BACKGROUND SUBTRACTION WITH ADAPTIVE SPATIO-TEMPORAL NEIGHBORHOOD ANALYSIS

Marco Cristani, Vittorio Murino
Computer Science Dep., Università degli Studi of Verona, Strada Le Grazie 15, Verona, Italy
cristanm@sci.univr.it, vittorio.murino@univr.it

Keywords: Surveillance, Motion analysis, Robustness.

Abstract: In the literature, visual surveillance methods based on joint pixel and region analysis for background subtraction are proven to be effective in discovering foreground objects in cluttered scenes. Typically, per-pixel foreground detection is contextualized in a local neighborhood region in order to limit false alarms. However, such methods have an heavy computational cost, depending on the size of the surrounding region considered for each pixel. In this paper, we propose an original and efficient joint pixel-region analysis technique able to automatically select the sampling rate with which pixels in different areas are checked out, while adapting the size of the neighborhood region considered. The algorithm has been validated on standard videos with benchmark tests, proving the goodness of the approach, especially in terms of quality of the detection with respect to the frame rate achieved.

1 Introduction

Background subtraction is a fundamental step in automated video surveillance. It aims at classifying pixel values as *background* (BG), i.e., the expected part of the monitored scene, and the *foreground* (FG), i.e., the interesting visual information (e.g., moving objects). It is widely accepted that BG subtraction cannot be adequately performed by per-pixel methods, i.e., considering every temporal pixel evolution as an independent process. Instead, per-pixel methods augmented with per-region strategies (also called hybrid methods, see Section 2) better behave, deciding the class of a pixel value by inspecting the related neighborhood. Using hybrid schemes, several BG subtraction issues can be effectively faced [1]; anyway, the price to pay in hybrid systems is undoubtedly an increase of the computational load required.

In this paper, we propose a hybrid BG subtraction scheme which gives two contributes to the related state of the art: first, it is accurate, i.e., the number of false alarms is low: this is due to a dynamic definition of the neighborhood zone around each pixel which permits to capture aperiodic chromatic oscillations of scene components, such as boats in a harbour scenario or moving tree branches. Second, our method is fast, outperforming the time performances of the most-known BG subtraction algorithms. In practice, zones where the background is static with the same visual aspect are seldom observed. Vice versa, zones where the background varies or where foreground is visible are examined more often. We call our method *adaptive spatio-temporal neighborhood analysis* (ASTNA). Experiments carried out on standard and ad-hoc benchmark data show the goodness of ASTNA.

The rest of the paper is organized as follows. Section 2 reviews briefly the state of the art of the background subtraction methods; details of the proposed strategy are reported in Section 3; in Section 4, experiments on real data and critical observations are reported, and, finally, in Section 5 conclusions are drawn and future perspectives envisaged.

2 Related literature

The actual BG subtraction literature is large and multifaceted; here we propose a taxonomy in which the BG subtraction methods are organized in i) per pixel, ii) per region, iii) per frame, and iv) hybrid methods.
The class of per-pixel approaches is formed by methods that perform BG/FG discrimination by considering each pixel signal as an independent process. One of the first BG modeling was proposed in the surveillance system Pfinder [2], where each pixel signal was modeled as a unimodal Gaussian distribution. In [3], the pixel evolution is modeled as a multimodal signal, described with a time-adaptive mixture of Gaussian components (TAPPMOG). In [4], the authors specified i) how to cope with color signals (the original version was proposed for gray values), proposing a normalization of the RGB space taken from [5], ii) how to avoid overfitting and underfitting (due to values of the variances too low or too high), proposing a thresholding operation, and iii) how to deal with sudden and global changes of the illumination, by tuning the learning rate parameter. For the latter, the idea was to increase the learning rate when the foreground increases from one frame to another more than 70%: in this way the BG model can faster evolve and produce less false alarms. Note that this model cannot more be called TAPPMOG, because global reasoning is applied.
In [6], the number of Gaussian components is automatically chosen, using a Maximum A-Posteriori (MAP) test and employing a negative Dirichlet prior, able to associate more than a single Gaussian component where the BG exhibits a multimodal behavior, thus allowing a faster BG maintenance.
Another per-pixel approach is proposed in [5]: this model uses a non-parametric prediction algorithm to estimate the probability density function of each pixel, which is continuously updated to capture fast gray level variations. In [13], pixel value probability densities, represented as normalized histograms, are accumulated over time, and BG label are assigned by the Maximum A Posteriori criterion.
Region-based algorithms usually divide the frames into blocks and calculate block-specific features; change detection is then achieved via block matching, considering for example fusion of edge and intensity information [7]. In [8] a region model describing local texture characteristic is presented:the method is prone to errors when shadows and sudden global changes of illumination occur.
Frame-level class is formed by methods that look for global changes in the scene. Usually, they are used jointly with other pixel or region BG subtraction approaches. In [9], a graphical model was used to adequately model illumination changes of the scene. In [10], a BG model was chosen from a set of pre-computed ones, in order to minimize massive false alarm.
Hybrid models describe the BG evolution using jointly pixel and region models, and adding in general post-processing steps. In Wallflower [1], a 3-stage algorithm is presented, which operates respectively at pixel, region and frame level. Wallflower test sequences are widely used as comparative benchmark for BG subtraction algorithms. In [14], a non parametric, per pixel FG estimation is followed by a set of morphological operations in order to solve a set of BG subtraction common issues. In [15] a region level step, in which the scene is modeled by a set of local spatial-range codebook vectors, is followed by an algorithm that decides at the frame-level whether an object has been detected, and several mechanisms that update the background and foreground set of codebook vectors. For a good BG subtraction methods review, see [11].

3 Proposed method

Let n, $n = 1, \ldots, N$ be a pixel location, z_n be the pixel signal observed at location n and $z_n^{(t)}$ be a realization of such signal at time t. The decision to classify $z_n^{(t)}$ as BG or FG is given by a two-step process. The first step is the *per-pixel* (PP) process, the second step is the *per-region* (PR) process.

3.1 The per-pixel process

The PP process controls whether the per-pixel information is sufficient to explain $z_n^{(t)}$ as a BG value. In this paper, each pixel signal is modeled using a set of R Gaussian pdf's $\mathcal{N}(\cdot)$, as proposed by [3]. The probability of observing the value $z_n^{(t)}$ is:

$$P(z_n^{(t)}) = \sum_{r=1}^{R} w_{n,r}^{(t)} \mathcal{N}(z_n^{(t)} | \mu_{n,r}^{(t)}, \sigma_{n,r}^{(t)}) \qquad (1)$$

where $w_{n,r}^{(t)}$, $\mu_{n,r}^{(t)}$ and $\sigma_{n,r}^{(t)}$ are the time adaptive mixing coefficients, the mean, and the standard deviation, respectively, of the r-th Gaussian of the mixture associated with $z_n^{(t)}$. At each time instant, the Gaussian components are evaluated in descending order with respect to w/σ to find the first matching with $z_n^{(t)}$ (a *match* occurs if the value falls within 2.5σ of the mean of the component). If no match occurs, the least ranked component is replaced with a new Gaussian with the mean equal to the current value, high variance and a low mixing coefficient. If r_{hit} is the matched Gaussian component, the value $z^{(t)}$ is labeled FG if

$$\sum_{r=1}^{r_{\text{hit}}} w_r^{(t)} \; > \; T \tag{2}$$

where T is a standard threshold, the summation $\sum_{r=1}^{r_{\text{hit}}} w_r^{(t)}$ represents the probability that the Gaussian components considered do model the background. We call the test in (2) as the *background per-pixel test* (BG PP test), which is true ($= 1$) if the value is labeled BG ($z_n^{(t)} \in BG$), false ($= 0$) vice versa. For further details, see [3].

3.2 The per-region process

If $z_n^{(t)}$ is not recognized as BG by the BG PP test, then the PR process determines if $z_n^{(t)}$ is similar to another BG signal value, located in a close position n'. In formulae, $z_n^{(t)}$ is labeled BG by the PR process if the following *background per-region test* (BG PR test) is true:

$$\bigvee_{n' \in G_n} \left(z_n^{(t)} \in \mathcal{N}(\mu_{n',\tilde{k}}, \sigma_{n',\tilde{k}}) \bigwedge z_{n'}^{(t)} \in BG \right) \tag{3}$$

where $\bigvee, \bigwedge$ indicate *or* and *and* operators respectively, G_n is the squared neighborhood zone related to location n and $\tilde{k}$ addresses whatever Gaussian component that models the pixel location n', which satisfies the condition above. Eq.(3) is true if $z_n^{(t)}$ matches with a particular Gaussian component located at position n' *and* the signal value $z_{n'}^{(t)}$, modeled by such Gaussian pdf, is labeled BG by the PP process. The BG labeling mechanism exploited in the PR process mirrors the policy proposed in [12]. If some part of the background (a tree branch for example) moves to occupy a new pixel, but it was not part of the per-pixel model for that pixel, then it will be detected as a FG object by a classical per-pixel method. However, this object will have a high probability to be a part of the BG distribution in its original position. Clearly, the bigger the neighborhood zone G_n, the heavier will be the computational load required for evaluating Eq.(3). In this paper, we propose to adopt a strategy for changing G_n on-line, in order to turn down the computational effort. From here, we indicate with $s_n^{(t)}$ half the size of the neighborhood zone G_n at time t, resulting in a square of odd size $1 + 2s_n^{(t)}$. At time $t = 0$, $s_n^{(0)} = s_{min}$, where s_{min} indicates the smallest length permitted. In this paper, we set s_{min} as 0, resulting in a neighborhood zone of a single pixel location. At time $t = 1$, if the BG PP test is negative (i.e., we have a FG per-pixel detection at location n), the PR process does not contributes to find a BG neighborhood signal similar to $z_n^{(1)}$; therefore, the whole process will give a FG detection at position n. After this, s_n is enlarged by a factor γ_s, obtaining a squared region of size $\lfloor 1 + 2 * \gamma_s \rfloor$. If the PR+PP process continues to identify a FG value at position n, a maximal length s_{max} is considered. Viceversa, if the PR test is positive, we have a BG detection, and the growing process of s_n stops. Conversely, let us suppose that at frame t the BG PP test is positive. This means that the per-pixel statistic is enough to explain the pixel signal $z_n^{(t)}$ as a BG instance. Therefore, having a large neighborhood zone of size s_n to analyze brings to a useless computational burden. Consequently, the size $s_n^{(t)}$ is diminished by the factor γ_s. Hence, if the PP BG test continues to be positive, then s_n is diminished until the smallest size s_{min} is reached. Summarizing, the size of $s_n^{(t)}$ changes as follows:

$$s_n^{(t)} = \begin{cases} s_{max} & \text{if PP test}=0 \wedge \\ & \text{PR test}=0 \wedge s_n^{(t-1)}=s_{max} \\ s_n^{(t-1)} + \gamma_s & \text{if PP test}=0 \wedge \text{PR test}=0 \\ s_n^{(t-1)} & \text{if PP test}=0 \wedge \text{PR test}=1 \\ s_n^{(t-1)} - \gamma_s & \text{if PP test}=1 \wedge s_n^{(t-1)} > s_{min} \\ s_{min} & \text{if PP test}=1 \wedge s_n^{(t-1)}=s_{min} \end{cases} \tag{4}$$

A graphical example of the process is given in Fig.1.

When the smallest size s_{min} has been reached for the location n, if the signal $z_n^{(t)}$ is detected as BG simply by the BG PP process, we decide to sample it every $\lfloor I(n)^{(t)} \rfloor$ frames, where $I(n)^{(t)}$ is a *skipping* time set initially to 0, which increments by a factor γ_t each time $z_n^{(t)}$ is discovered consecutively as BG by the BG PP test. This happens until the maximal skip interval I_{max} is reached.

During the skip, the Gaussian parameters that describe the signal z_n are left unchanged. This temporal sampling process stops immediately as soon as the BG PP test is negative, and $I(n)^{(t)}$ is set to 0.

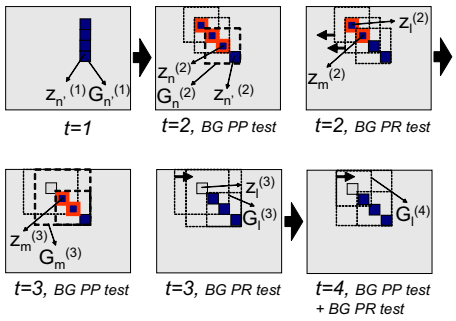

Figure 1: Scheme of ASTNA: At time $t = 1$, suppose that only the blue pixels are sampled. At time $t = 2$ location $z_n^{(2)}$ is detected as FG (the pixel is rounded by a red square) by the BG PP test, but the BG PR test is positive, having the region $G_n^{(2)}$ a BG pixel value $z_n'^{(2)}$ similar to $z_n^{(2)}$ inside. The other values $z_l^{(2)}$ and $z_m^{(2)}$ are detected as FG by the PR BG test, thus their neighborhood zones are enlarged. At time $t = 3$ the value $z_m^{(3)}$ is labelled BG by the PR test, while the pixel value $z_l^{(3)}$ is labelled BG by the PP test, and its neighborhood zone can be diminished. Only BG values are detected at time $t = 4$.

The quality of the results obtained justifies the heuristic aspect of the proposed method.

4 Experiments and discussion

Several tests have been performed to validate the proposed approach. In the first benchmark, the well-known "Wallflower" dataset [1] has been considered; it contains sequences which present different hard BG subtraction issues; each sequence has a ground truth frame. Here, we processed four of the most difficult sequences of the dataset, i.e., sequences whose best BG subtraction results are far from the ground truth. The sequences are: 1) *Waving Tree* (WT): a tree is swaying and a person walks in front of the tree; 2) *Camouflage* (C): a person walks in front of a monitor, which has rolling interference bars on the screen. The bars include color similar to the persons clothing; 3) *Bootstrapping* (B): a busy cafeteria where each frame contains FG objects; 4) *Foreground Aperture* (FA): a person with uniformly colored shirt wakes up and begins to move. Considering the other three Wallflower sequences, two of them refer i) to the capacity of the background model to incorporate a moved object in the background model after a reasonable time it is still, and ii) to the capacity of the background model to adapt to a gradual change of illumination. Both these problems are solved by the ordinary TAPPMOG, and thus our model does not add any deterioration in the performance. The third sequence present an instance of the sudden global change in illumination issue. Our method fails in this case, being absent a per-frame module. Anyway, such problem does not represent an important issue, being present several techniques able to solve it with very low computational effort (for example, using a set of pre-learnt global model of the scene, and using the most appropriate, as done in [10]). Another issue that we do not face in this papers are the shadows issue, another problem in video surveillance. This will be addressed in a future work, as explained in the last section.

All the RGB sequences are captured at resolution of 160×120 pixels, sampled at 4Hz. In our pure MATLAB implementation, on a Pentium IV, 3Ghz, 1Gb RAM, we set $A_{max} = 5, I_{max} = 4$, $\gamma_s = \gamma_t = 0.2$; such quantities are intuitive and easy to set. For each sequence, we show qualitative results in Fig.2: for lack of space only the results related to the TAPPMOG [3] and Elgammal [12] methods are reported[1]): for a more extensive listing of the existent results, please see [1]. In Fig.3, a wider set of quantitative results are provided in terms of amount of false positive and false negative FG detections. In particular, Wallflower, SACON, Tracey Lab LP and Bayesian Decision refer to [1, 14, 15, 13] respectively, which have been previously discussed in Section 2. As visible, all the results provided by ASTNA are comparable with the best performances obtained by other techniques; in particular, our method reach optimal results in the *Waving Tree* test, by correctly modelling as BG the tree. In the *Bootstrap* test, our method correctly considers as BG the light reflexes on the floor, even if they are irregularly occurring with sudden small displacements. This because our method learns the zones of the scene in which oscillating or flickering background is present, permitting to use in those areas large neighborhood zones. At the same time, zones in which the BG

[1]Regarding the Elgammal method, the neighborhood zone is represented by a fixed squared zone of size 5 for all the pixel locations $n = 1, \ldots, N$.

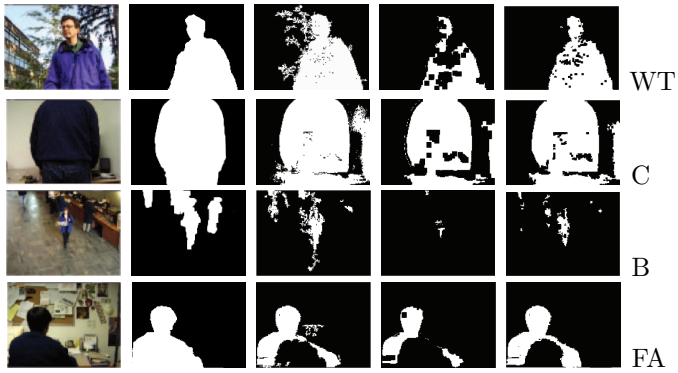

Figure 2: Wallflower qualitative results: on the 1st col., the test frames; on the 2nd col. the ground truth; the 3rd col. shows the TAPPMOG results; Elgammal results and results obtained with our method ASTNA are on the last two columns, respectively.

Methods	*Err.*	*WT*	*C*	*B*	*FA*	*T.Err.*
Wallflower	**f.neg.**	877	229	2025	320	9170
	f.pos.	1999	2706	365	649	
	t.e.	2876	2935	2390	969	
SACON	**f.neg.**	41	47	1150	1508	4084
	f.pos.	230	462	125	521	
	t.e.	271	509	1275	2029	
Tracey LAB LP	**f.neg.**	191	1998	1974	2403	7219
	f.pos.	136	69	92	356	
	t.e.	327	2067	2066	2759	
Bayesian decision	**f.neg.**	629	1538	2143	2501	14043
	f.pos.	334	2130	2764	1974	
	t.e.	963	3688	4907	4485	
TAPPMOG	**f.neg.**	56	220	1732	2217	10059
	f.pos.	1533	2398	1033	870	
	t.e.	1589	2618	2765	3087	
Elgammal	**f.neg.**	2899	2239	2688	3101	13019
	f.pos.	1	881	0	310	
	t.e.	2900	3120	2688	3411	
ASTNA	**f.neg.**	253	823	2349	1900	7031
	f.pos.	100	1173	73	360	
	t.e.	353	1996	2422	2260	

Figure 3: Quantitative results obtained by the proposed ASTNA method: *f.neg*, *f.pos.*,*t.e.* and *T.Err* mean false negative, false positive per-pixel FG detections, total errors on the specific sequence and total errors summed on all the sequences analyzed, respectively. Our method outperformed the most effective general purposes BG subtraction scheme (Wallflower, Bayesian decision, TAPPMOG, Elgammal), and is comparable with methods which are more time demanding (see Tab.1) and strongly constrained by data-driven initial hypotheses (SACON and Tracey Lab LP).

is still and stable are considered as formed by independent pixel locations with minimal 1-pixel neighborhood zone, and, as a consequence, they are more sensible to FG occurrences. Another observation is that our method can be thought as improving the performances of the TAPPMOG method. Actually, TAPPMOG models over the same pixel different Gaussian components, taking into account different chromatic modes of the background. Our method permits to share these modes among adjacent pixels locations, if necessary. One can afford that similar results can be provided by augmenting the number of per-pixel Gaussian components, but doing this way occasional reflections of the background cannot effectively be modeled. As a further result, on Table 1 the total execution time needed by the different algorithms to process the test sequences is reported. One can notice the timings of SACON, which is the only one that outperforms our method for what concerns quality of the results. All the other methods of Fig.3 not reported in Table 1 exhibit worse performances. These results show that ASTNA outperforms both the fixed-neighborhood zones method [12] and the classical TAPPMOG method. This last result is due to the fact that the computational effort required by ASTNA to inspect a neighborhood zone for each pixel is counterbalanced by the fact that ASTNA avoids to sample locations with stable pixel value at each iteration.

Methods	WT	C	B	FA
SACON	47.33	49.33	50.52	81.5
TAPPMOG	64.15	65.04	67.46	108.44
Elgammal	75.20	67.80	72.16	108.64
ASTNA	**33.49**	**28.12**	**33.02**	**39.52**

Table 1: Times of execution in seconds of the different BG subtraction methods when applied to the Wallflower sequences.

To give a better insight on how our method performs, we consider another hard sequence, 320×240 pixels, 1170 frames long, in which a docking scenario is portrayed. In Fig.4a) some frames are shown and one can notice that reflecting sea and oscillating boats are present.

During the sequence, a person arrives near the camera, goes up on a boat, and lastly goes away. The images in Fig.4d), at each pixel, indicate the area of the related neighborhood zone calculated by our algorithm. As one can note, our method considers bigger zones only where significant oscillations are present (near the masts). The time occurred for process such sequence is 1478.20 sec. for Elgammal, 1067.60 sec. for TAPPMOG and 374.28 sec. for ASTNA. In Fig.5, the time required for each iteration is reported, for all the three methods, at different frames; it is evident that our method is upper-bounded by the fixed-neighborhood Elgammal technique, while outper-

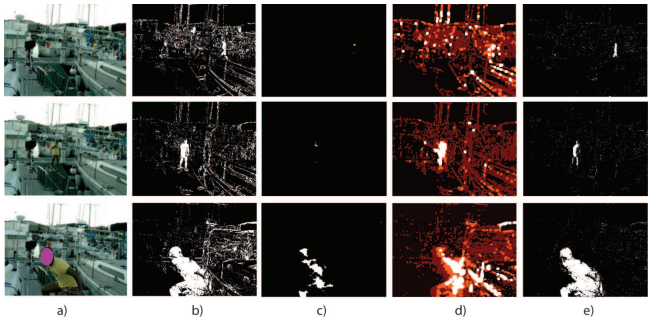

Figure 4: Dock sequence: a) some frames of the sequence (the face of the person is obscured due to the anonymity issues); b) TAPPMOG results; c) Elgammal results; d) ASTNA neighborhood image: brighter pixels mean wider neighborhood zones for that pixel; d) ASTNA results. Note that in all these results were not applied morphological operations to clean up small FG detections.

forms TAPPMOG method after the on-line learning of the most adapt neighborhood zones has been performed.

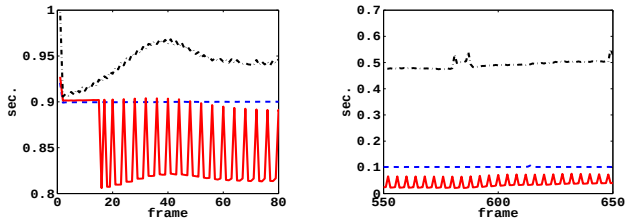

Figure 5: Timings for the "Dock" sequence: the dashed line, the point-dashed line and the solid line indicates TAPPMOG, Elgammal and ASTNA timings, respectively.

5 Conclusions and future perspectives

In this paper, we focus on producing a strategy which is able to perform background subtraction in a fast and robust way. The idea is to use an already present effective background subtraction technique, which operates per-pixel, namely the TAPPMOG algorithm, and to adapt it in order to deal with patch of pixels. This contributes to avoid false alarms caused by irregular scene variations, such as happens in a sea-docking scenario. Hence, we introduce a method which effectively selects the area of support over which the algorithm can operate. The idea is that, the larger the background variations, the wider will be the pixel area where the algorithm can look for a unstable background pixel. The proposed method is also able to change the sampling rate with which the pixels values are processed: in short, where no foreground activities are present, and where the background is spatially stable, the sampling rate will become very low, otherwise it will be high. This permits to compensate the computational burden to to the per region processing, improving time performances. In the future, we intend to apply the RGB normalization of [4] in order to cope successfully with the shadows, and to add the Gaussian model selection algorithm proposed by [6], (for the explanation of such methods, see Sec. 2 in order to further speed up the BG subtraction performances. Our goal is to use this method as a base module in a distributed video surveillance framework, where the computational load has to be maintained as low as possible.

REFERENCES

[1] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers, "Wallflower: Principles and practice of background maintenance," in *Int. Conf. Computer Vision*, 1999, pp. 255–261.

[2] C.R. Wren, A. Azarbayejani, T. Darrell, and A. Pentland, "Pfinder: Real-time tracking of the human body," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 780–785, 1997.

[3] C. Stauffer and W.E.L Grimson, "Adaptive background mixture models for real-time tracking," in *Int. Conf. Computer Vision and Pattern Recognition*, 1999, vol. 2, pp. 246–252.

[4] D. Suter H. Wang, "A re-evaluation of mixture of gaussian background modeling," in *Proc. of the IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, 2005*, 2005, vol. 2, pp. ii/1017– ii/1020.

[5] A. Mittal and N. Paragios, "Motion-based background subtraction using adaptive kernel density estimation," in *Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 2004, pp. 302–309, IEEE Computer Society.

[6] Z. Zivkovic, "Improved adaptive gaussian mixture model for background subtraction," in *Proc. of the International Conference on Pattern Recognition*, 2004, pp. 28–31.

[7] P. Noriega and O. Bernier, "Real time illumination invariant background subtraction using local kernel histograms," in *Proc. of the British Machine Vision Conference*, 2006.

[8] M. Heikkila and M.Pietikainen, "A texture-based method for modeling the background and detecting moving objects," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 4, pp. 657–662, 2006.

[9] B. Stenger, V. Ramesh nad N. Paragios, F.Coetzee, and J. M. Buhmann, "Topology free hidden Markov models: Application to background modeling," in *Int. Conf. Computer Vision*, 2001, vol. 1, pp. 294–301.

[10] N. Ohta, "A statistical approach to background subtraction for surveillance systems," in *Int. Conf. Computer Vision*, 2001, vol. 2, pp. 481–486.

[11] Massimo Piccardi, "Background subtraction techniques: a review," in *SMC (4)*, 2004, pp. 3099–3104.

[12] A. Elgammal, D. Harwood, and L.S. Davis, "Non-parametric model for background substraction," in *European Conf. Computer Vision*, 2000.

[13] H. Nakai, "Non-parameterized bayes decision method for moving object detection," in *Proc. Second Asian Conf. Computer Vision*, 1995, pp. 447–451.

[14] H. Wang and D. Suter, "Background subtraction based on a robust consensus method," in *ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition* , Washington, DC, USA, 2006, pp. 223–226, IEEE Computer Society.

[15] D. Kottow, M. Köppen, and J.Ruiz del Solar, "A background maintenance model in the spatial-range domain," in *ECCV Workshop SMVP*, 2004, pp. 141–152.