8th IEEE International Conference on Advanced Video and Signal-Based Surveillance, 2011

# Background Subtraction by Non-parametric Probabilistic Clustering

Alessandro Lanza Samuele Salti Luigi Di Stefano
DEIS, University of Bologna
Viale Risorgimento 2, 40136 Bologna, Italy

## Abstract

*We present a background subtraction approach aimed at efficiency and robustness to common source of disturbance such as gradual and sudden illumination changes, camera gain and exposure variations, noise. At each new frame, a non-parametric mixture-based probabilistic clustering is performed to segment the image into changed and unchanged pixels with respect to a fixed background. A two-components mixture, a two-dimensional discrete feature space, a non-parametric model for the components likelihood and a proper initial guess are the key ingredients of this novel algorithm that, besides dealing effectively with the discrimination of photometric and semantic changes, exhibits very high computational efficiency. Experiments are presented, proving the achieved state-of-the-art robustness-efficiency trade-off.*

## 1. Introduction

The main difficulty with background subtraction consists in discerning efficiently and effectively semantic changes of the monitored scene due to foreground objects in presence of spurious intensity variations yielded by disturbs such as noise, gradual or sudden illumination changes (e.g. due to the time of the day or a light switch), dynamic adjustments of camera parameters (e.g. auto-exposure, auto-gain), persistent background motion (e.g. waving trees). Many different algorithms for dealing with these issues have been proposed (see [3] for a recent survey).

A first class of popular algorithms based on statistical, time-adaptive, per-pixel background models, such as e.g. Mixture of Gaussians [10] or kernel-based non-parametric models [2], are effective only when applied to sequences acquired at high frame-rates and only in case of noise, gradual illumination changes and persistent background motion. Unfortunately, they are inherently unable to deal with those disturbs causing sudden intensity variations, yielding in such cases lots of false positives.

A second class of algorithms relies on a-priori modeling the possible intensity changes yielded by disturbs over small image patches with respect to a fixed background. Following this idea, a pixel from the current frame is classified as changed if the intensity transformation between its local neighborhood and the corresponding neighborhood in the background can not be explained by the chosen a-priori model. As a result, gradual as well as sudden photometric distortions can be dealt with effectively provided that they are explained by the model. Thus, the main issue concerns the choice of the model: in principle, the more restrictive such a model, the higher is the ability to detect changes (sensitivity) but the lower is robustness to disturbs (specificity). Some proposals assume disturbs to yield linear or affine intensity transformations [6, 8]. Nevertheless, as discussed in [11], many non-linearities may arise in the imaging process, so that a less constrained model is often required to achieve adequate robustness. Hence, other algorithms adopt order-preserving models, i.e. assume monotonic non-decreasing intensity transformations [4, 5, 7, 11].

We propose a background subtraction approach that, instead of assuming a-priori a neighborhood-wise model for changes caused by disturbs, estimates on-line, i.e at each new frame, a frame-wise model for changes yielded by both disturbs and foreground objects in the form of a two-components mixture of discrete distributions. Once the mixture has been estimated, the probability for each pixel to be changed is obtained as the posterior probability for the pixel to belong to the foreground objects component. On-line frame-by-frame estimation of the mixture holds the potential for deploying models of intensity variations as restrictive as needed to discriminate between the two classes, so that the algorithm can exhibit a high sensitivity without a significant loss of specificity.

## 2. Non-parametric mixture-based clustering

By taking the $N$ pixels in lexicographical order, let us denote the sensed integer intensities of the gray level background and current frame to be compared, respectively, as

$$\boldsymbol{x} = (x_1, \ldots, x_N) \quad \text{and} \quad \boldsymbol{y} = (y_1, \ldots, y_N) \qquad (1)$$

with $x_i, y_i \in [0, 255]$ for 8-bit images, $i = 1, \ldots, N$. Given $\boldsymbol{x}$ and $\boldsymbol{y}$, the goal of a background subtraction al-

978-1-4577-0845-9/11/$26.00 

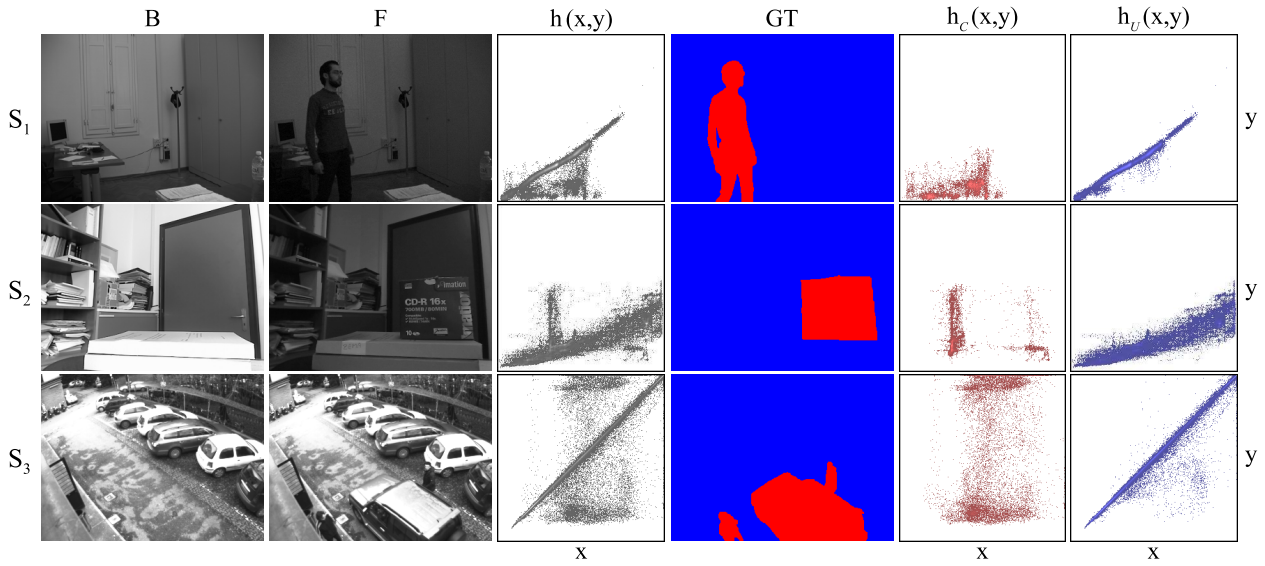



Figure 1. Some samples providing empirical evidence supporting our formulation of the background subtraction problem.

gorithm is to compute the binary change mask

$$\boldsymbol{c} = (c_1, \ldots, c_N) \tag{2}$$

i.e. to classify each pixel $i$ into one of the two classes:

$c_i = \mathcal{C}$: the pixel is sensing a scene change;
$c_i = \mathcal{U}$: the pixel is not sensing a scene change.

The main ideas at the basis of our proposal can be summarized as follows:

a) the background subtraction problem is casted as a pixels clustering problem;
b) the pair of background-frame intensities sensed at a pixel is used as feature vector, so that clustering is performed in the discrete 2-D feature space $[0,255]^2 \subset \mathbb{N}^2$;
c) the clustering problem is solved by a probabilistic mixture-based approach, thus allowing for soft-labeling of pixels in terms of probabilities;
d) a mixture of only two components is considered (changed and unchanged pixels), so as to avoid the subtle problem of components number estimation;
e) a non-parametric model for the two bivariate likelihoods based on discrete kernel estimation is assumed, thus allowing for arbitrarily shaped 2-D clusters;
f) a (rectangular) box-kernel is deployed so as to allow for very efficient likelihoods estimation;

According to this framework, the sensed pairs of background-frame intensities $(x_i, y_i)_{\,i=1,\ldots,N}$ are regarded as $N$ independent realizations of the 2-D discrete random vector $(X, Y)$ having support $[0,255]^2 \subset \mathbb{N}^2$ and distributed according to the following two-components mixture of probability mass functions:

$$P(x, y \,|\, \boldsymbol{\theta}) = \pi_{\mathcal{C}}\, P_{\mathcal{C}}(x, y \,|\, \boldsymbol{\theta}_{\mathcal{C}}) + \pi_{\mathcal{U}}\, P_{\mathcal{U}}(x, y \,|\, \boldsymbol{\theta}_{\mathcal{U}}) \tag{3}$$

where $P(x, y|\,\cdot)$ stands for $P(X = x, Y = y \,|\, \cdot)$, $\pi_{\mathcal{C}}$ and $\pi_{\mathcal{U}} = 1 - \pi_{\mathcal{C}}$ are the *mixing probabilities*, i.e. the prior probabilities for a pixel to belong to the changed or unchanged cluster, $\boldsymbol{\theta}_{\mathcal{C}} = (b_{\mathcal{C}}^{(x)}, b_{\mathcal{C}}^{(y)})$ and $\boldsymbol{\theta}_{\mathcal{U}} = (b_{\mathcal{U}}^{(x)}, b_{\mathcal{U}}^{(y)})$ are the parameters of the two likelihoods, i.e. the four bandwidths defining the two box-kernels, and $\boldsymbol{\theta} = (\pi_{\mathcal{C}}, \pi_{\mathcal{U}}, \boldsymbol{\theta}_{\mathcal{C}}, \boldsymbol{\theta}_{\mathcal{U}})$ is the entire vector of six parameters defining the mixture.

Evidence for the validity of this formulation for background subtraction is provided in Fig. 1. The figure depicts: the background and a sample frame from three different sequences (first and second column), the features extracted from the entire data set, i.e. the background-frame pairs of intensities for all pixels $(x_i, y_i)_{\,i=1,\ldots,N}$ in the form of a joint histogram $h(x, y)$ (third column), the binary ground truth mask (fourth column) and the features partitioned into the changed and unchanged clusters according to the ground truth, $h_{\mathcal{C}}(x, y)$ and $h_{\mathcal{U}}(x, y)$ (fifth and sixth column).

Once the mixture has been identified, a soft-labeling of the current frame can be carried out by computing the posterior probability for each pixel $i$ to belong to the changed pixels component given the observed feature vector $(x_i, y_i)$, so that the final change mask can be obtained as follows:

$$P(c_i{=}\mathcal{C} \,|\, x_i, y_i) = \frac{\pi_{\mathcal{C}}\, P_{\mathcal{C}}(x_i, y_i \,|\, \boldsymbol{\theta}_{\mathcal{C}})}{P\,(x_i, y_i \,|\, \boldsymbol{\theta})} \begin{matrix} c_i = \mathcal{C} \\ \geqslant \\ \leqslant \\ c_i = \mathcal{U} \end{matrix} T \tag{4}$$

where $T \in [0, 1] \subset \mathbf{R}$ is a fixed threshold.

## 2.1. Mixture identification

In order to identify the mixture, we start from performing a preliminary hard clustering of the pixels, i.e. from computing a preliminary binary change mask, by means of a not very accurate but extremely efficient neighborhood-based

| | | |
|---|---|---|
| $x_{i,1}$ | $x_{i,2}$ | $x_{i,3}$ |
| $x_{i,4}$ | $x_i$ | $x_{i,5}$ |
| $x_{i,6}$ | $x_{i,7}$ | $x_{i,8}$ |

| | | |
|---|---|---|
| $y_{i,1}$ | $y_{i,2}$ | $y_{i,3}$ |
| $y_{i,4}$ | $y_i$ | $y_{i,5}$ |
| $y_{i,6}$ | $y_{i,7}$ | $y_{i,8}$ |

Figure 2. Notations for the background (left) and the current frame (right) neighborhood intensities in case of a $3 \times 3$ neighborhood.

background subtraction algorithm. Like [4, 5, 7, 11], the algorithm is robust to sudden photometric changes thanks to the implicit assumption of an order-preserving model for intensity variations yielded by disturbs. For a generic pixel $i$, let the intensities of a surrounding $n \times n$ neighborhood be denoted as in Fig. 2, let the intensity differences between the $j$-th and the central pixel of the neighborhood in the background and in the current frame be, respectively,

$$d_{i,j}^{(x)} = x_{i,j} - x_i \quad \text{and} \quad d_{i,j}^{(y)} = y_{i,j} - y_i \tag{5}$$

and let the pixel in the neighborhood yielding the maximum absolute value of the background intensity difference be

$$\bar{j}_i = \underset{j=1,\ldots,n^2-1}{\operatorname{argmax}} \left| d_{i,j}^{(x)} \right| \tag{6}$$

The preliminary change mask $\tilde{\boldsymbol{c}} = (\tilde{c}_i, \ldots, \tilde{c}_N)$ is computed by classifying each pixel $i$ as changed if the intensity differences $d_{i,\bar{j}_i}^{(x)}$ and $d_{i,\bar{j}_i}^{(y)}$ have opposite signs and not negligible magnitudes, unchanged otherwise:

$$d_{i,\bar{j}_i}^{(x)} \cdot d_{i,\bar{j}_i}^{(y)} \begin{array}{c} \tilde{c}_i = \mathcal{C} \\ \lessgtr \\ \tilde{c}_i = \mathcal{U} \end{array} \tau \tag{7}$$

where $\tau$ is a negative integer threshold. This algorithm is a simplified version of that proposed in [11] and exhibits $O(N)$ complexity. In particular, the complexity does not depend on the neighborhood size $n$. In fact, since the background is not updated, the neighborhood index $\bar{j}_i$ for each pixel can be computed off-line by (6) after background initialization and stored to be used on-line at each frame.

The obtained change mask is thus used to estimate the mixing probabilities $\pi_{\mathcal{C}}$, $\pi_{\mathcal{U}}$ and the joint histograms of background-frame intensities at a pixel $h_{\mathcal{C}}(x, y)$, $h_{\mathcal{U}}(x, y)$:

$$\hat{\pi}_{\mathcal{C}} = \frac{N_{\mathcal{C}}}{N}, \quad \hat{h}_{\mathcal{C}}(x, y) = \frac{\sum_{i=1}^{N} I_{\mathcal{C}}(\tilde{c}_i)\delta_{x-x_i, y-y_i}}{N_{\mathcal{C}}} \tag{8}$$

where $N_{\mathcal{C}}$ is the number of pixels labeled as changed in the preliminary change mask, $I_{\mathcal{C}}(\tilde{c}_i) = 1$ iff $\tilde{c}_i = \mathcal{C}$ is an indicator function, $\delta_{i,j}$ is the 2-D Kronecker delta and $\hat{\pi}_{\mathcal{U}}$, $\hat{h}_{\mathcal{U}}(x, y)$ are similarly defined.

To complete the mixture identification, we have to estimate the two likelihoods $\hat{P}_{\mathcal{C}}(x, y \,|\, \boldsymbol{\theta}_{\mathcal{C}})$ and $\hat{P}_{\mathcal{U}}(x, y \,|\, \boldsymbol{\theta}_{\mathcal{U}})$ starting, respectively, from the two just computed histograms $\hat{h}_{\mathcal{C}}(x, y)$ and $\hat{h}_{\mathcal{U}}(x, y)$. Since the deployed kernel density estimation procedure is the same for the two likelihoods, for simplicity we focus here on the changed pixels component. To estimate the likelihood parameters $\boldsymbol{\theta}_{\mathcal{C}} = (b_{\mathcal{C}}^{(x)}, b_{\mathcal{C}}^{(y)})$, i.e. the two bandwidths univocally defining the box-kernel, first we marginalize the 2-D histogram $\hat{h}_{\mathcal{C}}(x, y)$ so as to obtain the two 1-D marginal histograms:

$$\hat{h}_{\mathcal{C}}^{(x)}(x) = \sum_{y=1}^{255} \hat{h}_{\mathcal{C}}(x, y), \quad \hat{h}_{\mathcal{C}}^{(y)}(y) = \sum_{x=1}^{255} \hat{h}_{\mathcal{C}}(x, y) \tag{9}$$

then we estimate the two bandwidths by applying the Silverman's rule for bandwidth selection in 1-D kernel density estimation [9] to each of the 1-D marginal histograms:

$$\hat{b}_{\mathcal{C}}^{(x)} = 1.06 \cdot N_{\mathcal{C}}^{-1/5} \cdot \hat{\sigma}_{\mathcal{C}}^{(x)}, \;\; \hat{b}_{\mathcal{C}}^{(y)} = 1.06 \cdot N_{\mathcal{C}}^{-1/5} \cdot \hat{\sigma}_{\mathcal{C}}^{(y)} \tag{10}$$

where $\hat{\sigma}_{\mathcal{C}}^{(x)}$ and $\hat{\sigma}_{\mathcal{C}}^{(y)}$ are the standard deviations of the 1-D marginal histograms $\hat{h}_{\mathcal{C}}^{(x)}$ and $\hat{h}_{\mathcal{C}}^{(y)}$, respectively. Finally, the component likelihood is obtained by discrete kernel estimation with a box-kernel characterized by a rectangular support with dimensions defined by the estimated bandwidths:

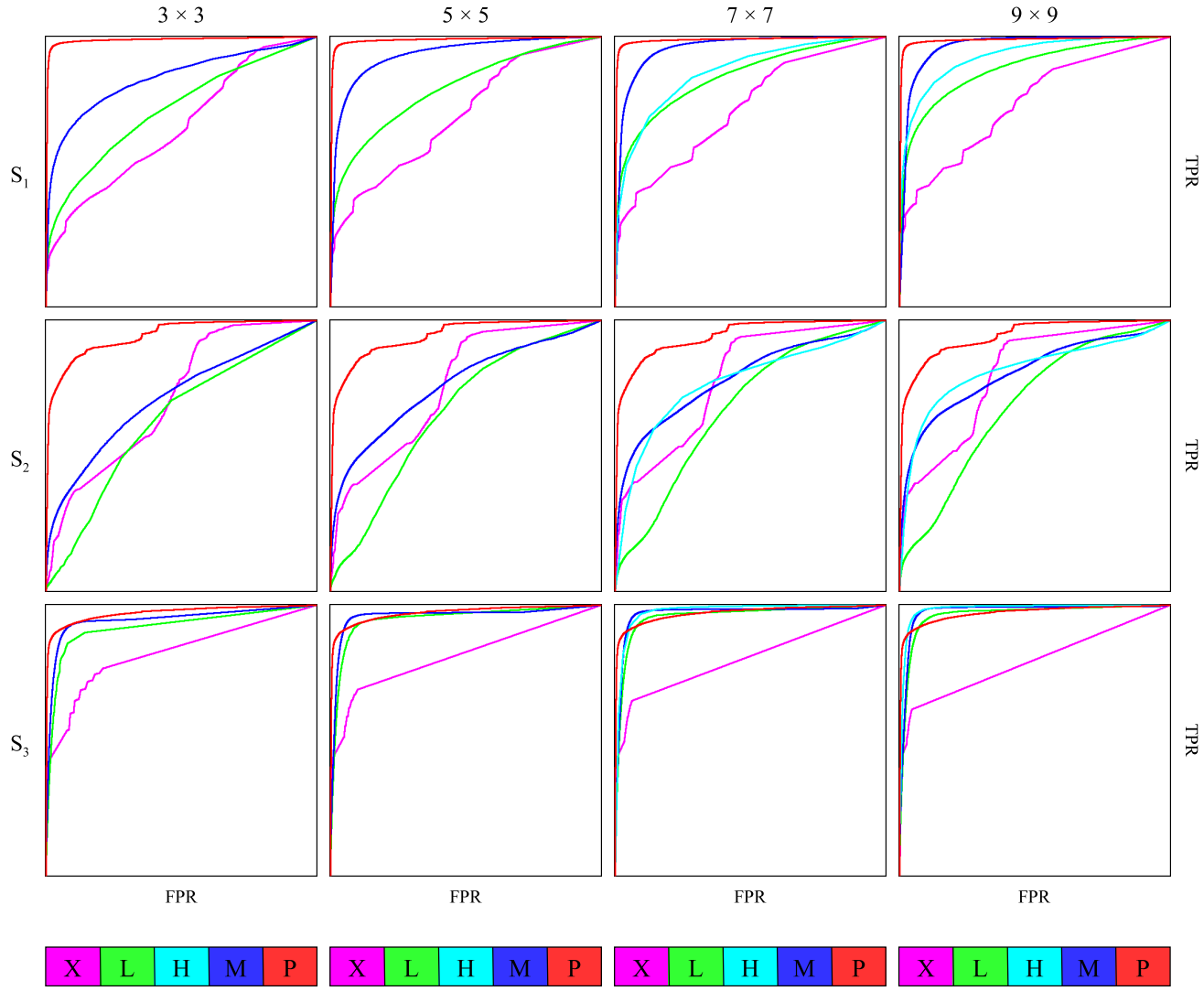
$$\hat{P}_{\mathcal{C}}(x, y \,|\, \boldsymbol{\theta}_{\mathcal{C}}) = \frac{1}{k} \sum_{i=x-\hat{b}_{\mathcal{C}}^{(x)}}^{x+\hat{b}_{\mathcal{C}}^{(x)}} \sum_{j=y-\hat{b}_{\mathcal{C}}^{(y)}}^{y+\hat{b}_{\mathcal{C}}^{(y)}} \hat{h}_{\mathcal{C}}(i, j) \tag{11}$$

where $k = (\hat{b}_{\mathcal{C}}^{(x)} \cdot 2 + 1) \cdot (\hat{b}_{\mathcal{C}}^{(y)} \cdot 2 + 1)$ is the kernel support area. In practice, the moving average operation in (11) is performed by means of the Summed Area Table incremental technique [1], thus exhibiting constant complexity with respect to the estimated bandwidths.

## 3. Experimental results

We present experimental results aimed at comparing the accuracy-efficiency performance of our proposal to that of four robust state-of-the-art background subtraction approaches [4, 5, 7, 11] belonging to the second class outlined in Section 1, namely those algorithms that model a-priori the intensity changes yielded by disturbs over small neighborhoods of pixels. We point out that we do not consider here algorithms belonging to the first class, such as [2, 10], since we are primarily interested in robustness against sudden intensity changes that such algorithms are inherently unable to deal with effectively.

All the considered algorithms, hereinafter referred to as X (Xie) [11], L (Lanza) [5], H (Heikkila) [4] and M (Mittal) [7], more or less explicitly assume an order-preserving model for local intensity variations caused by disturbs. However, they rely on different order-consistency tests and are characterized, respectively, by $O(N)$, $O(N \times n)$, $O(N \times n)$ and $O(N \times n^2)$ complexity, where $N$ and $n$ denote, respectively, the number of pixels in the image and in the considered neighborhood. It is straightforward to observe that the proposed approach, from now on denoted as P, exhibits $O(N)$ complexity.




| | AUC | | | | |
|---|---|---|---|---|---|
| $S_1$ | .666 | .731 | | .846 | **.987** |
| $S_2$ | .714 | .633 | | .714 | **.942** |
| $S_3$ | .822 | .918 | | .945 | **.970** |

| | AUC | | | | |
|---|---|---|---|---|---|
| $S_1$ | .708 | .809 | | .934 | **.987** |
| $S_2$ | .757 | .661 | | .761 | **.942** |
| $S_3$ | .812 | .947 | | .958 | **.970** |

| | AUC | | | | |
|---|---|---|---|---|---|
| $S_1$ | .732 | .852 | .868 | .956 | **.987** |
| $S_2$ | .783 | .675 | .770 | .786 | **.942** |
| $S_3$ | .801 | .957 | **.976** | .968 | .970 |

| | AUC | | | | |
|---|---|---|---|---|---|
| $S_1$ | .751 | .882 | .927 | .963 | **.987** |
| $S_2$ | .799 | .687 | .816 | .806 | **.942** |
| $S_3$ | .791 | .962 | **.984** | .974 | .970 |

| | TNR (TPR = 0.8) | | | | |
|---|---|---|---|---|---|
| $S_1$ | .359 | .625 | | .734 | **.994** |
| $S_2$ | .489 | .535 | | .441 | **.934** |
| $S_3$ | .777 | .948 | | .963 | **.993** |

| | TNR (TPR = 0.8) | | | | |
|---|---|---|---|---|---|
| $S_1$ | .437 | .630 | | .928 | **.994** |
| $S_2$ | .569 | .525 | | .533 | **.934** |
| $S_3$ | .896 | .954 | | .970 | **.993** |

| | TNR (TPR = 0.8) | | | | |
|---|---|---|---|---|---|
| $S_1$ | .493 | .725 | .875 | .948 | **.994** |
| $S_2$ | .613 | .488 | .635 | .555 | **.934** |
| $S_3$ | .937 | .959 | .987 | .970 | **.993** |

| | TNR (TPR = 0.8) | | | | |
|---|---|---|---|---|---|
| $S_1$ | .529 | .809 | .925 | .957 | **.994** |
| $S_2$ | .663 | .497 | .719 | .598 | **.934** |
| $S_3$ | .955 | .960 | .981 | .968 | **.993** |

| | FRAME RATE (FPS) | | | | |
|---|---|---|---|---|---|
| $S_{1,2,3}$ | 320 | 239 | | 20 | **451** |

| | FRAME RATE (FPS) | | | | |
|---|---|---|---|---|---|
| $S_{1,2,3}$ | 320 | 103 | | 3 | **451** |

| | FRAME RATE (FPS) | | | | |
|---|---|---|---|---|---|
| $S_{1,2,3}$ | 320 | 57 | 21 | 1 | **451** |

| | FRAME RATE (FPS) | | | | |
|---|---|---|---|---|---|
| $S_{1,2,3}$ | 320 | 35 | 19 | .3 | **451** |

Figure 3. Quantitative results: ROC curves (top), AUC and TNR values (center), frame rates (bottom) reported by the evaluated algorithms X (violet), L (green), H (cyan), M (blue) and P (red).

Experimental results have been obtained on three test sequences $S_1, S_2, S_3$ characterized by sudden and strong intensity variations due to illumination changes and camera gain and exposure adjustments. The background and two sample frames for each sequence are shown in the first and second row of Fig. 4, respectively. Backgrounds have been inferred off-line by temporally averaging an initial sequence of frames free of moving objects and then rounding the computed intensities to the nearest integer. A $3 \times 3$ neighborhood and a value $\tau = -3$ have been used in our algorithm for the preliminary change mask computation.

To allow for quantitative evaluation of performance in terms of accuracy, ground truth binary masks have been obtained by manual labeling and accuracy measures have been derived by computing the true positive rate (TPR) versus false positive rate (FPR) receiver operating characteristic (ROC) curve. Unlike P, which uses a fixed $3\times3$ patch in the first stage, algorithms X, L, H and M have been evaluated using neighborhoods of increasing size ($3\times3$, $5\times5$, $7\times7$, $9\times9$). As for H, evaluations start from $7\times7$ since, in the implemented original formulation of the algorithm, this is the minimum allowable size. Fig. 3, top, shows the ROC curves obtained for each algorithm, sequence and neighborhood size. Since P does not depend on the neighborhood size, for each sequence its ROC curve is shown in each of the four graphs. From each curve we have also extracted two scalar accuracy measures, the area under the curve (AUC), which represents the probability for the algorithm to assign to a randomly chosen changed pixel a higher change score than to a randomly chosen unchanged pixel, and the true negative rate (TNR) corresponding to a TPR of 80% . These measures are reported in the tables shown in Fig. 3, center.

As far as neighborhood-based algorithms (i.e. X, L, H, M) are concerned, the results show that their performances tend to increase with the neighborhood size. Results show also that, independently of the considered neighborhood size and test sequence, X turns out to be the worst performing algorithm while H and M are the best performing ones, with H prevailing in $S_3$ and M in $S_1$. As for the proposed approach P, it clearly outperforms all the considered algorithms. Only H and M with $7 \times 7$ and $9 \times 9$ neighborhood size on sequence $S_2$ exhibits a similar AUC. However, given a complexity of $O(N)$ for our algorithm and of $O(N \times n)$, $O(N \times n^2)$ for H and M, respectively, P turns out much faster than H and M, i.e. orders of magnitude faster.

This is confirmed by Fig. 3, bottom, where we report the average frame rates (in frames per second, FPS) of the evaluated algorithms over the three test sequences for different neighborhood sizes. Averaging is justified by test sequences having the same resolution (i.e. $320 \times 240$ pixels). The target PC is an Intel Core i3 2.27GHz, 4GB RAM. Results provide clear evidence that P outperforms X, L and, in particular, H and M in terms of efficiency.

Some qualitative results are also presented in Fig. 4. In particular, for each sequence we show, from top to bottom, the inferred background, two sample frames, the binary change masks yielded, respectively, by X, L, H, M, and P, the ground truth masks. The change masks have been obtained by choosing, for each algorithm and sequence, the threshold value yielding TPR equal to 80% and, for X, L, H and M, by using neighborhoods of size $9 \times 9$. Strong intensity variations are yielded by disturbs (i.e. illumination changes, camera gain and exposure adjustments) along the sequences, as can be judged by visually comparing the corresponding background-frame pairs. The reported change masks confirm the above observations that X and H, M are, respectively, the worst and best performing among the considered neighborhood-based approaches and that P outperforms X, L, H and M. In particular, it is worth pointing out how in the two sample frames of the challenging indoor sequence $S_2$, characterized by very strong lightening and darkening effects, all the neighborhood-based approaches exhibit very poor performance. On the contrary, the proposed algorithm performs very well, as vouched by the good quality of the obtained change mask.

## 4. Conclusions

We have proposed on-line probabilistic clustering of pixels into the background or foreground class based on a non-parametric mixture of two discrete distributions to perform background subtraction efficiently and robustly in presence of common sources of disturbance such as illumination changes, camera gain and exposure variations, noise.

As shown by experimental results on challenging sequences, this method improves over algorithms based on the a-priori definition of a model of locally admissible intensity variations: its ability to estimate such a model frame by frame allows it to obtain high sensitivity without sacrificing specificity. This promising trade-off is yielded without penalizing efficiency: our linear-complexity algorithm runs 1-2 orders of magnitude faster than those obtaining comparable accuracy.

An interesting prosecution for this work is a sound treatment of the kernel bandwidth estimation by replacing the current heuristic approach with an actual M-step so as to allow for a proper iterative deployment of the EM algorithm.

## References


[1] F. Crow. Summed-area tables for texture mapping. *Computer Graphics*, 18(3):207–212, 1984.

[2] A. Elgammal, D. Harwood, and L. Davis. Non-parametric model for background subtraction. In *Proc. ECCV'00*, volume 1843, pages 751–767, apr 2000.

[3] S. Y. Elhabian, K. M. El-Sayed, and S. H. Ahmed. Moving object detection in spatial domain using background re-

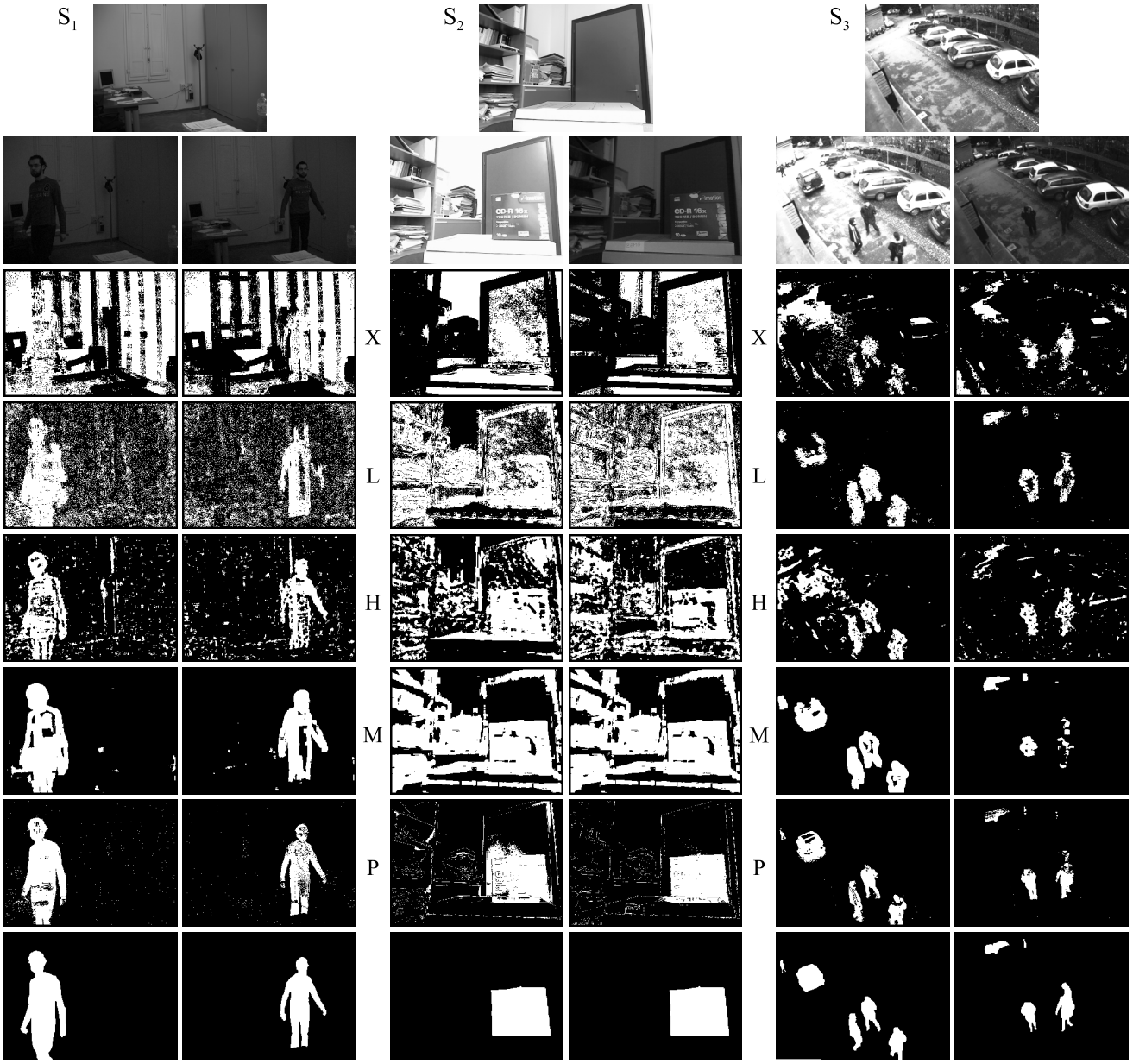



Figure 4. Qualitative results: change masks obtained by using, for each algorithm, the threshold yielding TPR = 0.8.

moval techniques - state-of-art. *Recent Patents on Computer Sciences*, 1:32–54, 2008.

[4] M. Heikkila and M. Pietikanen. A texture-based method for modeling the background and detecting moving objects. *PAMI*, 28(4):657–662, apr 2006.

[5] A. Lanza and L. D. Stefano. Detecting changes in grey level sequences by ML isotonic regression. In *Proc. AVSS'06*, pages 1–4, nov 2006.

[6] J. Lou, H. Yang, W. Hu, and T. Tan. An illumination-invariant change detection algorithm. In *Proc. ACCV'02*, volume 1, pages 13–18, jan 2002.

[7] A. Mittal and V. Ramesh. An intensity-augmented ordinal measure for visual correspondence. In *Proc. CVPR'06*, volume 1, pages 849–856, jun 2006.

[8] N. Ohta. A statistical approach to background subtraction for surveillance systems. In *Proc. ICCV'01*, volume 2, pages 481–486, jul 2001.

[9] B. W. Silverman. *Density Estimation for Statistics and Data Analysis*. Chapman and Hall, London, 1986.

[10] C. Stauffer and W. E. L. Grimson. Adaptive background mixture models for real-time tracking. In *Proc. CVPR'99*, volume 2, pages 246–252, jun 1999.

[11] B. Xie, V. Ramesh, and T. Boult. Sudden illumination change detection using order consistency. *IVC*, 22(2):117–125, feb 2004.