Second-Order Polynomial Models for Background Subtraction

Alessandro Lanza, Federico Tombari, Luigi Di Stefano

DEIS, University of Bologna
Bologna, Italy
{*alessandro.lanza, federico.tombari, luigi.distefano* }*@unibo.it*

Abstract. This paper is aimed at investigating background subtraction based on second-order polynomial models. Recently, preliminary results suggested that quadratic models hold the potential to yield superior performance in handling common disturbance factors, such as noise, sudden illumination changes and variations of camera parameters, with respect to state-of-the-art background subtraction methods. Therefore, based on the formalization of background subtraction as Bayesian regression of a second-order polynomial model, we propose here a thorough theoretical analysis aimed at identifying a family of suitable models and deriving the closed-form solutions of the associated regression problems. In addition, we present a detailed quantitative experimental evaluation aimed at comparing the different background subtraction algorithms resulting from theoretical analysis, so as to highlight those more favorable in terms of accuracy, speed and speed-accuracy tradeoff.

1 Introduction

Background subtraction is a crucial task in many video analysis applications, such as e.g. intelligent video surveillance. One of the main challenges consists in handling disturbance factors such as noise, gradual or sudden illumination changes, dynamic adjustments of camera parameters (e.g. exposure and gain), vacillating background, which are typical nuisances within video-surveillance scenarios. Many different algorithms for dealing with these issues have been proposed in literature (see [1] for a recent survey). Popular algorithms based on statistical per-pixel background models, such as e.g. Mixture of Gaussians (MoG) [2] or kernel-based non-parametric models [3], are effective in case of gradual illumination changes and vacillating background (e.g. waving trees). Unfortunately, though, they cannot deal with those nuisances causing sudden intensity changes (e.g. a light switch), yielding in such cases lots of false positives.

Instead, an effective approach to tackle the problem of sudden intensity changes due to disturbance factors is represented by a priori modeling over small image patches of the possible spurious changes that the scene can undergo. 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. Thanks to this approach, gradual as well as sudden photometric distortions do not yield false positives provided that they are explained by the model. Thus, the main issue concerns the choice of the a priori model: in principle, the more restrictive such a model, the higher is the ability to detect changes (sensitivity) but the lower is robustness to sources of disturbance (specificity). Some proposals assume disturbance factors to yield *linear* intensity transformations [4, 5]. Nevertheless, as discussed in [6], many non-linearities may arise in the image formation process, so that a more liberal model than linear is often required to achieve adequate robustness in practical applications. Hence, several other algorithms adopt *order-preserving* models, i.e. assume monotonic non-decreasing (i.e. non-linear) intensity transformations [6–9].

Very recently, preliminary results have been proposed in literature [10, 11] that suggest how second-order polynomial models hold the potential to yield superior performance with respect to the classical previously mentioned approaches, being more liberal than linear proposals but still more restrictive than the order preserving ones. Motivated by these encouraging preliminary results, in this work we investigate on the use of second-order polynomial models within a Bayesian regression framework to achieve robust background subtraction. In particular, we first introduce a family of suitable second-order polynomial models and then derive closed-form solutions for the associated Bayesian regression problems. We also provide a thorough experimental evaluation of the algorithms resulting from theoretical analysis, so as to identify those providing the highest accuracy, the highest efficiency as well as the best tradeoff between the two.

2 Models and solutions

For a generic pixel, let us denote as $\mathbf{x} = (x_1, \ldots, x_n)^T$ and $\mathbf{y} = (y_1, \ldots, y_n)^T$ the intensities of a surrounding neighborhood of pixels observed in the two images under comparison, i.e. background and current frame, respectively. We aim at detecting scene changes occurring in the pixel by evaluating the local intensity information contained in $\mathbf{x}$ and $\mathbf{y}$. In particular, classification of pixels as changed or unchanged is carried out by a priori assuming a model of the local photometric distortions that can be yielded by sources of disturbance and then testing, for each pixel, whether the model can explain the intensities $\mathbf{x}$ and $\mathbf{y}$ observed in the surrounding neighborhood. If this is the case, the pixel is likely sensing an effect of disturbs, so it is classified as unchanged; otherwise, it is marked as changed.

2.1 Modeling of local photometric distortions

In this paper we assume that main photometric distortions are due to noise, gradual or sudden illumination changes, variations of camera parameters such as exposure and gain. We do not consider here the vacillating background problem (e.g. waving trees), for which the methods based on multi-modal and temporally adaptive background modeling, such as [2] and [3], are more suitable.

As for noise, first of all we assume that the background image is computed by means of a statistical estimation over an initialization sequence (e.g. temporal averaging of tens of frames) so that noise affecting the inferred background intensities can be neglected. Hence, $\mathbf{x}$ can be thought of as a deterministic vector of noiseless background intensities. As for the current frame, we assume that noise is additive, zero-mean, i.i.d. Gaussian with variance σ^2. Hence, noise affecting the vector $\mathbf{y}$ of current frame intensities can be expressed as follows:

$$p(\mathbf{y}|\tilde{\mathbf{y}}) = \prod_{i=1}^{n} p(y_i|\tilde{y}_i) = \prod_{i=1}^{n} \mathcal{N}(\tilde{y}, \sigma^2) = \left(\sqrt{2\pi}\sigma\right)^{-n} \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \tilde{y}_i)^2\right) \quad (1)$$

where $\tilde{\mathbf{y}} = (\tilde{y}_1, \ldots, \tilde{y}_n)^T$ denotes the (unobservable) vector of current frame noiseless intensities and $\mathcal{N}(\mu, \sigma^2)$ the normal pdf with mean μ and variance σ^2.

As far as remaining photometric distortions are concerned, we assume that noiseless intensities within a neighborhood of pixels can change due to variations of scene illumination and of camera parameters according to a second-order polynomial transformation $\phi(\cdot)$, i.e.:

$$\tilde{y}_i = \phi\,(x_i; \boldsymbol{\theta}) = (1,\, x_i,\, x_i^2)\,(\theta_0,\, \theta_1,\, \theta_2)^T = \theta_0 + \theta_1 x_i + \theta_2 {x_i}^2 \quad \forall\, i = 1, \ldots, n \quad (2)$$

It is worth pointing out that the assumed model (2) does not imply that the whole frame undergoes the same polynomial transformation but, more generally, that such a constraint holds locally. In other words, each neighborhood of intensities is allowed to undergo a different polynomial transformation, so that local illumination changes can be dealt with.

From (1) and (2) we can derive the expression of the likelihood $p(\mathbf{x}, \mathbf{y}|\boldsymbol{\theta})$, that is the probability of observing the neighborhood intensities $\mathbf{x}$ and $\mathbf{y}$ given a polynomial model $\boldsymbol{\theta}$:

$$p(\mathbf{x}, \mathbf{y}|\boldsymbol{\theta}) = p(\mathbf{y}|\boldsymbol{\theta}; \mathbf{x}) = p(\mathbf{y}|\tilde{\mathbf{y}} = \phi(\mathbf{x}; \boldsymbol{\theta})) = \left(\sqrt{2\pi}\sigma\right)^{-n} \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{n}\left(y_i - \phi(x_i; \boldsymbol{\theta})\right)^2\right) \quad (3)$$

where the first equality follows from the deterministic nature of the vector $\mathbf{x}$ that allows to treat it as a vector of parameters.

In practice, not all the polynomial transformations belonging to the linear space defined by the assumed model (2) are equally likely to occur. In Figure (1), on the left, we show examples of less (in red) and more (in azure) likely transformations. To summarize the differences we can say that the constant term of the polynomial has to be small and that the polynomial has to be monotonic non-decreasing. We formalize these constraints by imposing a prior probability on the parameters vector $\boldsymbol{\theta} = (\theta_0, \theta_1, \theta_2)^T$, as illustrated in Figure (1), on the right. In particular, we implement the constraint on the constant term by assuming a zero-mean Gaussian prior with variance σ_0^2 for the parameter θ_0:

$$p(\theta_0) = \mathcal{N}(0, \sigma_0^2) \quad (4)$$

The monotonicity constraint is addressed by assuming for (θ_1, θ_2) a uniform prior inside the subset Θ_{12} of $\mathbb{R}^2$ that renders $\phi'(x; \boldsymbol{\theta}) = \theta_1 + 2\theta_2 \cdot x \geq 0$ for all

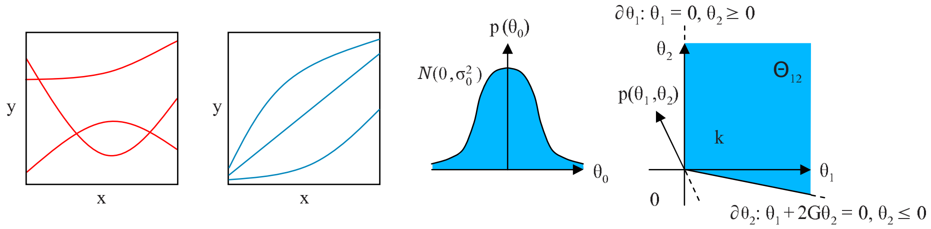

Fig. 1. Some polynomial transformations (left, in red) are less likely to occur in practice than others (left, in azure). To account for that, we assume a zero-mean normal prior for θ_0, a prior that is uniform inside Θ_{12}, zero outside for (θ_1,θ_2) (right).

$x \in [0\ ,\ G]$, with G denoting the highest measurable intensity ($G = 255$ for 8-bit images), zero probability outside Θ_{12}. Due to linearity of $\phi'(x)$, the monotonicity constraint $\phi'(x;\boldsymbol{\theta}) \geq 0$ over the entire x-domain $[0\ ,\ G]$ is equivalent to impose monotonicity at the domain extremes, i.e. $\phi'(0;\boldsymbol{\theta}) = \theta_1 \geq 0$ and $\phi'(G;\boldsymbol{\theta}) = \theta_1 + 2G \cdot \theta_2 \geq 0$. Hence, we can write the constraint as follows:

$$p(\theta_1,\theta_2) = \begin{cases} k & \text{if } (\theta_1,\theta_2) \in \Theta_{12} = \left\{(\theta_1,\theta_2)\in\mathbb{R}^2 :\ \theta_1 \geq 0\ \wedge\ \theta_1 + 2G\theta_2 \geq 0\right\} \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

We thus obtain the prior probability of the entire parameters vector as follows:

$$p(\boldsymbol{\theta}) = p(\theta_0) \cdot p(\theta_1,\theta_2) \tag{6}$$

In this paper we want to evaluate six different background subtraction algorithms, relying on as many models for photometric distortions obtained by combining the assumed noise and quadratic polynomial models in (1) and (2), that imply (3), with the two constraints in (4) and (5). In particular, the six considered algorithms (Q stands for quadratic) are:

Q_{∞} : (1) $\wedge$ (2) $\Rightarrow$ (3) plus prior (4) for θ_0 with $\sigma_0^2 \to \infty$ (θ_0 free);
Q_{f} : (1) $\wedge$ (2) $\Rightarrow$ (3) plus prior (4) for θ_0 with σ_0^2 finite positive;
Q_{0} : (1) $\wedge$ (2) $\Rightarrow$ (3) plus prior (4) for θ_0 with $\sigma_0^2 \to 0$ ($\theta_0 = 0$);
$\mathrm{Q}_{\infty,M}$: same as Q_{∞} plus prior (5) for (θ_1,θ_2) (monotonicity constraint);
$\mathrm{Q}_{f,M}$: same as Q_{f} plus prior (5) for (θ_1,θ_2) (monotonicity constraint);
$\mathrm{Q}_{0,M}$: same as Q_{0} plus prior (5) for (θ_1,θ_2) (monotonicity constraint);

2.2 Bayesian polynomial fitting for background subtraction

Independently from the algorithm, scene changes are detected by computing a measure of the distance between the sensed neighborhood intensities $\mathbf{x}$, $\mathbf{y}$ and the space of models assumed for photometric distortions. In other words, if the intensities are not well-fitted by the models, the pixel is classified as changed. The minimum-distance intensity transformation within the model space is computed by a maximum a posteriori estimation of the parameters vector:

$$\boldsymbol{\theta}_{\mathrm{MAP}} = \underset{\boldsymbol{\theta}\,\in\,\mathbb{R}^3}{\mathit{argmax}}\ p(\boldsymbol{\theta}|\mathbf{x},\mathbf{y}) = \underset{\boldsymbol{\theta}\,\in\,\mathbb{R}^3}{\mathit{argmax}}\ \left[p(\mathbf{x},\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})\right] \tag{7}$$

where the second equality follows from Bayes rule. To make the posterior in (7) explicit, an algorithm has to be chosen. We start from the more complex one, i.e. $\mathrm{Q}_{f,M}$. By substituting (3) and (4)-(5), respectively, for the likelihood $p(\mathbf{x},\mathbf{y}|\boldsymbol{\theta})$ and the prior $p(\boldsymbol{\theta})$ and transforming posterior maximization into minus log-posterior minimization, after eliminating the constant terms we obtain:

$$\boldsymbol{\theta}_{\mathrm{MAP}} = \underset{\boldsymbol{\theta} \in \Theta}{argmin} \left[E(\boldsymbol{\theta};\mathbf{x},\mathbf{y}) = d(\boldsymbol{\theta};\mathbf{x},\mathbf{y}) + r(\boldsymbol{\theta}) = \sum_{i=1}^{n} (y_i - \phi(x_i;\boldsymbol{\theta}))^2 + \lambda\theta_0^2 \right] \tag{8}$$

The objective function to be minimized $E(\boldsymbol{\theta};\mathbf{x},\mathbf{y})$ is the weighted sum of a data-dependent term $d(\boldsymbol{\theta};\mathbf{x},\mathbf{y})$ and a regularization term $r(\boldsymbol{\theta})$ which derive, respectively, from the likelihood of the observed data $p(\mathbf{x},\mathbf{y}|\boldsymbol{\theta})$ and the prior of the first parameter $p(\theta_0)$. The weight of the sum, i.e. the regularization coefficient, depends on both the likelihood and the prior and is given by $\lambda = \sigma^2/\sigma_0^2$. The prior of the other two parameters $p(\theta_1,\theta_2)$ expressing the monotonicity constraint has translated into a restriction of the optimization domain from $\mathbb{R}^3$ to $\Theta = \mathbb{R} \times \Theta_{12}$. It is worth pointing out that the data dependent term represents the least-squares regression error, i.e. the sum over all the pixels in the neighborhood of the square differences between the frame intensities and the background intensities transformed by the model. By making $\phi(x_i;\boldsymbol{\theta})$ explicit and after simple algebraic manipulations, it is easy to observe that the objective function is quadratic, so that it can be compactly written as:

$$E(\boldsymbol{\theta};\mathbf{x},\mathbf{y}) \;=\; (1/2)\,\boldsymbol{\theta}^T H\,\boldsymbol{\theta} - \mathbf{b}^T\boldsymbol{\theta} + c \tag{9}$$

with the matrix H, the vector $\mathbf{b}$ and the scalar c given by:

$$H \;=\; 2\begin{pmatrix} N & Sx & Sx^2 \\ Sx & Sx^2 & Sx^3 \\ Sx^2 & Sx^3 & Sx^4 \end{pmatrix} \qquad \mathbf{b} \;=\; 2\begin{pmatrix} Sy \\ Sxy \\ Sx^2y \end{pmatrix} \qquad c \;=\; Sy^2 \tag{10}$$

and, for simplicity of notation:

$$\begin{array}{llll} Sx = \sum_{i=1}^{n} x_i & Sx^2 = \sum_{i=1}^{n} x_i^2 & Sx^3 = \sum_{i=1}^{n} x_i^3 & Sx^4 = \sum_{i=1}^{n} x_i^4 \\ Sy = \sum_{i=1}^{n} y_i & Sxy = \sum_{i=1}^{n} x_i y_i & Sx^2y = \sum_{i=1}^{n} x_i^2 y_i & Sy^2 = \sum_{i=1}^{n} y_i^2 \end{array} \quad N = n + \lambda \tag{11}$$

As for the optimization domain $\Theta = \mathbb{R} \times \Theta_{12}$, with Θ_{12} defined in (5) and illustrated in Figure 1, it also can be compactly written in matrix form as follows:

$$\Theta \;=\; \left\{ \boldsymbol{\theta} \in \mathbb{R}^3 : \;\; Z\,\boldsymbol{\theta} \geq 0 \right\} \qquad \text{with} \qquad Z = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 1 & 2G \end{pmatrix} \tag{12}$$

The estimation problem (8) can thus be written as a quadratic program:

$$\boldsymbol{\theta}_{\mathrm{MAP}} \;=\; \underset{Z\,\boldsymbol{\theta} \geq 0}{argmin} \left[(1/2)\,\boldsymbol{\theta}^T H\;\boldsymbol{\theta} - \mathbf{b}^T\boldsymbol{\theta} + c \right] \tag{13}$$

If in the considered neighborhood there exist three pixels characterized by different background intensities, i.e. $\exists\, i,j,k : \; x_i \neq x_j \neq x_k$, it can be demonstrated that the matrix H is positive-definite. As a consequence, since H is the Hessian of the quadratic objective function, in this case the function is strictly convex. Hence, it admits a unique point of unconstrained global minimum $\boldsymbol{\theta}^{(u)}$ that can be easily calculated by searching for the unique zero-gradient point, i.e. by solving the linear system of normal equations:

$$\boldsymbol{\theta}^{(u)} = \boldsymbol{\theta} \in \mathbb{R}^3 : \quad \nabla E(\boldsymbol{\theta}) = 0 \quad \equiv \quad (H/2)\,\boldsymbol{\theta} = (\mathbf{b}/2) \tag{14}$$

for which a closed-form solution is obtained by computing the inverse of $H/2$:

$$\boldsymbol{\theta}^{(u)} = (H/2)^{-1}(\mathbf{b}/2) = \frac{1}{|H/2|}\begin{pmatrix} A & D & E \\ D & B & F \\ E & F & C \end{pmatrix}\begin{pmatrix} Sy \\ Sxy \\ Sx^2y \end{pmatrix} \tag{15}$$

where:

$$\begin{array}{lll} A = Sx^2Sx^4 - \left(Sx^3\right)^2 & B = NSx^4 - \left(Sx^2\right)^2 & C = NSx^2 - (Sx)^2 \\ D = Sx^2Sx^3 - Sx\,Sx^4 & E = Sx\,Sx^3 - \left(Sx^2\right)^2 & F = Sx\,Sx^2 - NSx^3 \end{array} \tag{16}$$

and

$$|H/2| = N\,A + Sx\,D + Sx^2E \tag{17}$$

If the computed point of unconstrained global minimum $\boldsymbol{\theta}^{(u)}$ belongs to the quadratic program feasible set Θ, i.e. satisfies the monotonicity constraint $Z\,\boldsymbol{\theta} \geq 0$, then the minimum distance between the observed neighborhood intensities and the model of photometric distortions is simply determined by substituting $\boldsymbol{\theta}^{(u)}$ for $\boldsymbol{\theta}$ in the objective function. A compact close-form expression for such a minimum distance can be obtained as follows:

$$\begin{aligned} E^{(u)} = E(\boldsymbol{\theta}^{(u)}) &= {\boldsymbol{\theta}^{(u)}}^T(H/2)\,\boldsymbol{\theta}^{(u)} - {\boldsymbol{\theta}^{(u)}}^T\mathbf{b} + c = {\boldsymbol{\theta}^{(u)}}^T(\mathbf{b}/2) - {\boldsymbol{\theta}^{(u)}}^T\mathbf{b} + c \\ &= c - {\boldsymbol{\theta}^{(u)}}^T(\mathbf{b}/2) = Sy^2 - |H/2|^{-1}\left(Sy\,{\theta_0}^{(u)} + Sxy\,{\theta_1}^{(u)} + Sx^2y\,{\theta_2}^{(u)}\right) \end{aligned} \tag{18}$$

The two algorithms Q_f and Q_∞ rely on the computation of the point of unconstrained global minimum $\boldsymbol{\theta}^{(u)}$ by (15) and, subsequently, of the unconstrained minimum distance $E^{(u)} = E(\boldsymbol{\theta}^{(u)})$ by (18). The only difference between the two algorithms is the value of the pre-computed constant $N = n + \lambda$ that in Q_∞ tends to n due to $\sigma_0^2 \to \infty$ causing $\lambda \to 0$. Actually, Q_f corresponds to the method proposed in [10].

If the point $\boldsymbol{\theta}^{(u)}$ falls outside the feasible set Θ, the solution $\boldsymbol{\theta}^{(c)}$ of the constrained quadratic programming problem (13) must lie on the boundary of the feasible set, due to convexity of the objective function. However, since the monotonicity constraint $Z\,\boldsymbol{\theta} \geq 0$ does not concern θ_0, again the partial derivative of the objective function with respect to θ_0 must vanish in correspondence of the solution. Hence, first of all we impose this condition, thus obtaining:

$${\theta_0}^{(c)} = (1/N)\left(Sy - \theta_1 Sx - \theta_2 Sx^2\right) \tag{19}$$

We thus substitute ${\theta_0}^{(c)}$ for θ_0 in the objective function, so that the original 3-d problem turns into a 2-d problem in the two unknowns θ_1, θ_2 with the feasible set Θ_{12} defined in (5) and illustrated in Figure 1, on the right. As previously mentioned, the solution of the problem must lie on the boundary of the feasible set $\partial\Theta_{12}$, that is on one of the two half-lines:

$$\partial\Theta_1 :\ \theta_1 = 0 \ \wedge\ \theta_2 \geq 0 \qquad\qquad \partial\Theta_2 :\ \theta_1 = -2\,G\,\theta_2 \ \wedge\ \theta_2 \leq 0 \tag{20}$$

The minimum of the 2-d objective function on each of the two half-lines can be determined by replacing the respective line equation into the objective function and then searching for the unique minimum of the obtained 1-d convex quadratic function in the unknown θ_2 restricted, respectively, to the positive ($\partial\Theta_1$) and the negative ($\partial\Theta_2$) axis. After some algebraic manipulations, we obtain that the two minimums $E_1^{(c)}$ and $E_2^{(c)}$ are given by:

$$E_1^{(c)} = Sy^2 - \frac{(Sy)^2}{N} - \begin{cases} \dfrac{T^2}{N\,B} & \text{if } T > 0 \\ 0 & \text{if } T \leq 0 \end{cases} \qquad E_2^{(c)} = Sy^2 - \frac{(Sy)^2}{N} - \begin{cases} \dfrac{V^2}{N\,U} & \text{if } V < 0 \\ 0 & \text{if } V \geq 0 \end{cases} \tag{21}$$

where:

$$\begin{aligned} T &= N\,Sx^2y - Sx^2Sy & \qquad V &= T + 2\,G\,W \\ W &= SxSy - N\,Sxy & \qquad U &= B + 4G^2C + 4G\,F \end{aligned} \tag{22}$$

The constrained global minimum $E^{(c)}$ is thus the minimum between $E_1^{(c)}$ and $E_2^{(c)}$. Hence, similarly to Q_f and Q_∞, the two algorithms $\mathrm{Q}_{f,M}$ and $\mathrm{Q}_{\infty,M}$ rely on the preliminary computation of the point of unconstrained global minimum $\boldsymbol{\theta}^{(u)}$ by (15). However, if the point does not satisfy the monotonicity constraint in (12), the minimum distance is computed by (21) instead of by (18). The two algorithms $\mathrm{Q}_{f,M}$ and $\mathrm{Q}_{\infty,M}$ differ in the exact same way as Q_f and Q_∞, i.e. only for the value of the pre-computed parameter N.

The two remaining algorithms, namely Q_0 and $\mathrm{Q}_{0,M}$, rely on setting $\sigma_0^2 \to 0$. This implies that $\lambda \to \infty$ and, therefore, $N = (n + \lambda) \to \infty$. As a consequence, closed-form solutions for these algorithms can not be straightforwardly derived from the previously computed formulas by simply substituting the value of N. However, $\sigma_0^2 \to 0$ means that the parameter θ_0 is constrained to be zero, that is the quadratic polynomial model is constrained to pass through the origin. Hence, closed-form solutions for these algorithms can be obtained by means of the same procedure outlined above, the only difference being that in the model (2) θ_0 has to be eliminated. Details of the procedure and solutions can be found in [11].

By means of the proposed solutions, we have no need to resort to any iterative approach. In addition, it is worth pointing out that all terms involved in the calculations can be computed either off-line (i.e. those involving only background intensities) or by means of very fast incremental techniques such as Summed Area Table [12] (those involving also frame intensities). Overall, this allows the proposed solutions to exhibit a computational complexity of $O(1)$ with respect to the neighborhood size n.

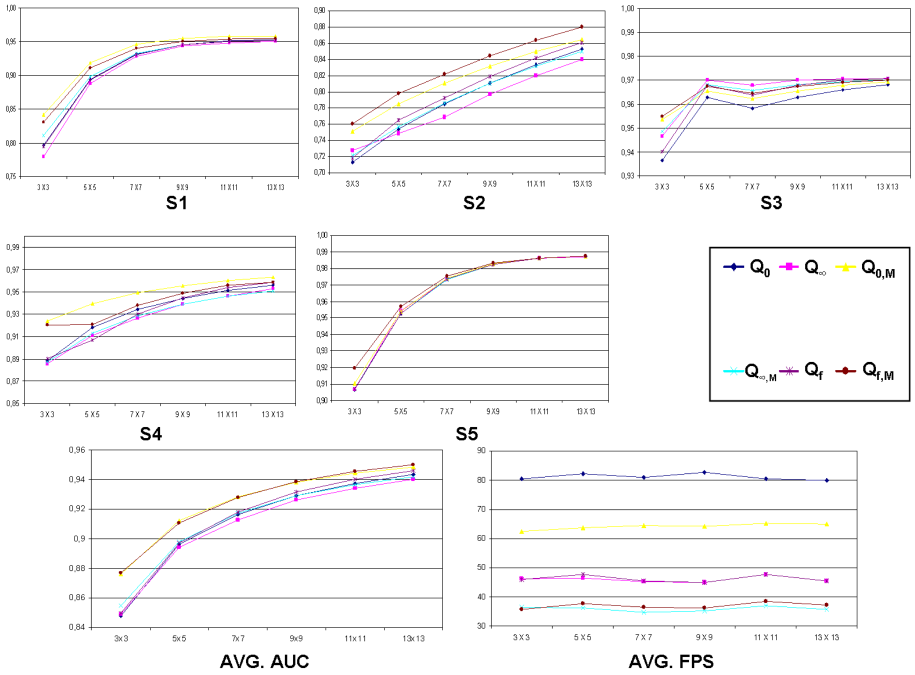

Fig. 2. Top: AUC values yielded by the evaluated algorithms with different neighborhood sizes on each of the 5 test sequences. Bottom: average AUC (left) and FPS (right) values over the 5 sequences.

3 Experimental results

This Section proposes an experimental analysis aimed at comparing the 6 different approaches to background subtraction based on a quadratic polynomial model derived in previous Section. To distinguish between the methods we will use the notation described in the previous Section. Thus, e.g., the approach presented in [10] is referred to here as Q_f, and that proposed in [11] as $Q_{0,M}$. All algorithms were implemented in C using incremental techniques [12] to achieve $O(1)$ complexity. They also share the same code structure so to allow for a fair comparison in terms not only of accuracy, but also computational efficiency.

Evaluated approaches are compared on five test sequences, S_1–S_5, characterized by sudden and notable photometric changes that yield both linear and non-linear intensity transformations. We acquired sequences S_1–S_4 while S_5 is a synthetic benchmark sequence available on the web [13]. In particular, S_1, S_2 are two indoor sequences, while S_3, S_4 are both outdoor. It is worth pointing out that by computing the 2-d joint histograms of background versus frame intensities, we observed that S_1, S_5 are mostly characterized by linear intensity changes, while S_2–S_4 exhibit also non-linear changes. Background images together with sample frames from the sequences are shown in [10, 11]. Moreover, we point out here that, based on an experimental evaluation carried out on S_1–S_5, methods Q_f and $Q_{0,M}$ have been shown to deliver state-of-the-art performance in [10, 11]. In particular, Q_f and $Q_{0,M}$ yield at least equivalent performance compared to the most accurate existing methods while being, at the same time, much more efficient. Hence, since results attained on S_1–S_5 by existing linear and order-preserving methods (i.e. [5, 7–9]) are reported in [10, 11], in this paper we focus on assessing the relative merits of the 6 developed quadratic polynomial

Experimental results in terms of accuracy as well as computational efficiency are provided. As for accuracy, quantitative results are obtained by comparing the change masks yielded by each approach against the ground-truths (manually labeled for S_1-S_4, available online for S_5). In particular, we computed the True Positive Rate (TPR) versus False Positive Rate (FPR) Receiver Operating Characteristic (ROC) curves. Due to lack of space, we can not show all the computed ROC curves. Hence, we summarize each curve with a well-known scalar measure of performance, the Area Under the Curve (AUC), which represents the probability for the approach to assign a randomly chosen changed pixel a higher change score than a randomly chosen unchanged pixel [14].

Each graph shown in Figure 2 reports the performance of the 6 algorithms in terms of AUC with different neighborhood sizes (3×3, 5×5, $\cdots$, 13×13). In particular, the first 5 graphs are relative to each of the 5 testing sequences, while the two graphs on the bottom show, respectively, the mean AUC values and the mean Frame-Per-Second (FPS) values over the 5 sequences. By analyzing AUC values reported in the Figure, it can be observed that two methods yield overall a better performance among those tested, that is, $Q_{0,M}$ and $Q_{f,M}$, as also summarized by the mean AUC graph. In particular, $Q_{f,M}$ is the most accurate on S_2 and S_5 (where $Q_{0,M}$ is the second best), while $Q_{0,M}$ is the most accurate on S_1 and S_4 (where $G_{f,M}$ is the second best). The different results on S_3, where Q_∞ is the best performing method, appear to be mainly due to the presence of disturbance factors (e.g. specularities, saturation, $\ldots$) not well modeled by a quadratic transformation: thus, the best performing algorithm in such specific circumstance turns out to be the less constrained one (i.e. Q_∞).

As for efficiency, the mean FPS graph in Figure 2 proves that all methods are $O(1)$ (i.e. their complexity is independent of the neighborhood size). As expected, the more constraints are imposed on the adopted model, the higher the computational cost is, resulting in a reduced efficiency. In particular, an additional computational burden is brought in if a full quadratic form is assumed (i.e. not homogeneous), similarly if the transformation is assumed to be monotonic. Given this consideration, the most efficient method turns out to be Q_0, the least efficient ones $Q_{\infty,M}$, $Q_{f,M}$, with Q_∞, $Q_{0,M}$, Q_f staying in the middle. Also, the results prove that the use of a non-homogeneous form adds a higher computational burden compared to the monotonic assumption. Overall, the experiments indicate that the method providing the best accuracy-efficiency tradeoff is $Q_{0,M}$.

4 Conclusions

We have shown how background subtraction based on Bayesian second-order polynomial regression can be declined in different ways depending on the nature of the constraints included in the formulation of the problem. Accordingly, we have derived closed-form solutions for each of the problem formulations. Experimental evaluation show that the most accurate algorithms are those based on the monotonicity constraint and, respectively a null $Q_{0,M}$ or finite $Q_{f,M}$ variance for the prior of the constant term. Since the more articulated the constraints within the problem the higher computational complexity, the most efficient algorithm results from a non-monotonic and homogeneous formulation (i.e. Q_0). This also explains why $Q_{0,M}$ is notably faster than $Q_{f,M}$, so as to turn out the method providing the more favorable tradeoff between accuracy and speed.

References

1. 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 Sciences **1** (2008) 32–54
2. Stauffer, C., Grimson, W.E.L.: Adaptive background mixture models for real-time tracking. In: Proc. CVPR'99. Volume 2. (1999) 246–252
3. Elgammal, A., Harwood, D., Davis, L.: Non-parametric model for background subtraction. In: Proc. ICCV'99. (1999)
4. Durucan, E., Ebrahimi, T.: Change detection and background extraction by linear algebra. Proc. IEEE **89** (2001) 1368–1381
5. Ohta, N.: A statistical approach to background subtraction for surveillance systems. In: Proc. ICCV'01. Volume 2. (2001) 481–486
6. Xie, B., Ramesh, V., Boult, T.: Sudden illumination change detection using order consistency. Image and Vision Computing **22** (2004) 117–125
7. Mittal, A., Ramesh, V.: An intensity-augmented ordinal measure for visual correspondence. In: Proc. CVPR'06. Volume 1. (2006) 849–856
8. Heikkila, M., Pietikainen, M.: A texture-based method for modeling the background and detecting moving objects. IEEE Trans. PAMI (2006)
9. Lanza, A., Di Stefano, L.: Detecting changes in grey level sequences by ML isotonic regression. In: Proc. AVSS'06. (2006) 1–4
10. Lanza, A., Tombari, F., Di Stefano, L.: Robust and efficient background subtraction by quadratic polynomial fitting. In: Proc. Int. Conf. on Image Processing (ICIP'10). (2010)
11. Lanza, A., Tombari, F., Di Stefano, L.: Accurate and efficient background subtraction by monotonic second-degree polynomial fitting. In: Proc. AVSS'10. (2010)
12. Crow, F.: Summed-area tables for texture mapping. Computer Graphics **18** (1984) 207–212
13. MUSCLE Network of Excellence: (Motion detection video sequences)
14. Bradley, A.P.: The use of the area under the ROC curve in the evaluation of machine learning algorithms. Pattern Recognition **30** (1997) 1145–1159