Self Organizing and Fuzzy Modelling for Parked Vehicles Detection

Lucia Maddalena[1] and Alfredo Petrosino[2]

[1] ICAR - National Research Council
Via P. Castellino 111, 80131 Naples, Italy
lucia.maddalena@na.icar.cnr.it
[2] DSA - University of Naples Parthenope
Centro Direzionale, Isola C/4, 80143 Naples, Italy
alfredo.petrosino@uniparthenope.it

Abstract. Our aim is to distinguish moving and stopped objects in digital image sequences taken from stationary cameras by a model based approach. A self-organizing model is adopted both for the scene background and for the scene foreground, that can handle scenes containing moving backgrounds or gradual illumination variations, helping in distinguishing between moving and stopped foreground regions. The model is enriched by spatial coherence to enhance robustness against false detections and fuzzy modelling to deal with decision problems typically arising when crisp settings are involved. We show through experimental results and comparisons that good accuracy values can be reached for color video sequences that represent typical situations critical for vehicles stopped in no parking areas.

Keywords: moving object detection, background subtraction, background modeling, foreground modeling, stopped object, self organization, neural network.

1 Introduction

Stopped object detection in an image sequence consists in detecting temporally static image regions indicating objects that do not constitute the original background but were brought into the scene at a subsequent time, such as abandoned and removed items, or illegally parked vehicles.

Great interest in the stopped object detection problem has been given by the PETS workshops held in 2006 [11] and in 2007 [12], where one of the main aims has been the detection of *left luggage*, that is luggage that has been abandoned by its owner, in movies taken from multiple cameras. Another example of strong interest in the considered problem is given by the *i-LIDS bag and vehicle detection challenge* proposed in the AVSS 2007 Conference [24], where the attention has been driven on abandoned bags and parked vehicles events, properly defined.

A broad classification of existing approaches to the detection of stopped objects can be given as *tracking-based* and *non tracking-based* approaches. In *tracking-based* approaches, where the stopped object detection is obtained on the basis of

J. Blanc-Talon et al. (Eds.): ACIVS 2009, LNCS 5807, pp. 422–433, 2009.

the analysis of object trajectories through an application dependent event detection phase. These include most of the papers in [11,12]. *Non tracking-based* approaches include pixel- and region-based approaches aiming at classifying pixels/ objects without the aid of tracking modules, and include [6,15,16,21,23].

Our approach to the problem is non tracking-based. The problem is tackled as *stopped foreground subtraction*, that, in analogy with the background subtraction approach, consists in maintaining an up-to-date model of the stopped foreground and in discriminating moving objects as those that deviate from such model. Both background subtraction and stopped foreground subtraction have the common issue of constructing and maintaining an image model that adapts to scene changes and can capture the most persisting features of the image sequence, i.e. the background and stationary foreground, respectively. For such modeling problem we adopt visual attention mechanisms that help in detecting features that keep the user attention, based on a self-organizing neural network.

One of the main issues to be pursued in background subtraction is the uncertainty in the detection caused by the cited background maintenance issues. Usually, crisp settings are needed to define the method parameters, and this does not allow to properly deal with uncertainty in the background model. Recently several authors have explored the adoption of fuzzy approaches to tackle different aspects of detecting moving objects. In [30] an approach using fuzzy Sugeno integral is proposed to fuse texture and color features for background subtraction, while in [2,3] the authors adopt the Choquet integral to aggregate the same features. In [26] a fuzzy approach to selective running average background modeling is proposed, and in [1] the authors model the background by the Type-2 Fuzzy Mixture of Gaussian Model proposed in [31].

The approach we propose is based on the background and the foreground model automatically generated by a self-organizing method without prior knowledge of the pattern classes. An automatic and data dependent fuzzy mechanism is introduced into the update phase for further reinforcing into the background model the contribution of pixels that belong to it. The approach consists in using biologically inspired problem-solving methods to solve motion detection tasks, typically based on visual attention mechanisms. The aim is to obtain the objects that keep the users attention by referring to a set of predefined features.

The paper is organized as follows. In Section 2 we describe a model-based pixelwise procedure allowing to discriminate foreground pixels into stopped and moving pixels, that is completely independent on the background and foreground models adopted. In Section 3 we describe the model for both background and foreground modeling that we adopted in our experiments, that is a variation of a previously presented model for background modeling. Section 4 presents results obtained with the implementation of the proposed approach, while Section 5 includes concluding remarks.

2 Stopped Foreground Detection

In this section we propose a model-based approach to the classification of foreground pixels into stopped and moving pixels. A foreground pixel is classified as *stopped* if it holds the same color features for several consecutive frames; otherwise it is classified as *moving*.

Assuming we have a model BG_t of the image sequence background, we compute a function $E(x)$ of color feature occurrences for pixel $I_t(x)$ as follows

$$E(x) = \begin{cases} \min(\tau_s, E(x)+1) \text{ if } & I_t(x) \notin BG_t \ \wedge \ I_t(x) \in FG_t \\ \max(0, E(x)-1) \text{ if } & I_t(x) \notin BG_t \ \wedge \ I_t(x) \notin FG_t \\ \max(0, E(x)-k) \text{ if } & I_t(x) \in BG_t \end{cases} \tag{1}$$

where model FG_t of the sequence foreground is iteratively built and updated using image pixels $I_t(x)$ for which $E(x) > 0$.
Every time pixel $I_t(x)$ belongs to the foreground model ($I_t(x) \in FG_t$), $E(x)$ is incremented, while it is decremented if it does not belong to the foreground model. The maximum value τ_s for $E(x)$ corresponds to the *stationarity threshold*, i.e. the minimum number of consecutive frames after which a pixel assuming constant color features is classified as stopped. The value for τ_s is chosen depending on the desired responsiveness of the system.
On the contrary, if pixel $I_t(x)$ is detected as belonging to the background ($I_t(x) \in BG_t$), $E(x)$ is decreased by a factor k. The decay constant k determines how fast $E(x)$ should decrease, i.e. how fast the system should recognize that a stopped pixel has moved again. To set the alarm flag off immediately after the removal of the stopped object, the value of decay should be large, eventually equal to τ_s.
Pixels $I_t(x)$ for which $E(x)$ reaches the stationarity threshold value τ_s are classified as stopped, and therefore the set ST_t defined as

$$ST_t = \{FG_t(x) : E(x) = \tau_s\}$$

supplies a model for the stopped objects, while the remaining part of FG_t represents moving objects.

The described procedure is completely independent on the model adopted for the scene background and foreground. The model that we have adopted for the background and the foreground will be described in the following section.

3 Background and Foreground Update

Relying on recent research in this area [18,19], for background and foreground modeling a self-organizing neural network, organized as a 3-D grid of neurons, is built up. Each neuron computes a function of the weighted linear combination of incoming inputs, with weights resembling the neural network learning, and can be therefore represented by a weight vector obtained collecting the weights related to incoming links. An incoming pattern is mapped to the neuron whose set of weight vectors is most similar to the pattern, and weight vectors in a neighborhood of such node are updated.

Specifically, for each pixel $p_t = I(x)$ we build a neuronal map consisting of L weight vectors $c^l(p_t), l = 1, \ldots, L,$. Each weight vector $c^l(p_t)$ is represented in the HSV colour space, that allows to specify colours in a way that is close to

Fig. 1. A simple image (left) and the modeling neuronal map with $L = 4$ layers (right)

human experience of colours, and is initialized to the HSV components of the corresponding pixel of the first sequence frame $I_0(p_t)$. The complete set of weight vectors for all pixels of an image I with N rows and M columns is organized as a 3D neuronal map $\tilde{B}$ with N rows, M columns, and L layers. An example of such neuronal map is given in Fig. 1, which shows that for each pixel $p_t = I_t(x)$ (identified by one of the colored circles in the sequence frame on the left) we have a weight vector $\tilde{B}_t(x) = (c^1(x), c^2(x), \ldots, c^L(x))$ (identified by correspondingly colored circles in the model layers on the right).

By subtracting the current image from the background model $\tilde{B}$, each pixel p_t of the t-th sequence frame I_t is compared to the current pixel weight vectors to determine if there exists a weight vector that matches it. The best matching weight vector is used as the pixel's encoding approximation, and therefore p_t is detected as foreground if no acceptable matching weight vector exists; otherwise it is classified as background.

Matching for the incoming pixel $p_t = I_t(x)$ is performed by looking for a weight vector $c^b(p_t)$ in the set $\tilde{B}_t(x) = (c^1(p_t), \ldots, c^L(p_t))$ of the current pixel weight vectors satisfying:

$$d(c^b(p_t), p_t) = \min_{i=1,\ldots,L} d(c^i(p_t), p_t) \leq \varepsilon \tag{2}$$

where the metric $d(\cdot)$ and the threshold ε are suitably chosen as in [18].

The best matching weight vector $c^l(p_t) = \tilde{B}_t(x)$ belonging to layer l and all other weight vectors in a $n \times n$ neighborhood N_{p_t} of $c^l(p_t)$ in the l-th layer of the background model $\tilde{B}$ are updated $\forall x \in N_{p_t}$ according to selective weighted running average:

$$\tilde{B}^l_t(x) = (1 - \alpha_t(x))\tilde{B}^l_{t-1}(x) + \alpha_t(x) I_t(x) \tag{3}$$

where $\alpha_t(x)$ is a learning factor, later specified, belonging to [0,1] and depends on scene variability. If the best match $c^b(p_t)$ satisfying eq. (2) is not found, the background model $\tilde{B}$ remains unchanged. Such selectivity allows to adapt the background model to scene modifications without introducing the contribution of pixels not belonging to the background scene.

Spatial coherence is also introduced in order to enhance robustness against false detections. Let $p = I(x)$ the generic pixel of image I, and let N_p a spatial square neighborhood of pixel $p \in I$. We consider the set Ω_p of pixels belonging to N_p that have a best match in their background model according to eqn. (2), i.e.

$$\Omega_p = \{q \in N_p : d(c^b(q), q) \leq \varepsilon\}.$$

In analogy with [8], the *Neighborhood Coherence Factor* is defined as:

$$NCF(p) = \frac{|\Omega_p|}{|N_p|}$$

where $|\cdot|$ refers to the set cardinality. Such factor gives a relative measure of the number of pixels belonging to the spatial neighborhood N_p of a given pixel p that are well represented by the background model $\tilde{B}$. If $NCF(p) > 0.5$, most of the pixels in such spatial neighborhood are well represented by the background model, and this should imply that also pixel p is well represented by the background model. Values for $\alpha_t(x)$ in eq. (3) are therefore expressed as

$$\alpha_t(x) = M(p_t)\ \alpha(t)\ w(x), \quad \forall x \in N_{p_t}, \tag{4}$$

where $w(x)$ are Gaussian weights in the neighborhood N_{p_t}, $\alpha(t)$ represents the learning factor, that is the same for each pixel of the t-th sequence frame, and $M(p_t)$ is the crisp hard-limited function

$$M(p_t) = \begin{cases} 1 & \text{if} \quad NCF(p_t) \geq 0.5 \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

The background updating rule is formulated in terms of a production rule of the type: if (condition) then (action), incorporating knowledge of the world in which the system works, such as knowledge of objects and their spatial relations. When the condition in the production rule is satisfied, the action is performed. Both condition and action are described in linguistic terms and a numeric method should be adopted to represent the vagueness inherent in these labels effectively. In particular, the flexibility and power provided by fuzzy set theory for knowledge representation makes fuzzy rule-based systems very attractive when compared with traditional rule-based systems. In our case, the uncertainty resides in determining suitable thresholds in the back- ground model. According to this way of reasoning, the fuzzy background subtraction and update algorithm for the generic pixel $p_t \in I_t$ can be stated through a fuzzy rule-based system as follows:

Fuzzy rule-based background subtraction and update algorithm

if ($d(c_m(p_t), p_t)$ is `low`) and ($NCF(p_t)$ is `high`) then
  Update $\tilde{B}_t$
endif

Let $F_1(p_t)$ the fuzzy membership function of $d(c_m(p_t), p_t)$ to the fuzzy set `low` and $F_2(p_t)$ the fuzzy membership function of $NCF(p_t)$ to the fuzzy set `high`; the fuzzy rule becomes:

$$\alpha_t(x) = F_1(p_t)\ F_2(p_t)\ \alpha(t)\ w(x) \tag{6}$$

In order to take into account the uncertainty in the background model deriving by the need of the choice of a suitable threshold ε in eqn. (2), $F_1(p_t)$ is chosen as a saturating linear function given by

$$F_1(p_t) = \begin{cases} 1 - \dfrac{d(c_m(p_t), p_t)}{\varepsilon} & \text{if} \quad d(c_m(p_t), p_t) \leq \varepsilon \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

The function $F_1(p_t)$, whose values are normalized in $[0, 1]$, can be considered as the membership degree of p_t to the background model: the closer is the incoming sample p_t to the background model $C(p_t) = (c_1(p_t), c_2(p_t), \ldots, c_{n^2}(p_t))$, the larger is the corresponding value $F_1(p_t)$. Therefore, incorporating $F_1(p_t)$ in eq. (6) ensures that the closer is the incoming sample p_t to the background model, the more it contributes to the background model update, thus further reinforcing the corresponding weight vectors.

Also spatial coherence introduced through eqn. (5) can be formulated with a fuzzy approach. Indeed, we can observe that the greater is $NCF(p)$, the greater majority of pixels in N_p are well represented by the background model, and the better the pixel p can be considered as represented by the background model. Therefore we modify eq. (4) as follows:

$$\alpha_t(x) = F_2(p_t)\ \alpha(t)\ w(x), \tag{8}$$

where $F_2(p_t)$ is given as

$$F_2(p_t) = \begin{cases} 2 * NCF(p_t) - 1 & \text{if} \quad NCF(p_t) \geq 0.5 \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

and can be considered as the membership degree of pixel p_t to the background model.

The described model $\tilde{B}_t$ has been adopted for both the background model BG_t and the foreground model FG_t described in Section 2 for the classification of stopped and moving pixels.

4 Experimental Results

Experimental results for the detection of stopped objects using the proposed approach have been produced for several image sequences. Some of them are here described; some others can be found in the Supplement Material [20].

In Fig. 2 some frames are shown for sequence *Dataset1*, belonging to the publicly available *PETS 2001* dataset (ftp://ftp.pets.reading.ac.uk/pub/PETS2001). Here a blue car parks (from frame 700 till the sequence end in frame 2688), while a white van stops on the street side for about 600 frames (from frame 990 till to frame 1590). Choosing as stationarity threshold $\tau_S = 1500$, the presented approach allows to correctly detect the blue car as stationary, from frame 2160 till to the sequence end (green/red pixels indicate moving/stopped pixels, respectively). The white van, instead, is correctly detected as a non-stationary object, since it stops

700 990 2160 2220 2584

Fig. 2. Sequence *Dataset1*: selected original frames (first line), and corresponding stopped objects detection (second line). The blue car parks in frame 700, and it is detected as stationary after about $\tau_S = 1500$ frames (starting with frame 2160), till to the sequence end. The white van stops on the street side for about 600 frames (from frame 990 till to frame 1590), and therefore it is not detected as a stationary object.

for less than 1500 frames. It should be observed that the availability of the stationary model ST_t allows to clearly disambiguate moving cars passing in front of the stopped blue car, as in frame 2584.

In order to compare our results with those obtained by other existing approaches, we further consider parked vehicle sequences, named *PV-easy*, *PV-medium*, and *PV-hard*, belonging to the publicly available *i-LIDS 2007* dataset (ftp://motinas.elec.qmul.ac.uk/pub/iLids/), that have corresponding annotated ground truths provided for the AVSS 2007 contest [24]. Such scenes represent typical situations critical for moving object detection in outdoor sequences. Specifically, all the scenes present strong shadows cast by objects on the ground, light positional instability caused by small movements of the camera due to the wind, and strong and long-lasting illumination variations due to clouds covering and uncovering the sun. Experimental results showing the behavior of the adopted 3D neural model in such situations can be found in [20].
Concerning our specific purpose of detecting stopped objects, the considered i-LIDS scenes are devoted to detecting vehicles in no parking areas, where the street under control is more or less crowded with cars, depending on the hour of the day the scene refers to. The no parking area adopted for the AVSS 2007 contest [24] is defined as the main street borders, and the stationarity threshold is defined as $\tau_S = 1500$. This means that an object is considered irregularly parked if it stops in the no parking area for more than 60 seconds (scenes are captured at 25 fps).

Results obtained for sequence *PV-medium* are reported in Fig. 3, where we report only stationary pixels (in red), and not moving pixels that would overlap to stopped objects, hiding such detection results. The empty scene available at the beginning of the sequence (starting from frame 469) allows to train a quite faithful background model. As soon as the dark car stops (starting from frame 700), the function $E(x)$ described in Section 2 starts incrementing for

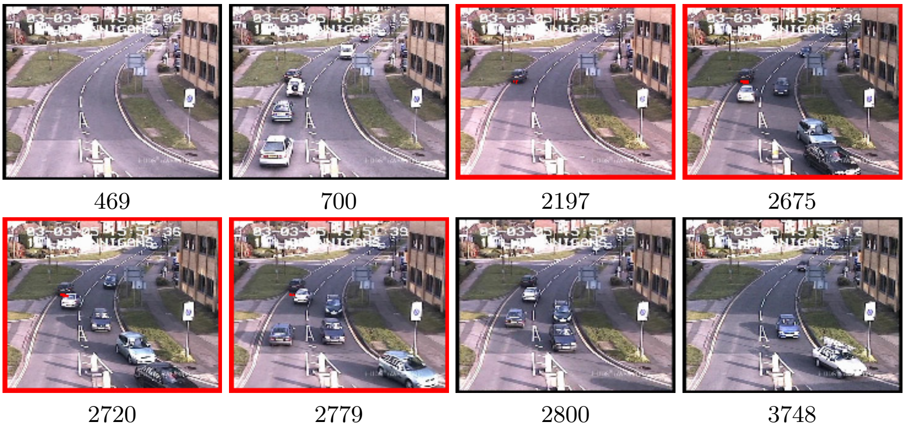

Fig. 3. Detection of stopped objects in sequence *PV-medium*. The car first stops in frame 700. The first stationary object is detected in frame 2197; further stationary pixels are later detected, even if the stopped object is occluded by foreground pixels (e.g. in frame 2720, where the white car covers the stopped car). The car is detected as stopped till to frame 2779, and no more stopped objects are detected till to frame 3748 (end of the sequence).

pixels belonging to the car; such pixels are inserted into the foreground model FG_t and used for the model update. After approximately τ_S=1500 frames, $E(x)$ reaches the stationarity threshold τ_S, thus signaling the first stopped object (frame 2197). From this moment till to the end of the stopped car event, the stopped object model allows to distinguish moving objects from the stopped object, as for example in frame 2720, where the white car covers the stopped car. When the car leaves again (from frame 2779), the part of the scene uncovered by the car is again recognized as belonging to the background model, and previously stopped pixels are deleted from the stopped object model.

It should be stressed that illumination conditions have changed quite a bit between the stopping and the leaving of the car. Therefore the actual background area uncovered by the car is very different from the background area that was modeled before the car stop, appearing as subject to a cast shadow. Our background model recognized that area again as background since it includes a mechanism, similar to the one adopted in [18], for detecting shadows and incorporating them into the background model. Specifically, shadow pixels are detected adopting the argument proposed in [7] and are incorporated into the background model according to eq. (3).

Moreover, it should be clarified that we do not identify the whole car, but only its part belonging to the no parking area, since, as suggested for the AVSS 2007 contest, we restrict our attention only to the street including the no parking area (masking out the remaining part of the scene).

Analogous qualitative results can be observed for the other considered *i-LIDS* sequences, as reported in [20].

Table 1. Comparison of ground truth (GT) stopped object event start and end times (in minutes) with those computed with our approach and with different approaches reported in [4,14,17,28], for considered sequences. Related absolute errors ($\varepsilon_O, \dots, \varepsilon_D$) are expressed in seconds; total error is computed as the sum of absolute errors over the three sequences.

Sequence	**Event**	**GT**	**Our**	ε_O	**A**	ε_A	**B**	ε_B	**C**	ε_C	**D**	ε_D
PV-easy	Start	02:48	02:45	3	02:48	0	02:46	2	02:52	4	02:52	4
"	End	03:15	03:19	4	03:19	4	03:18	3	03:19	4	03:16	1
PV-medium	Start	01:28	01:28	0	01:28	0	01:28	0	01:41	13	01:43	15
"	End	01:47	01:51	4	01:55	8	01:54	7	01:55	8	01:47	0
PV-hard	Start	02:12	02:12	0	02:12	0	02:13	1	02:08	4	02:19	7
"	End	02:33	02:34	1	02:36	3	02:36	3	02:37	4	02:34	1
Total error				12		15		16		37		28

We compared results obtained with our approach with those obtained with other approaches for the same sequences. Specifically we considered results obtained by four tracking-based approaches to the detection of stopped objects in the following denoted as: Method **A**, by Boragno et al. [4], who employ a DSP-based system for automatic visual surveillance where block matching motion detection is coupled with MOG-based foreground extraction; Method **B**, by Guler et al. [14], who extend a tracking system, inspired by the human visual cognition system, introducing a stationary object model where each region represents hypotheses stationary objects whose associated probability measures the endurance of the region; Method **C**, by Lee et al. [17], who present a detection and tracking system operating on a 1D projection of images; and Method **D**, by Venetianer et al. [28], who employ an object-based video analysis system featuring detection, tracking and classification of objects.

In Table 1 we report stopped object event start and end times provided with the ground truth and those computed with all considered approaches. Corresponding absolute errors show that generally our approach compares favorably to the other approaches, independently from scene traffic density, and this is still more evident if we consider the total error over the three considered sequences. It should be emphasized that, since our approach to stopped object detection is pixel-based and no region-based post-processing is performed in order to identify objects, in our case a stopped object event starts as soon as a single pixel is detected as stopped and ends soon as no more stopped pixels are detected.

Computational complexity of the proposed algorithm, both in terms of space and time, is $O(LNM)$, where L is the number of layers used for background and foreground models and $N \times M$ is the image dimension. To complete our analysis, in Fig. 4 we report execution times (in msecs/frame) for color image sequences acquired at a frequency of 25 frames per second, varying the spatial resolution and the number L of background and foreground model layers. Image sequences with Small (180x144), Medium (360x288), and High (720x576)

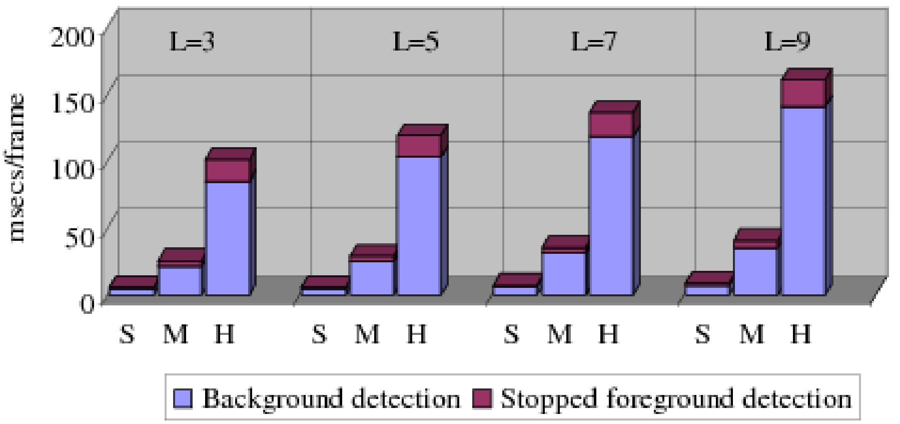

Fig. 4. Execution times (in msecs/frame) for the proposed algorithm on color image sequences with Small (S), Medium (M), and High (H) resolution, varying the number L of background and foreground model layers

resolution have been obtained by subsampling sequence *PV-medium*. Timings have been obtained by prototype implementations in C programming language on a Pentium 4 with 2.40 GHz and 512 MB RAM, running Windows XP operating system, and do not include I/O. The plot shows that only for high resolution sequences the frame rate is not sufficient to obtain real-time processing (about 40 msecs/frame). Nonetheless, we can observe that the stopped foreground detection times represent a small percentage (around 15%) of total execution times. Therefore, stopped foreground detection can be considered a useful and inexpensive by-product of background subtraction.

5 Conclusions

The paper reports our approach to the problem of *stopped foreground subtraction*, consisting in maintaining an up-to-date model of the stopped foreground and in discriminating moving objects as those that deviate from such model. For such modeling problem we adopt visual attention mechanisms that help in detecting features that keep the user attention, based on a 3D self-organizing neural network, without prior knowledge of the pattern classes. The aim is to obtain the objects that keep the user attention in accordance with a set of predefined features, by learning the trajectories and features of moving and stopped objects in a self-organizing manner. Such models allow to construct a system able to detect motion and segment foreground objects into moving or stopped objects, even when they appear superimposed.

References

1. Baf, F.E., Bouwmans, T., Vachon, B.: Type-2 Fuzzy Mixture of Gaussians Model: Application to Background Modeling. In: Bebis, G., Boyle, R., Parvin, B., Koracin, D., Remagnino, P., Porikli, F., Peters, J., Klosowski, J., Arns, L., Chun, Y.K., Rhyne, T.-M., Monroe, L. (eds.) ISVC 2008, Part I. LNCS, vol. 5358, pp. 772–781. Springer, Heidelberg (2008)
2. Baf, F.E., Bouwmans, T., Vachon, B.: A Fuzzy Approach for Background Subtraction. In: IEEE International Conference on Image Processing, ICIP 2008, San Diego, California, USA (October 2008)
3. Baf, F.E., Bouwmans, T., Vachon, B.: Fuzzy Integral for Moving Object Detection. In: IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 2008, Hong-Kong, China, June 1-6, pp. 1729–1736 (2008)
4. Boragno, S., Boghossian, B., Black, J., Makris, D., Velastin, S.: A DSP-based system for the detection of vehicles parked in prohibited areas. In: [24]
5. Cheung, S.-C., Kamath, C.: Robust Techniques for Background Subtraction in Urban Traffic Video. In: Proceedings of EI-VCIP, pp. 881–892 (2004)
6. Collins, R.T., Lipton, A.J., Kanade, T., Fujiyoshi, H., Duggins, D., Tsin, Y., Tolliver, D., Enomoto, N., Hasegawa, O., Burt, P., Wixson, L.: A System for Video Surveillance and Monitoring. The Robotics Institute, Carnegie Mellon University, Tech. Rep. CMU-RI-TR-00-12 (2000)
7. Cucchiara, R., Piccardi, M., Prati, A.: Detecting moving objects, ghosts, and shadows in video streams. IEEE Trans. Pattern Anal. Mach. Intell. 25(10), 1–6 (2003)
8. Ding, J., Ma, R., Chen, S.: A Scale-Based Connected Coherence Tree Algorithm for Image Segmentation. IEEE Transactions on Image Processing 17(2), 204–216 (2008)
9. Elhabian, S.Y., El-Sayed, K.M., Ahmed, S.H.: Moving Object Detection in Spatial Domain using Background Removal Techniques - State-of-Art. Recent Patents on Computer Science 1, 32–54 (2008)
10. Elgammal, A., Duraiswami, R., Harwood, D., Davis, L.S.: Background and Foreground Modeling Using Nonparametric Kernel Density Estimation for Visual Surveillance. Proceedings of the IEEE 90(7), 1151–1163 (2002)
11. Ferryman, J.M. (ed.): Proceedings of the 9th IEEE International Workshop on PETS, New York, June 18 (2006)
12. Ferryman, J.M. (ed.): Proceedings of the 10th IEEE International Workshop on PETS, Rio de Janeiro, Brazil, October 14 (2007)
13. Fisher, R.B.: Change Detection in Color Images, `http://homepages.inf.ed.ac.uk/rbf/PAPERS/iccv99.pdf`
14. Guler, S., Silverstein, J.A., Pushee, I.H.: Stationary objects in multiple object tracking. In: [24]
15. Herrero-Jaraba, E., Orrite-Urunuela, C., Senar, J.: Detected Motion Classification with a Double-Background and a Neighborhood-based Difference. Pattern Recognition Letters 24, 2079–2092 (2003)
16. Kim, K., Chalidabhongse, T.H., Harwood, D., Davis, L.S.: Real-time Foreground-Background Segmentation using Codebook Model. Real-Time Imaging 11, 172–185 (2005)
17. Lee, J.T., Ryoo, M.S., Riley, M., Aggarwal, J.K.: Real-time detection of illegally parked vehicles using 1-D transformation. In: [24]
18. Maddalena, L., Petrosino, A.: A Self-Organizing Approach to Background Subtraction for Visual Surveillance Applications. IEEE Transactions on Image Processing 17(7), 1168–1177 (2008)

19. Maddalena, L., Petrosino, A., Ferone, A.: Object Motion Detection and Tracking by an Artificial Intelligence Approach. International Journal of Pattern Recognition and Artificial Intelligence 22(5), 915–928 (2008)
20. Maddalena, L., Petrosino, A.: Further Experimental Results with Self Organizing and Fuzzy Modelling for Parked Vehicles Detection. Tech. Rep. No. RT-DSA-UNIPARTHENOPE-09-03, Dept. of Applied Sciences, University of Naples Parthenope (2009), http://www.dsa.uniparthenope.it
21. Patwardhan, K.A., Sapiro, G., Morellas, V.: Robust Foreground Detection in Video Using Pixel Layers. IEEE Transactions on PAMI 30(4) (2008)
22. Piccardi, M.: Background Subtraction Techniques: A Review. In: Proceedings of IEEE Int. Conf. on Systems, Man and Cybernetics, pp. 3099–3104 (2004)
23. Porikli, F., Ivanov, Y., Haga, T.: Robust Abandoned Object Detection Using Dual Foregrounds. EURASIP Journal on Advances in Signal Processing (2008)
24. Proceedings of 2007 IEEE Conference on Advanced Video and Signal Based Surveillance (AVSS 2007). IEEE Computer Society (2007)
25. Radke, R.J., Andra, S., Al-Kofahi, O., Roysam, B.: Image Change Detection Algorithms: A Systematic Survey. IEEE Transactions on Image Processing 14(3), 294–307 (2005)
26. Sigari, M.H., Mozayani, N., Pourreza, H.R.: Fuzzy Running Average and Fuzzy Background Subtraction: Concepts and Application. Int. J. of Computer Science and Network Security 8(2), 138–143 (2008)
27. Toyama, K., Krumm, J., Brumitt, B., Meyers, B.: Wallflower: Principles and Practice of Background Maintenance. In: Proceedings of the Seventh IEEE Conference on Computer Vision, vol. 1, pp. 255–261 (1999)
28. Venetianer, P.L., Zhang, Z., Yin, W., Lipton, A.J.: Stationary target detection using the objectvideo surveillance system. In: [24],
29. Wren, C., Azarbayejani, A., Darrell, T., Pentland, A.: Pfinder: Real-Time Tracking of the Human Body. IEEE Transactions on PAMI 19(7), 780–785 (1997)
30. Zhang, H., Xu, D.: Fusing Color and Texture Features for Background Model. In: Wang, L., Jiao, L., Shi, G., Li, X., Liu, J. (eds.) FSKD 2006. LNCS (LNAI), vol. 4223, pp. 887–893. Springer, Heidelberg (2006)
31. Zeng, J., Xie, L., Liu, Z.: Type-2 Fuzzy Gaussian Mixture Models. Pattern Recognition 41(12), 3636–3643 (2008)