AN EFFICIENT CHANGE DETECTION ALGORITHM BASED ON A STATISTICAL NON-PARAMETRIC CAMERA NOISE MODEL

Alessandro Bevilacqua, Luigi Di Stefano and Alessandro Lanza

University of Bologna
Department of Electronics, Computer Science and Systems (DEIS)
Advanced Research Centre "Ercole De Castro" (ARCES)
Via Toffano, 2/2
40125 Bologna, ITALY
{abevilacqua, ldistefano, alanza}@arces.unibo.it

ABSTRACT

In this paper we present a change detection algorithm for grey level sequences based on the background subtraction technique, which achieves a good trade-off between time performance and detection quality. The basic idea consists in separating the background process into a deterministic background process and a stochastic camera noise process. The assumption that statistics of the camera noise for a pixel only depends on its current grey level allows to infer a non-parametric statistical camera noise model once and for all arising from a short bootstrap sequence. Hence, 256 couples of lower and upper deterministic thresholds are extracted, to be used in the background subtraction step. While the deterministic nature of the background model as well as of the thresholds lead to an efficient algorithm, utilising 256 couples of different thresholds results in a very sensitive detection. Experimental results allow to assess both the efficiency and the effectiveness of the method we devised.

1. INTRODUCTION

Real-time extraction of moving objects from video sequences represents a crucial step in many computer vision applications. Background subtraction is one of the most common methods used to detect motion in case of sequences coming from a stationary camera. This method relies on the feasibility to have a reliable background at one's disposal along the whole processing stage. The background needs to be maintained and many approaches exist to face this problem: they differ from each other basically for the type of background model used and for the procedure used to generate and update the background itself. Generally speaking, the more precise a method, the lower the efficiency.

In this paper we present a change detection algorithm based on the background subtraction technique, which achieves a good trade-off between time performance and quality of the detection. The basic idea consists in separating the background process into two distinct processes. The first is deterministic and represents the background as if it were measured by an ideal noiseless camera. The second is stochastic and measures the camera noise. A short bootstrap sequence where moving objects can also be present is used to infer for each pixel ensemble statistics of the random process representing the camera noise. The assumption that statistics of the camera noise for a pixel only depends on its current grey level allows to extract a non-parametric *statistical* noise model for the camera used to monitor the scene. Subsequently, we build also a *deterministic* model by extracting a lower and an upper percentile from the noise distribution of each grey level. Finally, these percentiles are used to determine 256 (one for each grey level) couples of lower and upper thresholds, to be used in the background subtraction step. Therefore, these deterministic thresholds lead to both an efficient and an accurate change detection algorithm.

This paper is organised as follows. In Section 2 some previous works are discussed. Section 3 outlines the probabilistic framework which is behind our approach. The algorithm conceived to generate the camera noise model is described in detail in Section 4. Section 5 depicts the background subtraction step. Experimental results are discussed in Section 6 and Section 7 draws conclusions.

2. PREVIOUS WORKS

In [1] the authors model the background process for each pixel as a unique spatially independent stochastic gaussian process. The parameters of the gaussian distribution representing the ensemble pdf for each pixel are initialised through a bootstrap sequence free of moving objects. While the mean is recursively updated using a simple adaptive filter, the covariance matrix is extracted once and for all, thus yielding a threshold that does not adapt to scene changes.

0-7803-8554-3/04/$20.00 ©2004 IEEE.

Authors in [2] model the pixel process instead of the background process only: a spatially independent random process is used for each pixel, representing both the background and the foreground processes due to moving objects and to cast shadows possibly covering the pixel. A weighted sum of three gaussian distributions (background, moving objects and shadow distributions) is used to model the ensemble pdf for each pixel. Nevertheless, the background is still represented by a unique gaussian random process for each pixel. Background subtraction consists in choosing for each pixel which of the three classes has the highest a posteriori probability. An incremental EM algorithm is used to both learn and update the distribution parameters.

A generalization of the previous approach is presented in [3]. Each pixel is still modelled as a spatially independent stochastic process having a mixture of K (a small number from 3 to 5) gaussian distributions as ensemble pdf. At each time step and for each pixel, the distributions are ordered according to the value of a ratio attained dividing the evidence of the distribution by its variance. The first B distributions are selected to represent the background process and if the pixel value is not represented by any of these distributions it is classified as moving. The parameters of the mixture are updated by means of a simple adaptive filter.

In [4] authors improves the method outlined in [3]. In particular, they present a different approach for initialising and for updating the parameters of the mixture model, based on an incremental EM algorithm.

A further generalization of the previous approaches is outlined in [5]. The ensemble pdf of the spatially independent stochastic process of each pixel is modelled in a non-parametric manner. At each time step and for each pixel the ensemble pdf is non-parametrically estimated by means of a gaussian kernel estimator function applied to a window of recent sample intensities for that pixel. The model update consists in simply shifting the samples window.

Even though the methods described in [2]-[5] model the background more and more accurately, their complexity make them not suitable to be used efficiently in many real-time applications.

3. PROBABILISTIC FRAMEWORK

Let us consider the scalar values $p_{i,j}(t)$ that a pixel (i,j) assumes during all the 8-bit grey level frames of a time interval I. Then we define a *time series* $S^I_{i,j}$ as follows:

$$S^I_{i,j} = \{p_{i,j}(t) : t \in I\} \tag{1}$$

Now let us define the *relative temporal histogram* $h^I_{i,j}(v)$ and the *absolute temporal histogram* $H^I_{i,j}(v)$ as the relative and the absolute frequency of the values $v \in [0; 255]$ the pixel (i,j) assumes. $\mu^I_{i,j}$ and $med^I_{i,j}$ represent the *temporal mean* and the *temporal median*, respectively.

Using the terms of Mathematical Statistics, the values that a pixel may assume over time can be considered as a one-sided discrete time scalar stochastic process called *pixel stochastic process* $(P_{i,j}(t))$. Therefore, a time series $S^I_{i,j}$ represents a realization of the underlying random process $P_{i,j}(t)$. A pixel stochastic process is characterized by ensemble statistics, such as the *ensemble probability density function* $pdf_{i,j}(t,v)$, the *ensemble mean* $\mu_{i,j}(t)$ and the *ensemble median* $med_{i,j}(t)$.

Let us now consider a stationary background pixel (i,j), by assuming to this purpose that the lighting changes and also the background motion (e.g., swaying trees) be negligible (the pixel measures a constant radiance). The pixel (i,j) stochastic process $P^B_{i,j}(t)$ can be modelled as the sum of two distinct processes:

$$P^B_{i,j}(t) = B_{i,j}(t) + N_{i,j}(t) = B_{i,j} + N_{i,j}(t) \tag{2}$$

where $B_{i,j}$ is a deterministic constant process, giving the value of the background pixel as if it were measured by an ideal noiseless camera, and $N_{i,j}(t)$ is a stochastic process representing the camera noise (*CN*) affecting the stationary pixel. Besides, as for $N_{i,j}(t)$ in case of a pixel measuring a constant radiance we claim three hypotheses:

- Hyp. 1: $N_{i,j}(t)$ is a scalar random process, that is any spatial statistical dependence is neglected;

- Hyp. 2: $N_{i,j}(t)$ is a stationary and ergodic stochastic process (briefly, a *SESP*);

- Hyp. 2: $N_{i,j}(t)$ statistical properties only depends on $B_{i,j}$, that is on the pixel (i,j) deterministic noiseless value.

As for Hyp. 1, it is worth noticing that as a matter of fact the CN for a pixel is statistically correlated with the noise affecting the pixel's neighbours. We have chosen to neglect this correlation in order to keep the model as simple as possible. Hyp. 2 and Hyp. 3 have been suggested by the extensive experiments carried out with different cameras. Based on the three hypotheses, for N-bit grey level stationary sequences the CN can be modelled by means of 2^N (Hyp. 3) scalar (Hyp. 1) SESP (Hyp. 2), $N_v(t)$, one for each possible grey value $v \in [0; 2^N - 1]$. Hence Expr. 2 becomes:

$$P^B_{i,j}(t) = B_{i,j} + N_{v=B_{i,j}}(t) \tag{3}$$

Since a SESP is completely defined by its ensemble pdf and we cope with 8-bit grey level sequences, the CN model consists of 256 ensemble probability density functions pdf^N_v, where $v \in [0; 255]$. As for Hyp. 2, stationary (*stat*) means that ensemble statistics are constant over time while ergodic (*erg*) means that ensemble statistics are approximately equivalent to the temporal ones if the cardinality of the samples set $card(I)$ is greater than a certain value $card_{erg}$. Hence, for $N_v(t)$ and for a SESP in general:

$$\begin{cases} pdf_{i,j}(t,v) \stackrel{stat}{=} pdf_{i,j}(v) \stackrel{erg}{\simeq} h^I_{i,j}(v) \\ \mu_{i,j}(t) \stackrel{stat}{=} \mu_{i,j} \stackrel{erg}{\simeq} \mu^I_{i,j} \\ med_{i,j}(t) \stackrel{stat}{=} med_{i,j} \stackrel{erg}{\simeq} med^I_{i,j} \end{cases} \tag{4}$$

Let us now look at Eq. 3: since both $N_v(t)$ and $B_{i,j}$ (it is a deterministic and constant process) are SESP, we can state that the stochastic process $P^B_{i,j}(t)$ for a stationary background pixel is a SESP as well, thus satisfying Expr. 4. As for $card_{erg}$, experiments pointed out that incrementally computing the temporal relative histogram for a stationary background pixel, the shape of the histogram did not change significantly after about 25 sample intensities. Therefore we have chosen $card_{erg} = 25$.

4. A NON-PARAMETRIC NOISE MODEL

The non-parametric statistical CN model is generated by means of a two-stage algorithm, using a bootstrap sequence of few seconds in which moving objects can also be present. While the first stage generates a *rough* background, the second stage uses that background to identify *reliable* background areas where extract the CN model from. As well as other initialization methods ([6], [7]) our algorithm relies on a background that must be stationary along the bootstrap sequence. We divide the bootstrap sequence into two consecutive time intervals: $I^1 = [t_0; t_1]$, $I^2 = (t_1; t_2]$.

Stage 1. A temporary rough background is generated utilising "blind" pixel temporal statistics computed during I^1. In particular, we vote the temporal median $med^{I^1}_{i,j}$ as the value of the background for each pixel:

$$B_{i,j} = med^{I^1}_{i,j} \tag{5}$$

Stage 2. A "background detection" is performed for each frame in I^2: the subtraction between the current frame and the rough background of Stage 1 is computed to identify *reliable* background regions. To this purpose, the outcome of the subtraction is thresholded by a low value thus allowing to restrict the background areas to regions really not moving. For each pixel (i,j) we build a set $I^2_{i,j} \subseteq I^2$, containing only the frames in which the pixel has been detected as belonging to the background. Hence, the time series $S^{I^2_{i,j}}_{i,j}$ are "selective", because they retain information just about the background. Besides, since the background is assumed to be stationary along the whole bootstrap sequence, these time series represent a part of the realization of the SESP $P^B_{i,j}(t)$ (Expr. 2) for each pixel. As a consequence, Expr. 4 holds if we replace I with $I^2_{i,j}$. Since for a given probability distribution the mean is the value that minimises the expected prediction error, we vote the "selective" temporal mean $\mu^{I^2_{i,j}}_{i,j}$ as the final good background value for each pixel:

$$B_{i,j} = \mu_{i,j} = \mu^{I^2_{i,j}}_{i,j} \tag{6}$$

By using the selective temporal statistics computed along I^2 and by exploiting the hypotheses claimed in Section 3, we extract the non-parametric statistical model of the CN. Computing $N_{i,j}(t)$ from Expr. 2 yields Eq. 7:

$$N_{i,j}(t) = P^B_{i,j}(t) - B_{i,j} \tag{7}$$

Hence, from $S^{I^2_{i,j}}_{i,j}$ the time series of the CN values for the pixel (i,j) can be deducted by simply subtracting the background value of Expr. 6. Following from Eq. 3 and Eq. 7:

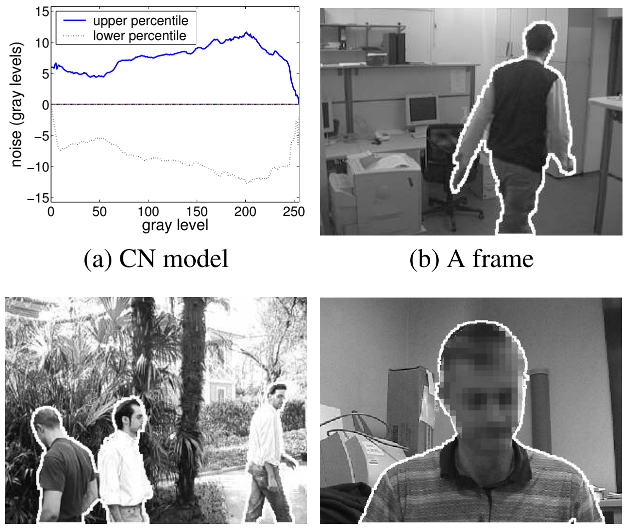
$$N_{v=B_{i,j}}(t) = P^B_{i,j} - B_{i,j} \tag{8}$$

The time series of the CN values for a pixel (i,j) can thus be considered not just a realization of the random process $N_{i,j}(t)$ representing the CN for that pixel, but also a realization of the more general stochastic process $N_{v=B_{i,j}}$ representing the CN for the grey value $v = B_{i,j}$. Therefore, time series of the CN values for the pixels which have had the same background value according to Expr. 6 represent different realizations of the same random process. Now we can build a unique non-parametric ensemble pdf (that is a relative histogram) for each grey level CN, thus attaining the statistical CN model. By extracting a lower and an upper percentile from each grey level CN pdf, we build the deterministic CN model. Since some grey levels may exist which are characterized by few or even no sample pixels, in those cases we perform a linear interpolation of the CN model. In order to obtain a model that is less sensitive to illumination changes, we artificially scale all the percentiles just identified by a unique factor greater than one. We thus attain the 256 couples of thresholds ($ts_{inf}(v)$ and $ts_{sup}(v)$), one for each grey level $v \in [0; 255]$, to be used in the background difference. In this way we retain both the advantages arising from the simplicity of setting up a unique threshold and the effectiveness of 256 different couples of thresholds. This results in an effective yet efficient thresholding operation.

5. BACKGROUND SUBTRACTION AND UPDATE

Due to the deterministic nature and to the simplicity of the background and of the threshold models, the background subtraction results to be very efficient. For each pixel, the algebraic difference between the current frame $F_{i,j}$ and the background $B_{i,j}$ is computed. The outcome is then compared with the couple of thresholds $ts_{inf}(v)$ and $ts_{sup}(v)$, depending on the current background value $B_{i,j}$, thus attaining a binary image $M_{i,j}$ representing the moving pixels:

$$M_{i,j}(t) = \begin{cases} 1 & \text{if} \quad F_{i,j}(t) - B_{i,j}(t) \notin A_{i,j}(t) \\ 0 & \text{if} \quad F_{i,j}(t) - B_{i,j}(t) \in A_{i,j}(t) \end{cases} \tag{9}$$



(a) CN model (b) A frame

(c) A frame (d) A frame

Fig. 1. The detection results.

where $A_{i,j} = [ts_{inf}(B_{i,j}(t)); ts_{sup}(B_{i,j}(t))]$.

The deterministic background is updated by a simple and efficient adaptive recursive filter:

$$B_{i,j}(t+1) = (1-\alpha)B_{i,j}(t) + \alpha F_{i,j}(t) \qquad (10)$$

where $\alpha \in [0; 1]$ represents the adaptation rate.

6. EXPERIMENTAL RESULTS

Tests were performed on several 8-bit grey level sequences representing typical surveillance scenes, taken by a single stationary CCD camera and sampled at 25 Hz at a resolution of 320x240. The target PC is an AMD Athlon MP 1800+, 1 GB RAM. The experimental results we accomplished assess both the efficiency and the effectiveness of our algorithm. As for the effectiveness, we stated that our method acts as we apply 256 couples of different thresholds, one for each grey level. As one could infer, it is impracticable such an experiment, therefore we will assess the effectiveness by showing the capability of our algorithm to detect moving pixels in situation of camouflage between the background and the moving objects. We present the results of our algorithm run on three challenging sequences. Fig. 1(a) shows the deterministic CN model extracted for the CCD camera we used in the experiments. It is worth noticing how the percentiles (the noise level) vary significantly by the grey level. Fig. 1(b-d) depicts the detection results by showing the boundaries of the detected blobs for a sample frame of each test sequence.

As regards time performance, our method reveals to be very efficient, working off-line at 40 fps.

7. CONCLUSIONS AND FUTURE WORKS

In this paper we have presented a change detection algorithm for 8-bit grey level sequences, that achieves a good trade-off between efficiency and quality of the detection. The choice to model both the background and the thresholds to be applied in the background subtraction step in a deterministic manner leads to an efficient algorithm. An accurate detection is attained by automatically extracting a statistical non-parametric model for the noise of the camera used to capture the sequence. Experimental results show how the proposed method succeeds in detecting moving blobs even in the challenging case of camouflage. The periodic update of the camera noise model and the extension of the algorithm to colour sequences are matters being studied.

8. REFERENCES

[1] C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland, “Pfinder: Real-time tracking of the human body,” in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, July 1997, vol. 19, pp. 780–785.

[2] N. Friedman and S. Russell, “Image segmentation in video sequences: A probabilistic approach,” in *Annual Conference on Uncertainty in Artificial Intelligence*, 1997, pp. 175–181.

[3] C. Stauffer and W. E. L. Grimson, “Adaptive background mixture models for real-time tracking,” in *Proceedings of 1999 IEEE International Conference on Computer Vision and Pattern Recognition (CVPR99)*, June 1999, vol. 2, pp. 246–252.

[4] P. KaewTraKulPong and R. Bowden, “An improved adaptive background mixture model for real-time tracking with shadow detection,” in *Proceedings of* 2^{nd} *European Workshop on Adavanced Video Based Surveillance Systems (AVBS01)*, September 2001.

[5] A. Elgammal, D. Harwood, and L. Davis, “Non-parametric model for background subtraction,” in *Proceedings of 1999 IEEE International Conference on Computer Vision (ICCV99) FRAME-RATE workshop*, September 1999.

[6] D. Gutchess, M. Trajkovic, E. Cohen-Solal, D. Lyons, and A. K. Jain, “A background model initialization algorithm for video surveillance,” in *Proceedings of the* 8^{th} *International Conference on Computer Vision (ICCV01)*, July 2001, vol. 1, pp. 733–740.

[7] A. Bevilacqua, “A novel background initialization method in visual surveillance,” in *IAPR Workshop on Machine Vision Applications (MVA02)*, December 2002, pp. 614–617.