that uses edges to extract the moving objects in a video sequence. Their method relies on a simple background model with frame difference approach to extract the moving objects. Figure 4 shows the similarity metric of the methods on the different sequences. The proposed method has good performance on the indoor sequences (MR and LB), with an average improvement over the second best of 25%. Moreover, it performs well in the large indoor sequence SC, which has challenging illumination reflections. Moreover, in average the proposed method outperforms the other methods.

The proposed method is capable of segmenting the moving objects in the presence of dynamic background in most of the test sequences. Representative frames and corresponding foreground segmentation results are shown in Fig. 5. This figure shows that the proposed method can cope with complex backgrounds. The shadow and illumination robustness is shown in the sequence SC. The proposed method proved to be robust against dynamic illumination environments. The precision and recall values for these sequences are shown in Table 1. This table shows the high recall values of our method. However, the precision of the method can be improved by including more information into the model.

5. Conclusion

We presented a statistical edge-segment-based method to model background and detect moving objects in dynamic environments. The proposed method builds statistical distributions for each edge-segment, using each edge-segment unique information to compare other edges resulting in a robust adaptive verification process. Moreover, thanks to these features we overcome the most common edge problems, such as shape and position changes. Furthermore, these mechanisms can be incorporated in other edge-based methods to extend their functionality and make them robust in dynamic environments. The proposed statistical map can be used to split foreground edges that merge with the background, increasing the detection accuracy. Additionally, the proposed method explores the edge domain, which has not been researched as much as the pixel domain, for object detection. We found promising results that can be used in several applications, including surveillance in dynamic backgrounds and content-based video encoding.

References

[1] T. Bouwmans, F. El Baf, and B. Vachon, "Statistical background modeling for foreground detection: A survey," in *HPRCV09*, 2009, pp. IV: 181–199.

[2] M. Piccardi, "Background subtraction techniques: A review," in *Systems, Man and Cybernetics, 2004 IEEE International Conference on*, vol. 4, 10-13 2004, pp. 3099 – 3104 vol.4.

[3] R. J. Radke, S. Andra, O. Al-kofahi, B. Roysam, S. Member, and S. Member, "Image change detection algorithms: A systematic survey," *IEEE Trans. Image Process.*, vol. 14, pp. 294–307, 2005.

[4] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Comput. Surv.*, vol. 38, no. 4, p. 13, 2006. [Online]. Available: http://portal.acm.org/citation.cfm?id=1177352.1177355&coll=ACM&dl=ACM&CFID=197115&CFTOKEN=59525587#

[5] O. Barnich and M. Van Droogenbroeck, "Vibe: A powerful random technique to estimate the background in video sequences," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, april 2009, pp. 945 –948.

[6] E. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *Journal of the ACM*, vol. 58, no. 3, May 2011.

[7] J. McHugh, J. Konrad, V. Saligrama, and P.-M. Jodoin, "Foreground-adaptive background subtraction," *IEEE Signal Process. Lett.*, vol. 16, no. 5, pp. 390–393, may 2009.

[8] D.-Y. Lee, J.-K. Ahn, and C.-S. Kim, "Fast background subtraction algorithm using two-level sampling and silhouette detection," in *Image Processing (ICIP), 2009 16th IEEE International Conference on*, Nov. 2009, pp. 3177 –3180.

[9] L. Li and M. K. H. Leung, "Integrating intensity and texture differences for robust change detection," *IEEE Trans. Image Process.*, vol. 11, no. 2, pp. 105–112, 2002.

[10] L. Li, W. Huang, I. Y. Gu, and Q. Tian, "Statistical modeling of complex backgrounds for foreground object detection," *IEEE Trans. Image Process.*, vol. 13, no. 11, pp. 1459–1472, 2004.

[11] D. J. Dailey, F. W. Cathey, and S. Pumrin, "An algorithm to estimate mean traffic speed using uncalibrated cameras," *IEEE Trans. Intell. Transp. Syst.*, vol. 1, pp. 98–107, 2000.

[12] G. Jiang, S. Wang, M. Yu, T. Choi, and Y. Kim, "New method of vision based vehicle detection and tracking in complicated background," in *TENCON 2004. 2004 IEEE Region 10 Conference*, vol. A, 21–24 2004, pp. 387 – 390 Vol. 1.

[13] J.-n. Chi, C. Zhang, H. Zhang, Y. Liu, and Y.-t. Yan, "Approach of moving objects detection in active video surveillance," in *Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on*, 15-18 2009, pp. 3130–3136.

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

[15] Q.-y. Yang and X.-y. Gao, "Tracking on motion of small target based on edge detection," in *Computer Science and Information Engineering, 2009 WRI World Congress on*, vol. 6, march 2009, pp. 619 –622.

[16] O. Yang, "Moving object detecting method in video," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 23, no. 1, pp. 18 –20, jan. 2008.

[17] L. Zhi-fang and Y. Zhisheng, "A real-time vision-based vehicle tracking and traffic surveillance," in *Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, 2007. SNPD 2007. Eighth ACIS International Conference on*, vol. 1, july 2007, pp. 174 –179.

[18] V. Jain, B. Kimia, and J. Mundy, "Background modeling based on subpixel edges," in *Image Processing, 2007. ICIP 2007. IEEE International Conference on*, vol. 6, sept. 2007, pp. VI –321–VI –324.

[19] M. J. Hossain, M. A. A. Dewan, and O. Chae, "Moving object detection for real time video surveillance: An edge based approach," *IEICE Transactions*, vol. 90-B, no. 12, pp. 3654–3664, 2007.

[20] J. F. Canny, "A computational approach to edge detection," pp. 184–203, 1987.

[21] C. Sun and P. Vallotton, "Fast linear feature detection using multiple directional non-maximum suppression," in *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, vol. 2, 0-0 2006, pp. 288 –291.

[22] L. Li, W. Huang, I. Y.-H. Gu, and Q. Tian, "I2R data set," http://perception.i2r.a-star.edu.sg/bk_model/bk_index.html, Nov. 2011.

[23] D. Culibrk, O. Marques, D. Socek, H. Kalva, and B. Furht, "Neural network approach to background modeling for video object segmentation," *Neural Networks, IEEE Transactions on*, vol. 18, no. 6, pp. 1614 –1627, Nov. 2007.

[24] C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking," in *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, vol. 2, 1999, p. 252 Vol. 2.

[25] L. Maddalena and A. Petrosino, "A self-organizing approach to background subtraction for visual surveillance applications," *Image Processing, IEEE Transactions on*, vol. 17, no. 7, pp. 1168 –1177, 2008.