

Efficient Incorporation of Motionless Foreground Objects for Adaptive Background Segmentation

I. Huerta¹, D. Rowe¹, J. González², and J.J. Villanueva¹

¹ Computer Vision Centre & Dept. d'Informàtica.
Edifici O. Campus UAB, 08193, Bellaterra, Spain
`ivan.huerta@cvc.uab.es`

² Institut de Robòtica i Informàtica Ind. UPC
Llorens i Artigas 4-6, 08028, Barcelona, Spain

Abstract. In this paper, we want to exploit the knowledge obtained from those detected objects which are incorporated into the background model since they cease their movement. These motionless foreground objects should be handled in security domains such as video surveillance. This paper uses an adaptive background modelling algorithm for moving-object detection. Those detected objects which present no motion are identified and added into the background model, so that they will be part of the new background. Such motionless agents are included for further appearance analysis and agent categorization.

1 Introduction

The analysis of human-motion image sequences involves different tasks, such as movement segmentation and tracking, action recognition and behaviour reasoning [7]. Therefore, the basis for high-level interpretation of observed patterns of human motion relies on *when* and *where* motion is being detected. Consequently, this low-level task still constitutes the most critical step towards Image Sequence Evaluation (ISE) [10].

In this work, the aim is to exploit at the Image Signal Level of the ISE architecture the knowledge obtained from those detected objects which could be incorporated into the background since they cease their movement. These "newly motionless" objects should be handled in security domains such as video surveillance. For example, if a suspicious bag is detected in an airport, some knowledge can be inferred: who left it there, where this bag within in the scene, when the person has left it. In traffic monitoring, if a car is stopped a predefined period of time, the position of this car is inferred within the scene, but also when and (the appearance of) who has left this car. Thus, incorporated objects constitute additional knowledge, which can be represented using feature-based models.

Different techniques have been used for motion segmentation [12], such as temporal differencing, optical flow and background subtraction. The latter consists of a background model used to compare the current image with such a model. Thus, foreground objects in motion are identified. To achieve this objective, many researchers have proposed methods which have been used to solve the

problems found in segmentation, such as gradual or sudden illumination changes, shadows, camouflage, background in motion, or deposited and removed objects from scene, among other problems [8].

Thus, W⁴ [4] uses a bimodal distribution, Pfister [13] uses a single Gaussian to model the background, Stauffer et al. [2,3] use a mixture of Gaussians, and Elgammal et al. [1] present a non-parametric background model. On the other hand, the features used for segmentation vary in the literature: Horprasert et al. [5] use colour information to classify a pixel as foreground, background, shadow or highlighted background, while Wallflower [9] uses a three-level categorization: pixel, region and frame level. Jabri et al. [6] use colour and edge information, and Shen [11] uses a RGB colour space plus fuzzy classification.

These approaches incorporate *gradually* new motionless foreground objects into the background model, that is, the updating rule of the background model incorporates increasingly all the pixel values which constitute the object. Therefore, a suitable representation of motionless objects cannot be built for post-processing tasks, such as object recognition or classification. Moreover, adaptive rules do not usually distinguish between background changes due to illumination than those changes due to left or removed objects.

In particular, W⁴ [4] first presented a differentiation between pixel-based and object-based detection: the *pixel-based update* method updates the background model periodically to adapt it to illumination changes, and the *object-based update* method updates the background to adapt it to physical changes, such as those objects deposited or removed into the background scene. Consequently, our work will be based on W⁴ in order to obtain a fast background scene modelling and maintenance while considering new incorporated objects. Therefore, such an adaptive background model is updated according to observed developments within the scene in order to achieve fast and robust segmentation results.

This paper is organized as follows. Section 2 shows how the background model is created. Section 3 describes the foreground region detection, and section 4 how the background model parameters are updated. Section 5 presents our contribution to object incorporation, and section 6 shows the results obtained. Finally, chapter 7 concludes this paper and discusses different alternatives for future research.

2 Initial Background Model

W⁴ uses a model of background variation constructed from order statistics of background values during a training period. The background scene is modelled by representing each pixel by three values: its minimum $m(x)$ and maximum $n(x)$ intensity values, and the maximum intensity difference $d(x)$ between consecutive frames observed during this training period. Furthermore, W⁴ uses a two-stage method for exclude foreground objects during training period, such as moving people. First, the median filter $|V^z(x) - \lambda(x)| < 2 * \sigma(x)$ distinguishes moving pixels from stationary pixels. $V^z(x)$ is the intensity of a pixel location x in the z -th image of sequence V , $\lambda(x)$ is the median value, and $\sigma(x)$ is the standard

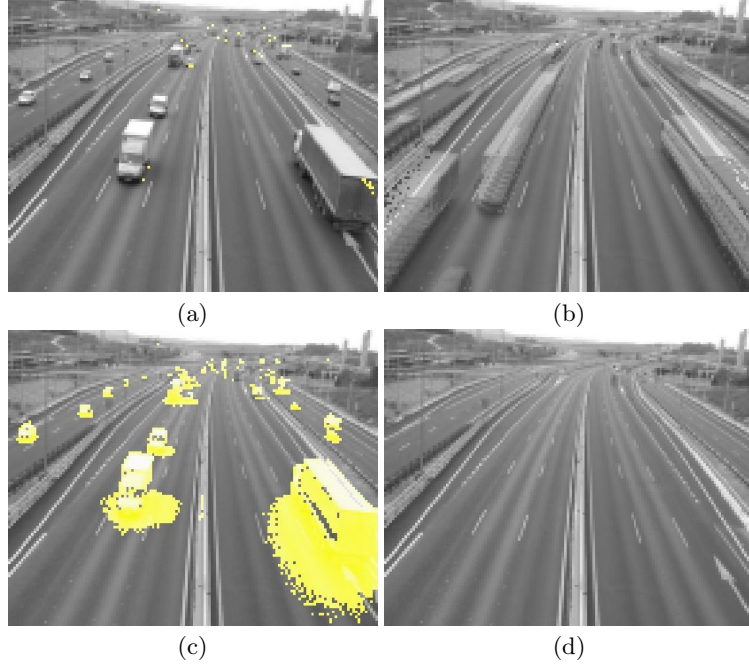


Fig. 1. Detection results in a road sequence with a high number of foreground objects during the training period: (a) foreground detection results using W^4 , showing that no detection is achieved; (b) the background is updated using W^4 , showing that cars are erroneously incorporated into the background model; (c) foreground detection results; and (d) background model update using our approach

deviation. After that, in the second stage, only stationary pixels are considered for building the initial background model.

However, a training period is not always available, because this period can contain multiple foreground objects in the scene, such as for the road sequence, see Fig. 1. The initial background model can be erroneous if foreground objects are incorporated into the background model. The two-stage method used for W^4 explained above is not sufficient for excluding all foreground objects. Fig. 1.(a) shows a frame with a high number of foregrounds objects during the training period, where foreground regions are shown and no detection is achieved. Consequently, the new background model will be wrong too. Fig. 1.(b) shows a wrong updated background model because of incorporated foreground objects in motion.

To solve the aforementioned problem, the first stage is applied in a recursive way. The median filter is applied until the standard deviation from the new background model is the same as the last background model. Now, the foreground objects are eliminated, as it can be seen in Fig. 1.(c), where foreground objects are detected in contrast to Fig. 1.(a). The background model is well updated as it can be seen in the Fig. 1.(d), and compared to Fig. 1.(b).

3 Foreground Region Detection

W⁴ uses a four-stage process to obtain a foreground object: thresholding, noise cleaning, morphological filtering, and object detection. The threshold stage classifies each pixel as either a background or a foreground pixel using the background model. A pixel is a foreground pixel if:

$$B(x) = \begin{cases} 0 & \text{background} \\ 1 & \text{foreground} \end{cases} \begin{cases} (I^t(x) + m(x)) > k_f * \max(d(x), dmin) \\ \wedge I^t(x) - n(x) < k_f * \max(d(x), dmin) \end{cases} \quad (1)$$

Parameter k_f serves for extend or reduce the detection range¹, and parameter $dmin$ is added to create a minimum background detection range. In this work, all the sequences are processed using only the first stage, i.e. thresholding, in order to evaluate in a better way the overall approach presented here.

4 Updating Background Model Parameters

The background model is updated using the *pixel-based update* and *object-based update* conditions as in W⁴. The first condition " $(gS(x) > k * N)$ " updates the background model periodically to adapt it to illumination changes in the background scene. And the second one " $(gS(x) < k * N \wedge mS(x) < r * N)$ " updates the background model to adapt it to physical changes in the background scene, when new objects are deposited or removed in the background scene.

W⁴ uses a detection support map (gS), to represent the number of times a pixel is classified as a background pixel:

$$gS(x, t) = \begin{cases} gS(x, t-1) + 1 & \text{if } x \text{ is background pixel} \\ gS(x, t-1) & \text{if } x \text{ is foreground pixel.} \end{cases} \quad (2)$$

A motion support map (mS) represents the number of times a pixel is classified as moving pixel:

$$mS(x, t) = \begin{cases} mS(x, t-1) + 1 & \text{if } M(x, t) = 1 \\ mS(x, t-1) & \text{if } M(x, t) = 0, \end{cases} \quad (3)$$

where $M(x, t)$ represents moving pixels, computed as:

$$M(x, t) = \begin{cases} 1 & \text{if } (|I(x, t) - I(x, t+1)| > 2 * \sigma) \wedge \\ & (|I(x, t-1) - I(x, t)| > 2 * \sigma) \\ 0 & \text{otherwise.} \end{cases} \quad (4)$$

The new background parameters $[m(x), n(x), d(x)]$ are updated after a predetermined number N of frames, and they are determined using the aforementioned maps as follows:

¹ The parameter k_f is set to 2, according to our experiments and the results presented in W⁴ [4].

$$[m(x), n(x), d(x)] = \begin{cases} [m^b(x), n^b(x), d^b(x)] & \text{if } (gS(x) > k * N) \text{ pixel-based update} \\ [m^f(x), n^f(x), d^f(x)] & \text{if } (gS(x) < k * N \wedge mS(x) < r * N) \\ & \text{object-based update} \\ [m^c(x), n^c(x), d^c(x)] & \text{otherwise,} \end{cases} \quad (5)$$

where k and r are typically 0.8 and 0.1, respectively [4]. The parameters $[m^b(x), n^b(x), d^b(x)]$ represent those pixels classified as background in this period of time, $[m^f(x), n^f(x), d^f(x)]$ those pixels classified as foreground pixels, and $[m^c(x), n^c(x), d^c(x)]$ are the value of the background parameters in the last background model. When the background model is updated, the maps are set to zero.

5 Improving Object-Based Update

Achieving a robust object-based update constitutes a challenging task thereby managing the incorporation of the new objects to the background, and removing the old background objects. The goal is to work with newly motionless foreground objects: detected objects in motion which have exhibited motion up to that moment. They should be identified, and the object-based update should take them into the background model. The first problem is that pixels which are no longer considered as motionless foreground pixels are updated as object-based, since the minimum number of times a particular pixel has been classified as foreground is usually not restrictive enough (according to object-based condition).

In addition, the foreground pixels considered to construct the background model $[m^f(x), n^f(x), d^f(x)]$ do not have to include foreground moving pixels, because these have different intensity values than foreground pixels considered object-based along the updating window.

Furthermore, other problems can be found with those pixels considered as object-based. If these pixels belong to a foreground motionless object which left the scene before the background is updated, such pixels can be included erroneously into the background model. This happens because these pixels also satisfy the object-based condition. Fig. 2.(a) shows a representative example of an background updating intensity value for a given pixel along the updating window with one foreground object which left the scene before the background is updated. Consequently, the pixel is added erroneously in the new background model.

Additionally, different foreground objects can appear at the same place in different times of the same background updating window, and they can be included together. Thus, the object-based parameters may be updated with a minimum intensity value $m(x)$ from one object and a maximum intensity value $n(x)$ from the other. Therefore, as both objects are different, the updating parameters $[m(x), n(x), d(x)]$ will be erroneous. Fig. 2.(b) illustrates the background updating when a new object appears at the same position where another object was before: these two objects will be incorporated into the background model, and the maximum and minimum intensity value from these two objects will wrongly

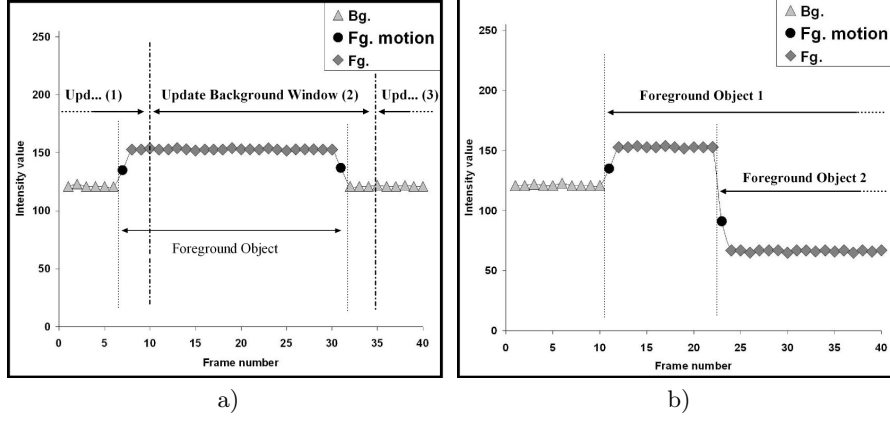


Fig. 2. Problems with object-based update. The pictures represent the background updating model for a given pixel along updating window, where: (a) a foreground object enters (background updating window 1) and leaves the scene before the background is updated (background updating window 2). Consequently, the pixel is incorporated erroneously in the new background model. (b) Background updating when a new object appears at the same position where another object was before: these two objects will be erroneously incorporated into the background model. See text for details.

constitute the new background model. The ghost which appears when an object belongs to the background awakes can present the similar problems above explained.

In order to solve the drawbacks explained above, object-based update is not performed when the pixels belongs to different foreground objects, or belongs to foreground objects that left the scene before the background is updated.

Our algorithm is based on the last detected object. In other words, the number of foreground pixels is computed from the latest foreground pixel in motion or background pixel. A new map is created, called *Foreground History Map*, $fS(x, t)$, which represents the number of times a pixel is detected as foreground continuously without pixels in motion, $M(x, t)$, neither background pixels during its history:

$$fS(x, t) = \begin{cases} fS(x, t-1) + 1 & \text{if } x \text{ is a foreground pixel and } M(x, t) = 0 \\ 0 & \text{otherwise.} \end{cases} \quad (6)$$

The Eq. (5) must be changed to include $fS(x, t)$ map instead of $gS(x, t)$. The minimum number of foreground pixels which are necessary for considering a pixel as object-based should be within the limits commented above. The foreground history map is more restrictive than gS map, and include this restriction. Furthermore, the use of $M(x, t)$ is avoided in the background updating parameters, because this restriction is already included inside $fS(x, t)$.

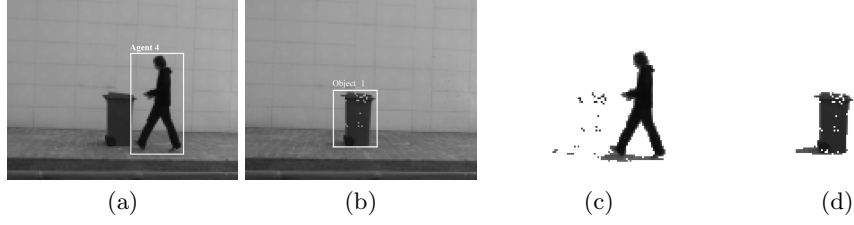


Fig. 3. Image sequence after background model updating. (a) Original sequence. (b) New background model with motionless foreground object (i.e. a rubbish bin). (c) Foreground detection without motionless foreground object. (d) Newly motionless foreground objects are added to the background model. Results are obtained without any kind of filtering.

The background updating parameters results as follows:

$$[m(x), n(x), d(x)] = \begin{cases} [m^b(x), n^b(x), d^b(x)] & \text{if } (gS(x) > k * N) \text{ pixel-based update} \\ [m^f(x), n^f(x), d^f(x)] & \text{if } (fS(x) > k * N) \text{ object-based update} \\ [m^c(x), n^c(x), d^c(x)] & \text{otherwise.} \end{cases} \quad (7)$$

With this approach, the problems mentioned above are solved. However, this means that foreground objects which cease their movements are no longer included into the background model. This happens since the pixels from those objects are often considered pixels in motion erroneously. The problem is that $M(x, t)$ map does not distinguish real motion from fluctuations.

A pixel in motion must show an evolution of its intensity value. Presently, $M(x, t)$ compare current intensity value for each pixel with its previous and its posterior intensity value. This can be enhance by comparing its previous value with its posterior one. The new $M(x, t)$ will be computed as follows:

$$M(x, t) = \begin{cases} 1 & \text{if } (|I(x, t) - I(x, t + 1)| > 2 * \sigma) \wedge \\ & (|I(x, t - 1) - I(x, t)| > 2 * \sigma) \wedge \\ & (|I(x, t + 1) - I(x, t - 1)| > 2 * \sigma) \\ 0 & \text{otherwise.} \end{cases} \quad (8)$$

Thus, the knowledge of motionless foreground objects is incorporated into the background model. Fig. 3.(b) shows a rubbish bin correctly updated as object-based. Furthermore, the problems with ghosts are also solved, see the rubbish bin ghost in Fig. 4.

6 Experimental Results

Our algorithm has been tested with multiple and different sequences which contain different motionless foreground objects and persons who interact with them. Fig.3 shows the scene where a rubbish bin is added to the background model. Fig.3.(a) shows the original image, Fig.3.(b) shows the background model where

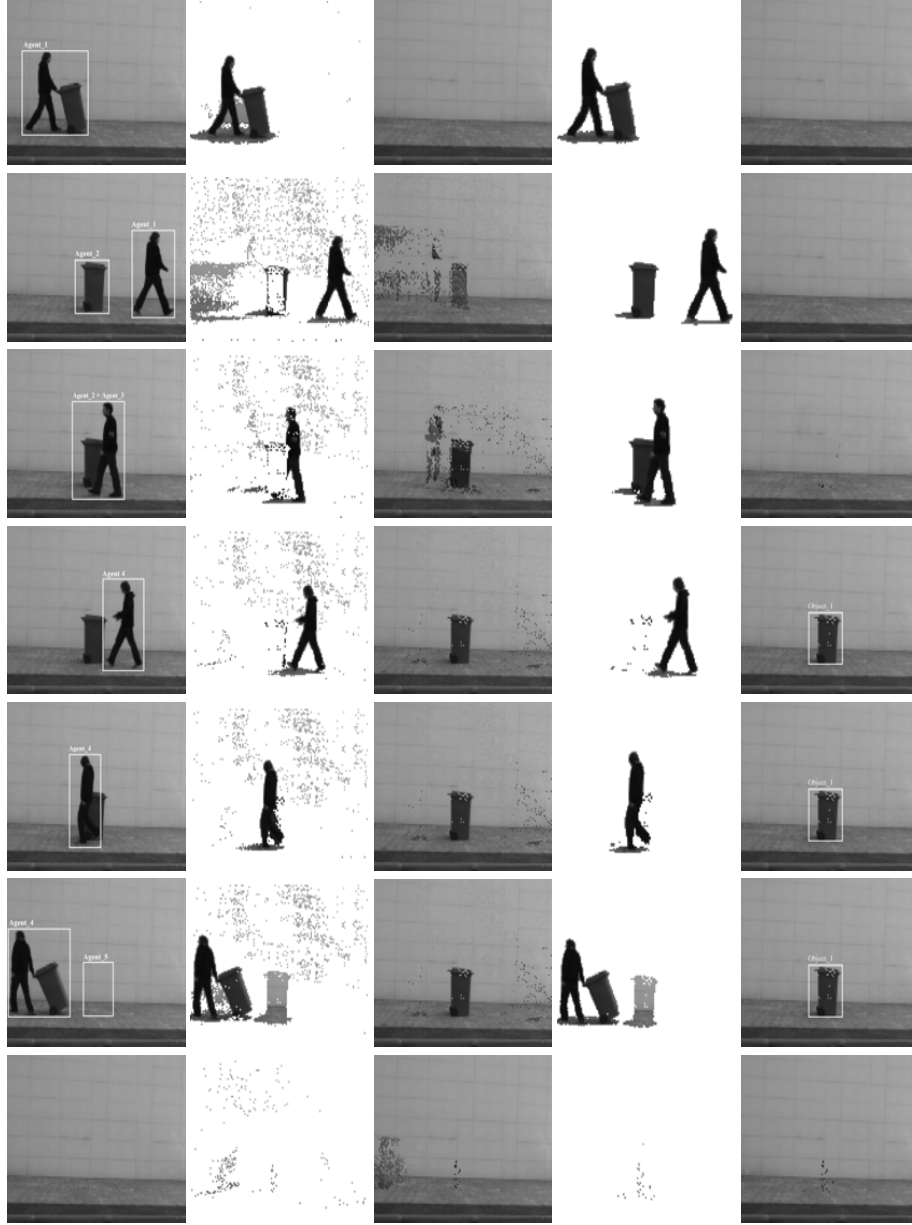


Fig. 4. First column shows the original image sequence, second column shows the results of foreground detection using W^4 , third column displays the background model using W^4 , fourth column shows foreground detection results using our approach and fifth column displays the background model using our approach. Image results are obtained without any kind of filtering. See text for details.

the motionless foreground object (i.e. a rubbish bin) is correctly added. Fig.3.(c) shows the foreground region detection without the motionless foreground object, and Fig.3.(d) shows the newly motionless foreground object which has been properly added to the background model following the object-based criterion. This newly motionless foreground object can then be used for further processing such as object classification or recognition.

Fig. 4 represents the same sequence using the same parameter values. This particular sequence contains 900 frames, and the first 200 frames are used to construct the initial background model, which do not contain any foreground object. The background model is updated every $N=100$ frames. The sequence corresponds to an agent who leaves a rubbish bin in the middle of the scene. Later on, a new agent enters into the scene. Subsequently, another agent takes the rubbish bin.

In Fig. 4, the first column shows the original image sequence at frame numbers 264, 372, 473, 612, 631, 682, 806. The second column shows the foreground region detection, and the third column shows the background model, and how it is updated using W^4 . Fourth and five columns show the foreground region detection and update background model using our approach. In those last two columns can be observed that the foreground person and the object are well segmented, and that the motionless foreground object is incorporated properly into the background model. The agent who passes in front of the incorporated object is also well segmented. After that, when this agent carries the object and leaves the scene, the ghost of this background object is solved and the background model is correctly updated.

7 Conclusions and Future Work

The proposed approach copes with (i) the non-incorporation of erroneous foreground objects to the background model, and (ii) the incorporation of motionless foreground objects. Pixels belonging to false foreground objects, foreground objects in motion, foreground objects that leaves the scene before the background is updated, and multiple foreground objects at the same time have been removed. Finally, a correct detection procedure of motionless foreground objects which have ceased their motion have been presented, and an efficient incorporation of such objects into the background model for a posterior processing have been proposed. Furthermore, the bootstrapping is solved even when many foreground objects are presented.

Future work needs to split the pixel-based and object-based update condition into two separate windows: problems corresponding to the first condition are detected earlier than those physical changes in the scene corresponding to the second one. That is to say, pixel-based update needs to be carried out more periodically than the object-based one. The approach copes with the physical changes in the scene, but the experimental results shows that it is necessary to improve the illumination-change modelling (i.e. the pixel-based update), and background in motion. Likewise, shadows are currently not handled, but these

can be eliminated by means of colour information [1,5]. The use of colour will also improve the detection of camouflage. Lastly, objects detected by object-based update should be part of a multilayer background model. In addition, an object appearance model is needed to cover situations involving crowds or multiple objects.

Acknowledgements

This work has been supported by EC grant IST-027110 for the HERMES project and by the Spanish MEC under projects TIC2003-08865 and DPI-2004-5414. Jordi González also acknowledges the support of a Juan de la Cierva Postdoctoral fellowship from the Spanish MEC.

References

1. A. Elgammal, D. Harwood, and L. S. Davis. Nonparametric background model for background subtraction. In *Proceedings European Conference Computer Vision (ECCV'00)*, pages 751–767, Dublin, 2000.
2. W.E.L. Grimson and C.Stauffer. Adaptive background mixture models for real-time tracking. volume 1, pages 22–29, 1999.
3. W.E.L. Grimson, C.Stauffer, and R.Romano. Using adaptive tracking to classify and monitor activities in a site. pages 22–29, 1998.
4. I. Haritaoglu, D. Harwood, and L.S. Davis. W4: Real-time surveillance of people and their activities. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(8):809–830, 2000.
5. T. Horprasert, D.Harwood, and L.S.Davis. A statistical approach for real-time robust background subtraction and shadow detection. *IEEE Frame-Rate Applications Workshop*, 1999.
6. H.W.S.Jabri, Z.Duric, and A.Rosenfeld. Detection and location of people in video images using adaptive fusion of color and edge information. volume 4, pages 627–630, September 2000.
7. Jordi González i Sabaté. *Human Sequence Evaluation: the Key-frame Approach*. PhD thesis, May 2004.
8. Mustafa Karaman, Lutz Goldmann, Da Yu, and Thomas Sikora. Comparison of static background segmentation methods. In *Visual Communications and Image Processing (VCIP '05)*, July 2005.
9. K.toyama, J.Krumm, B.Brumitt, and B.Meyers. Wallflower: Principles and practice of background maintenance. volume 1, pages 255–261, 1999.
10. H.-H. Nagel. Steps toward a cognitive vision system. *AI Magazine, Cognitive Vision*, 25(2):31–50, 2004.
11. J. Shen. Motion detection in color image sequence and shadow elimination. *Visual Communications and Image Processing*, 5308:731–740, January 2004.
12. L. Wang, W. Hu, and T. Tan. Recent developments in human motion analysis. *Pattern Recognition*, 36(3):585–601, 2003.
13. C.R. Wren, A. Azarbayejani, T. Darrell, and A.P. Pentland. Pfunder: Real-time tracking of the human body. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(7):780–785, 1997.