REGION-BASED STATISTICAL BACKGROUND MODELING FOR FOREGROUND OBJECT SEGMENTATION

Kristof Op De Beeck[a], *Irene Yu-Hua Gu*[b], *Liyuan Li*[c], *Mats Viberg*[b], *Bart De Moor*[a]

[a]Dept. of Electrical Engineering, Katholic Univ. Leuven, Belgium
[b]Dept. of Signals and Systems, Chalmers Univ. of Technology, Sweden
[c]Institute for Infocomm Research, Singapore

ABSTRACT

This paper proposes a novel region-based scheme for dynamically modeling time-evolving statistics of video background, leading to an effective segmentation of foreground moving objects for a video surveillance system. In [1] statistical-based video surveillance systems employ a Bayes decision rule for classifying foreground and background changes in individual pixels. Although principal feature representations significantly reduce the size of tables of statistics, pixel-wise maintenance remains a challenge due to the computations and memory requirement. The proposed region-based scheme, which is an extension of the above method, replaces pixel-based statistics by region-based statistics through introducing dynamic background region (or pixel) merging and splitting. Simulations have been performed to several outdoor and indoor image sequences, and results have shown a significant reduction of memory requirements for tables of statistics while maintaining relatively good quality in foreground segmented video objects.

Index Terms – video surveillance, object tracking, Bayes classification, statistical background modeling.

1. INTRODUCTION

Foreground object detection and segmentation from a video is one of the essential tasks in many applications for example, video surveillance, object-based video coding, and multimedia. A simple way of extracting foreground objects from videos captured by a stationary camera is through background subtraction techniques [2, 3]. However, these simple methods do not work well if the background contains illumination variations and other dynamic changes. A range of methods have been proposed in previous studies, e.g., filters are used along the temporal direction for smoothing illumination variations [4]; characterizing the intensity of an image pixel by mixture of Gaussians [5, 6, 7] followed by updating Gaussian parameters to adapt to gradual background changes. [1] has proposed a statistical method by Bayesian classification of foreground and background changes and by maintaining statistics of background changes dynamically. The estimated pdfs of background pixels are obtained by using principal feature representations and tables of statistics. Subsequently, these tables are updated at different rates depending on whether background changes are due to slow illumination changes (static background) or movement in the background (dynamic changes), hence it is more robust to a variety of background changes. A main disadvantage in [1] is that each pixel requires three tables of statistics. When the image size is large, this not only leads to using large memory space, but also to a significant amount of computations in updating tables. Motivated by this, we improve the previous method by using a region-based scheme through introducing dynamic pixel/region grouping and region splitting, which takes into account the spatial correlations of image pixels.

2. SYSTEM DESCRIPTION

The proposed system, aimed at foreground object segmentation from complex background, consists of 4 basic processing blocks: *change detection, change classification, foreground segmentation, and region-based background maintenance*. In the change detection block both temporal changes and the changes to a background reference image are detected. In the change classification block, pixels with detected changes are classified as either dynamic or static, each is then further classified between the foreground and the background by using the Bayes rule. In the foreground segmentation block, connected pixels are formed into segments where small holes are filled afterwards. In the region-based background maintenance block, tables of statistics for background regions are updated which include joining some background pixels/regions with similar statistics into regions, or splitting some background regions when the statistics of pixel(s) in a region start to deviate.

3. STATISTICAL MODELING USING PRINCIPAL FEATURE REPRESENTATIONS

3.1. Feature Selection

Let $\mathbf{I}(\mathbf{s}, t)$ be an input image, $\mathbf{v} = \mathbf{v}(\mathbf{s}, t)$ be the pixel-related feature vector extracted from $\mathbf{I}(\mathbf{s}, t)$, $\mathbf{s} = (x, y)$ be the position of pixel and t be the time instant. Two types of fea-

ture vectors are used, one is associated with changes in *static background* and another in *dynamic background*. A change in static background is mainly caused by illumination variations such as change of indoor lighting or outdoor weather resulting differences to a pre-stored background reference image. A change in dynamic background is related to a temporal change in two consecutive images commonly caused by movement in the scene. For changes in static background, we set 2 components for the feature vector. They are (color) intensity and gradient values,

$$\mathbf{v}^s = [\mathbf{c} \;\; \mathbf{e}]^T, \;\; \text{where } \mathbf{c} = \mathbf{I}(\mathbf{s}, t), \; \mathbf{e} = \left[\frac{\partial \mathbf{I}(\mathbf{s}, t)}{\partial x} \;\; \frac{\partial \mathbf{I}(\mathbf{s}, t)}{\partial y} \right]$$

These two component vectors of $\mathbf{v}^s$ are assumed to be independent. For changes in dynamic background, we define the feature vector as the co-occurrence of intensities,

$$\mathbf{v}^d = [\mathbf{cc}]^T, \;\; \text{where } \mathbf{cc} = [\mathbf{I}(\mathbf{s}, t-1), \mathbf{I}(\mathbf{s}, t)]$$

3.2. Estimation of Probability Distributions of Features using Principal Feature Representations

For characterizing image statistics, the probability distributions of features associated with a region $r = \{\mathbf{s}\}$ (see Section 5 for pixel/region grouping). Each region contains connected pixel(s) with similar background. The pdf's in each region are estimated using histograms and then truncated to a few principal feature components. We refer to this process as *principal feature representation*. Let a training set of feature vector samples be denoted as $\{\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_K\}$, where $\mathbf{v} \in \{\mathbf{v}^s, \mathbf{v}^d\}$, $P_r(b)$, $P_r(\mathbf{v}_i)$ and $P_r(\mathbf{v}_i|b)$ be the prior and conditional probabilities. For each region $r = \{\mathbf{s}\}$, tables of statistics (i.e., histograms with small values truncated) are stored as the approximation of pdf's.

Let $P_r(\mathbf{v}_i|b), i = 1, \cdots, K$, be arranged according to the descending values of $P_r(\mathbf{v}_i)$. For given M_1 and M_2, $1.0 > M_1 > M_2 > 0.0$, there exists a small integer number $N(\mathbf{v})$ such that the probability satisfies,

$$\sum_{i=1}^{N(\mathbf{v})} P_r(\mathbf{v}_i|b) > M_1 \quad \text{and} \quad \sum_{i=1}^{N(\mathbf{v})} P_r(\mathbf{v}_i|f) < M_2 \qquad (1)$$

where b and f denote the background and foreground, L is the number of the quantization levels and n is the size of $\mathbf{v}$, and $N(\mathbf{v}) \ll L^n$. The small $N(\mathbf{v})$ is supported empirically that the effective spread of histograms for background regions is much narrower as compare with the entire support L^n. Determining $N(\mathbf{v})$ is dependent on the feature vector type, the quantization level L and $\delta_{\mathbf{v}}$ (see Section 4.2). A table of statistics is formed as follows:

$$T_r(t; \mathbf{v}) = \begin{cases} P_r^t(\mathbf{v}_i), & P_r^t(\mathbf{v}_i|b), \quad i = 1, \cdots, M(\mathbf{v}) \\ P_r^t(b) \end{cases} \qquad (2)$$

where $M(\mathbf{v}) > N(\mathbf{v})$ is set, $\mathbf{v}_i$ is the ith feature vector in the table, $P_r(\mathbf{v}_i)$ and $P_r(\mathbf{v}_i|b)$ are sorted out according to the descending order of $P_r(\mathbf{v}_i)$, and $\mathbf{v} \in \{\mathbf{v}^s, \mathbf{v}^d\}$. The $N(\mathbf{v})$ features in the table are defined as the *principal features* for a background region r. For feature type $\mathbf{v}^s$ two separate tables $T_r(t; \mathbf{c})$ and $T_r(t; \mathbf{e})$ are formed since the component vectors are assumed to be independent. It is shown [1] that for M_1=0.85, M_2=0.15, $\delta_{\mathbf{v}^s}$=0.005, $\delta_{\mathbf{v}^d}$=2, $N(\mathbf{v}^s) = 15$ and $N(\mathbf{v}^d) = 50$ are good approximations when the features are quantized to L_s=256 and L_d=32 levels, respectively.

4. BAYES CLASSIFICATION OF CHANGES

4.1. Detect Regions with Different Types of Changes

For a new input image $\mathbf{I}(\mathbf{s}, t)$, region-based change detection is applied. If changes in pixels are detected from the temporal differencing $|\mathbf{I}(\mathbf{s}, t) - \mathbf{I}(\mathbf{s}, t-1)|$ and the average change within a region exceeds a pre-specified threshold then it is specified as a dynamic change region where the feature type $\tilde{\mathbf{v}} = \tilde{\mathbf{v}}^d$ is selected. Otherwise, if changes are detected from the differencing pixel values from the image frame and background reference image $|\mathbf{I}(\mathbf{s}, t) - \mathbf{B}(\mathbf{s}, t)|$ and the average change in a region exceeds a threshold then it is specified as a static change region where the feature type is set as $\tilde{\mathbf{v}} = \tilde{\mathbf{v}}^s$.

4.2. Estimate Probabilities for Input Image Regions

After the type of changes are determined, the probabilities of an input image region (including single-pixel regions) are estimated by using the existing table of statistics,

$$P_r(\tilde{\mathbf{v}}) = \sum_{\mathbf{v}_j \in U(\tilde{\mathbf{v}})} P_r^t(\mathbf{v}_j), \;\; P_r(\tilde{\mathbf{v}}|b) = \sum_{\mathbf{v}_j \in U(\tilde{\mathbf{v}})} P_r^t(\mathbf{v}_j|b)$$

where $\tilde{\mathbf{v}} = \tilde{\mathbf{v}}(\mathbf{s})$ are feature vectors extracted from $r = \{\mathbf{s}\}$ in $\mathbf{I}(\mathbf{s}, t)$, $U(\tilde{\mathbf{v}}) = \{\mathbf{v}_j \in T_r(t) \mid d(\tilde{\mathbf{v}}, \mathbf{v}_j) \le \delta_{\mathbf{v}}; \, j \le N(\mathbf{v})\}$ is a subset of features from the table $T_r(t; \mathbf{v})$ if the distance $d(\tilde{\mathbf{v}}, \mathbf{v}_j) = 1 - \frac{2 < \tilde{\mathbf{v}}, \mathbf{v}_j >}{\|\tilde{\mathbf{v}}\|^2 + \|\mathbf{v}_j\|^2}$ is smaller than a pre-specified $\delta_{\mathbf{v}}$.

4.3. Bayes Classification of Background and Foreground

In the 2-class (foreground and background) case, the Bayes decision rule for classifying a background region is,

$$P_r(b|\mathbf{v}) > P_r(f|\mathbf{v}) \qquad (3)$$

where $\{\mathbf{v}(\mathbf{s}) | \; \mathbf{s} \in r\}$. Noting the posterior probability of a region r being the background b, or the foreground f for feature vectors is $P_r(b|\mathbf{v}) = \frac{P_r(\mathbf{v}|b)P_r(b)}{P_r(\mathbf{v})}$, $P_r(f|\mathbf{v}) = \frac{P_r(\mathbf{v}|f)P_r(f)}{P_r(\mathbf{v})}$ where $P_r(\mathbf{v})$ denotes the prior probability for feature type $\mathbf{v} \in \{\mathbf{v}^s, \mathbf{v}^d\}$. Since $P_r(\mathbf{v}) = P_r(\mathbf{v}|b)P_r(b) + P_r(\mathbf{v}|f)P_r(f)$ holds, the Bayes decision rule becomes,

$$2P_r(\mathbf{v}|b)P_r(b) > P_r(\mathbf{v}) \qquad (4)$$

For feature type $\mathbf{v}^s$, $2P_r(\mathbf{c}|b)P_r(\mathbf{e}|b)P_r(b) > P_r(\mathbf{c})P_r(\mathbf{e})$ is replaced to Eq.(4) due to independent components. Eq.(4) can be used to classify image regions once $P_r(b)$, $P_r(\mathbf{v})$ and $P_r(\mathbf{v}|b)$ are estimated.

5. REGION-BASED BACKGROUND MAINTENANCE

5.1. Dynamic Region Merging and Splitting

Background maintenance based on regions takes into account of spatial correlation of pixels and may significantly reduce the computations and memory requirements. Using background regions instead of individual pixels is justified since most background pixels are connected patches whose statistics are similar and are evolving with time in similar ways. However, due to the dynamic nature of videos, background regions changes (e.g., merge, split, re-group, or, shift). Therefore, a region-based background maintenance scheme must be able to dynamically cope with these situations.

Dynamic region merging: Noting merging pixels is a special case of merging regions that contain single pixels, pixel merging and region merging are handled by the dynamic merging method described below. Dynamic region merging is performed after updating tables of statistics at each t. The mean peak of intensity distribution, which is a good approximation to the local maximum of pdf (or, the *local mode*) is used to characterize a region. Since a table of statistics is an approximation of pdf, the mean peak estimate of region r is computed using a few elements in the table of statistics (sorted in descending order) as follows

$$\mu_{pk}(r) = \frac{\sum_{i=1}^{m} \mathbf{v}_i P_r(\mathbf{v}_i)}{\sum_{i=1}^{m} P_r(\mathbf{v}_i)} \tag{5}$$

where m is small integer number whose value is a trade-off between the computation and the accuracy of mean peak estimate (m=5 in our tests). If two *connected* regions r_m and r_n whose learned statistics have a similar local mode, $|\mu_{pk}(r_m) - \mu_{pk}(r_n)| < \delta_\mu$, then they will be merged into one. Since pixels in a large region are unlikely to be evolving in a same rate over a long run, a constraint of a maximum region size A is imposed. Once regions are merged, their tables of statistics are merged.

Dynamic region splitting: Intensities of individual pixels in a background region from a new input may deviate from the previously learned statistics when time evolves. For example, one part of the region may become a part of a foreground object, while another part remains in the background; or statistics in different parts of the region start to evolve in different ways. Dynamically splitting background regions is hence necessary to maintain the effectiveness of the scheme. Region split is performed before a new image frame at t is processed. To determine whether a region is split, the intra-region image intensity spread is computed for r in a newly input image:

$$S_r = max_{\mathbf{s}\in r}\{\mathbf{v}(\mathbf{s})\} - min_{\mathbf{s}\in r}\{\mathbf{v}(\mathbf{s})\} \tag{6}$$

If $S_r > T_{\mathbf{v}}$ is satisfied ($T_{\mathbf{v}}$ is an empirically determined threshold, $T_{\mathbf{v}}$=15 in our tests), then the region r is split in two possible ways: (a) split into a foreground and a background region. This is related to two clusters of intensities. (b) split into two or more background regions, each containing connected pixels and allowing different behavior when time evolves.

Assume r contains n_r pixels, and $\mathbf{I}(\mathbf{s}_i, t)$ are sorted out in descending order resulting $\mathbf{I}(\tilde{\mathbf{s}}_i, t)$, $i = 1, 2, \cdots n_r$, $\tilde{\mathbf{s}}_i \in r$. For case (a), pixels $\tilde{\mathbf{s}}_i$ are split from the region and moved to the foreground if they satisfy,

$$\mathbf{I}(\tilde{\mathbf{s}}_i, t) - \mathbf{I}(\tilde{\mathbf{s}}_{i+1}, t) > \delta_s, \;\; i = 1, \cdots, n_r - 1, \; \tilde{\mathbf{s}} \in r \tag{7}$$

δ_s is chosen to be larger than the average feature spread in background. If no pixels satisfy (7), then case (b) is assumed. Pixels whose intensities are far away from the mean intensity of the region are removed from the current region and a new region is formed. It is worth mentioning the constraint that all pixels within each split region are spatially connected.

5.2. Type-Dependent Learning and Updating

Since video scenes change with time, statistics for each region are time-varying. It is also unrealistic to assume that there exist training sequences in advance for each image sequence to be processed. Therefore, the statistics from the previous image frames should be absorbed during the dynamical learning. For robust to various changes, two types of table update strategies are adopted as in [1] however modified to regions.

For sudden changes due to switching foreground and background, $\sum_{i=1}^{N(\mathbf{v})} P_r(\mathbf{v}_i) - P_r(b) \sum_{i=1}^{N(\mathbf{v})} P_r(\mathbf{v}_i|b) > M_1$ is satisfied. Tables of statistics $T_r(t; \mathbf{v})$ are updated by using:

$$\begin{array}{l} P_r^{t+1}(b) = 1 - P_r^t(b), \quad P_r^{t+1}(\mathbf{v}_i) = P_r^t(\mathbf{v}_i) \\ P_r^{t+1}(\mathbf{v}_i|b) = \left(P_r^t(\mathbf{v}_i) - P_r^t(b) P_r^t(\mathbf{v}_i|b)\right) / P_r^{t+1}(b) \end{array} \tag{8}$$

for $i = 1, \cdots, N(\mathbf{v})$, and the learning rate is set as $\alpha > 1 - (1 - M_1)^{1/N}$ where N is the number of frames required to learn the new background appearance (e.g. $\alpha > 0.00473$ implies the designed system will respond to a sudden background change in 20 seconds for $M_1 = 85\%$ and video frame rate 20fps). After updating, the contents in $T_r(t+1; \mathbf{v})$ are re-sorted according to the descending order of $P_r^{t+1}(\mathbf{v}_i)$.

For the remaining regions containing static or dynamic background changes, tables $T_r(t; \mathbf{v})$ are updated by using:

$$\begin{array}{l} P_r^{t+1}(b) = (1-\alpha) P_r^t(b) + \alpha L_b^t, \\ P_r^{t+1}(\mathbf{v}_i) = (1-\alpha) P_r^t(\mathbf{v}_i) + \alpha L_{\mathbf{v}_i}^t \\ P_r^{t+1}(\mathbf{v}_i|b) = (1-\alpha) P_r^t(\mathbf{v}_i|b) + \alpha L_b^t L_{\mathbf{v}_i}^t \end{array} \tag{9}$$

where $\mathbf{v}_i$ is chosen according to the feature type, the learning rate α is a small positive number, $i = 1, \cdots, M(\mathbf{v})$, L_b^t=1 if r is classified as background otherwise L_b^t=0, and $L_{\mathbf{v}_i}^t = 1$ if $\tilde{\mathbf{v}}$ matches $\mathbf{v}_i$ otherwise $L_{\mathbf{v}_i}^t = 0$. Further, if $L_{\mathbf{v}_i}^t = 0$, the M-th component in the table is replaced by,

$$P_r^{t+1}(\mathbf{v}_M) = \alpha, \quad P_r^{t+1}(\mathbf{v}_M|b) = \alpha, \quad \mathbf{v}_M = \mathbf{v} \tag{10}$$

In addition to table updating, *updating background reference image region* is performed by,

$$\mathbf{B}(\mathbf{s}, t+1) = \begin{cases} \mathbf{I}(\mathbf{s}, t) & \text{for sudden changes} \\ (1-\beta)\mathbf{B}(\mathbf{s}, t) + \beta \mathbf{I}(\mathbf{s}, t) & \text{other changes} \end{cases}$$

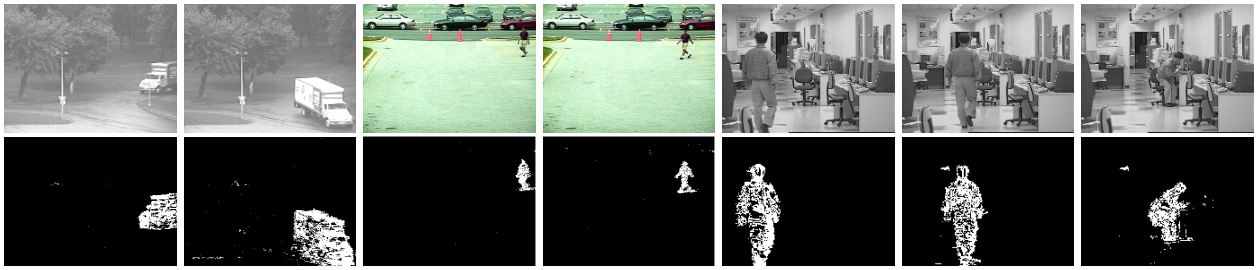

Fig. 1. Results obtained from the proposed method. Row 1-2: original image frame and segmented foreground objects (before post-processing of filling small holes). Columns 1-2: from outdoor video 'rain'; Columns 3-4: from outdoor video 'car parking'; Columns 5-7: from indoor video 'laboratory room'.

where $\mathbf{s} \in r$, and β controls the updating speed.

6. SEGMENTATION OF FOREGROUND OBJECTS

For detected pixels classified as foreground changes, segmentation of foreground objects is then applied followed by post-processing that fills small holes within segments, e.g. by morphological operators (this implies shifts some background pixels to the foreground) and merges small segments to a large neighboring segment.

7. SIMULATIONS AND RESULTS

Preliminary simulations have been conducted for several outdoor and indoor image sequences with some promising results. Fig.2 includes some statistics on the distribution of different sized regions as well as the average number of pixels per region for different image frames. The statistics show that a region contains an average of 5 pixels for the outdoor image sequence 'rain', hence in overall saved approximately 4/5 of memory used for the table of statistics. Since usually about 50% pixels (satisfying $F_{bd}(\mathbf{s},t) = 0$ and $F_{td}(\mathbf{s},t) = 0$) are removed during the change detection step, the required memory unit (Bytes) for a *color* image sequence is approximately equal to $(1/5 * 0.5 * (\#$ pixels in an image$) * (20 * 11 + 20 * 12 + 60 * 14))$, (where $M(\mathbf{v}^d)$=60, $M(\mathbf{c})$=20, $M(\mathbf{e})$=20 were used, $\mathbf{v}^s = \{\mathbf{c}, \mathbf{e}\}$, see (2), unsigned-char was used for color components and integer for probabilities). For example, for a color QCIF image sequence (image size 176*144), the required memory is about 3.3MB. Fig.1 includes several image frames from segmented outdoor and indoor videos, containing the segmented foreground results from the proposed region-based scheme. The parameters in the program were set to be M_1 =0.85, M_2=0.15, β=0.7, α=0.005, $\delta_{\mathbf{v}^d}$=2, $\delta_{\mathbf{v}^s}$=0.005, the table sizes were 15 and 50 for feature types $\mathbf{v}^s$ and $\mathbf{v}^d$, respectively. The segmented foreground images (before the post-processing of filling small holes) have shown that the proposed method works well however with some degradation as compared with the pixel-based method. Fine tuning of the parameters is required for obtaining a good tradeoff between the computations and region sizes.

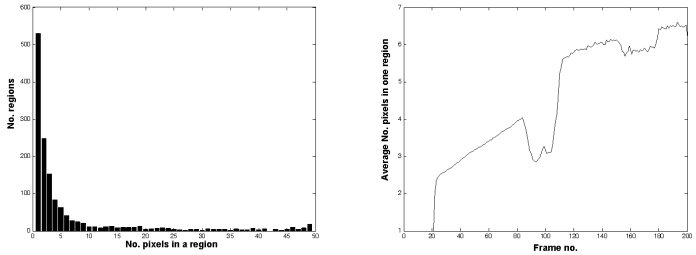

Fig. 2. Resulting statistics for 'rain' image sequence (statistics were computed from image frames $\mathbf{I}(x,y,t)$ in a small region $x \in [120, 200]$, $y \in [160, 260]$). Left: the total number of regions which contains the number of pixels indicated in the x axis; Right: the average region size versus image frames (merging regions starts from 20th frame).

8. CONCLUSION

The proposed region-based scheme, taking into account of the spatial correlation of pixels, is shown to be promising in dynamically modeling time-evolving statistics of video background and in effective segmenting foreground moving objects. The method has led to a significant reduction in the memory requirement and computation of tables of statistics at the price of some quality degradation in foreground object segmentation.

9. REFERENCES

[1] L. Li, W. Huang, I. Y.H. Gu, Q. Tian, "Statistical Modeling of Complex Backgrounds for Foreground Object Detection", *IEEE Trans. Image Processing*, vol.13, no.11, pp.1459-1472, 2004.

[2] E. Durucan and T. Ebrahimi, "Change Detection and Background Extraction by Linear Algebra," *Proceedings of the IEEE*, vol. 89, no. 10, pp. 1368-1381, 2001.

[3] L. Li, M.K. Leung, "Integrating Intensity and Texture Differences for Robust Change Detection," IEEE Trans.Image Processing, Vol.11, pp.105-112, 2002.

[4] D.Koller, J.Weber, T.Huang, J.Malik, G.Ogasawara, B.Rao, S.Russel, "Toward Robust Automatic Traffic Scene Analysis in Real-Time", *Proc. Int'l Conf. Pattern Recognition*, pp.126-131, 1994.

[5] C. Stauffer and W. Grimson, Learning Patterns of Activity Using Real-Time Tracking, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 22, no. 8, pp. 747-757, 2000.

[6] M. Harville, "A framework for high-level feedback to adaptive, per-pixel, mixture-of-gaussian background models", *Proc. European Conf. Computer Vision*, pp.543-560, 2002.

[7] D.S.Lee, "Effective Gaussian mixture learning for video background subtraction", PAMI, vol.27, no 5, pp.827-832, May 2005.

[8] I. Haritaoglu, D. Harwood, and L. Davis, "W^4: Real-Time Surveillance of People and Their Activities", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 809-830, 2000.