Foreground Segmentation in Videos Combining General Gaussian Mixture Modeling and Spatial Information

Article *in* IEEE Transactions on Circuits and Systems for Video Technology · February 2017

DOI: 10.1109/TCSVT.2017.2665970

2 authors:

Aissa Boulmerka
Centre universitaire de Mila

Mohand Saïd Allili
Université du Québec en Outaouais

Foreground Segmentation in Videos Combining General Gaussian Mixture Modeling and Spatial Information

Aïssa Boulmerka and Mohand Saïd Allili

***Abstract*—We present a new statistical approach combining temporal and spatial information for robust online background subtraction (BS) in videos. Temporal information is modeled by coupling finite mixtures of Generalized Gaussian (MoGG) distributions with foreground/background co-occurrence analysis. Spatial information is modeled by combining multi-scale inter-frame correlation analysis and histogram matching. We propose an online algorithm that efficiently fuses both information to cope with several BS challenges, such as cast shadows, illumination changes, and various complex background dynamics. In addition, global video information is used through a displacement measuring technique to deal with pan-tilt-zoom (PTZ) camera effects. Experiments with comparison with recent state-of-the-art methods have been conducted on standard datasets. Obtained results have shown that our approach surpasses several state-of-the-art methods on the aforementioned challenges while maintaining comparable computational time.**

***Index Terms*—Background subtraction (BS), temporal/spatial information, mixture models, co-occurrence/correlation analysis, cast shadows, dynamic backgrounds, pan-tilt-zoom (PTZ).**

I. Introduction

Background subtraction (BS) is a fundamental and crucial task for several video processing applications such as smart video surveillance [10], human activity recognition [52] and interactive gaming [16]. Over the past years, a tremendous number of BS techniques have been proposed (see [5], [38], [40] and refs therein). To simplify the problem formulation and ensure good success, BS algorithms generally assume three conditions [5]: stationary cameras, constant illumination conditions and static background (i.e., no dynamics or noise occur in the background). Several challenges arise from the violation of these conditions, among which: *cast shadows and illumination changes*, *dynamic backgrounds*, *noisy videos*, *camera jitter*, *stationary objects*, *camouflage* and *pan-tilt-zoom* (PTZ) camera effects. These challenges usually produce a huge amount of false positives and/or negatives (see [5] for an extended list of BS challenges).

Considering the case of static-camera videos, several approaches have been proposed to address some of the aforementioned challenges [5], [45]. *Temporal approaches* have been proposed to cope with multimodal backgrounds and gradual illumination changes, for example, *parametric statistical models* and *non-parametric models* have been used to model pixel history. Generally, statistical models can achieve good success in separating moving objects from mild background changes (e.g., swaying trees, gradual daylight changes, etc.). However, assuming independent temporal constraint is not sufficient to handle complex BS challenges. Therefore, temporal approaches may lead to a huge amount of false detections especially in the presence of shadows, sudden illumination changes and/or complex background dynamics (e.g. fountains, camera jitter).

To cope with BS challenges such as complex dynamic backgrounds or illumination changes, *spatiotemporal* approaches have been proposed. These approaches have the advantage of taking into account dependencies between pixels by incorporating both temporal and spatial information. Spatiotemporal analysis can be performed through several techniques, such as spatial GMM [53], texture analysis [21], [7], Markov random fields [41], video bricks [31], [9], and low-rank and sparse decomposition [14], [18], [23], [49], to name a few. Usually, these methods give more robust BS results than using only temporal information. However, most of the proposed algorithms are dedicated to dealing with one or two challenges but give poor performance for other challenges [6]. For example, methods dealing with dynamic backgrounds are generally less efficient to deal with cast shadows and sudden illumination changes [5]. Finally, most of these methods are limited by their high computational cost and they are not easily adaptable to online video processing.

Real-world videos may also be acquired using moving cameras. Most of methods dealing with such case use motion-compensated BS or video segmentation. Motion-compensated BS estimate first the motion of the camera from the video, and then operate the BS using static-camera techniques [47], [62]. *Video segmentation approaches* groups pixels into spatiotemporal regions that exhibit coherence in both appearance and motion [20]. Foreground objects are then detected using techniques such as ranking object proposals [29], [32], [63], [37], saliency detection [57], [59], [58], multi-state selection graph [17], constrained Laplacian optimization [48] and point trajectories [8]. Video segmentation techniques are more adapted for videos acquired by moving cameras. However, most of these methods are computationally intensive and are designed to perform in an offline scenario [57].

We propose an online approach for background subtraction (BS) that is efficient for coping with several challenges concerning videos acquired by static cameras or containing PTZ effects. Our model combines temporal and spatial information for BS. Temporal information is modeled locally using an online-learned mixture of generalized Gaussian (MoGG) distributions [1]. In fact, MoGGs are more efficient than GMMs to fit a broad range of data histograms (e.g., with *leptokurtic* and *platykurtic* modes) and is more robust to noise and outliers. Herein, we introduce a new procedure for real-time online updating of the MoGG parameters in the context of BS. Moreover, we use background/foreground co-occurrence analysis to enhance the MoGG ability to model different background dynamics. This allows to drastically decrease the amount of false positives caused by complex background dynamics (e.g., fountains) and give more flexibility to model variable pixel state duration. Spatial information is incorporated through multi-scale inter-frame correlation analysis and histogram matching. Our approach not only allows for dissociating changes due to shadows and illumination changes from those of moving objects, but also for enhancing the accuracy of BS in the presence of noise, camera jitter and complex background dynamics. Finally, we use a global technique based on inter-frame displacement analysis to deal with simple PTZ camera motion effects. Experiments on standard datasets have shown that our method outperforms several state-of-the-art methods on most of the challenges mentioned above.

This paper is organized as follows: Section II presents related work. Section III describes our approach combining temporal and spatial information modeling for BS. Section IV presents our experimental results. We end with a conclusion and perspectives.

II. Related work

A. Temporal-based approaches

Temporal approaches consider the history of independent pixels; and thus construct a global background model. They can be classified in two following subcategories.

1) Parametric models: Gaussian mixture models (GMMs) are the most popular parametric models used for BS [50], [25]. GMMs are able to cope with gradual illumination changes and backgrounds with small repetitive motions (e.g., moving vegetation, etc.) [35], [65]. However, slow objects tend to be rapidly absorbed by the background. In addition, sudden illuminations changes and shadows generate object-like patterns of motion that are classified as foreground. Finally, the GMM learning rate is usually hand-tuned and does not adapt to the video content. Several improvements have been proposed to mitigate these limitations by automatic updating of the GMM component number and learning rate [25], [65], using adaptive thresholds [35], or by replacing the Gaussian distribution with more flexible ones [13]. These improvements can achieve some automation in adapting the GMM parameters to background dynamics. However, the performance drastically decreases with challenges such as thick shadows and complex background dynamics.

Pixel history can also be modeled as finite states corresponding to events (e.g., lights on/off, cloudy/sunny or object presence/absence) [51]. For instance, hidden Markov models (HMMs) have been successfully used to model motorway events (”background”, ”shadow” and ”moving cars”) [26], [43]. Methods using HMMs can be effective in modeling scenarios with consistent temporal behavior. However, they lack flexibility by having tight temporal duration for each state [26].

2) Non-parametric models: Kernel Density Estimation (KDE) is a non-parametric pixel level background modeling approach [12]. KDE guarantees a smooth and continuous version of the background distribution. However, it is very demanding regarding computational time and memory storage. Besides, shadows and illumination changes are not well handled using this approach. In [27], the background model is built using codewords which are created by clustering sample background values at the pixel level during a training phase. This allows to describe dynamic background regions and avoid stopped object using a limited memory. However, this approach cannot handle permanent background changes since the updating mechanism does not allow the creation of new codewords.

Temporal-based approaches have been used a lot in the literature since they are simple to implement and give more precise detection. However, these methods may fail with complex scenarios, as they consider only the pixel history and disregard any kind of pixel spatial context.

B. Spatiotemporal-based approaches

Instead of exploiting temporal history of pixels independently, *spatiotemporal approaches* take into consideration both spatial and temporal information when modeling and/or separating the background/foreground. According to the level at which spatial information is used, spatiotemporal approaches can be broadly classified in three categories: region, brick and frame levels.

1) Region-level models: A fixed size region is used around a pixel to include neighborhood information. For example, [24] proposed a BS method where local temporal and spatial data are assumed to follow the same distribution. This method can achieve some robustness to noise and coherence for foreground detection. However, it is not efficient in dealing with shadows, illumination changes, and complex background dynamics. In [53], a GMM extension is proposed by taking into account the spatial dependency between pixels at the region level. This allows to perform BS in scenes with dynamic background and camera jitter. Hofmann *et al.* proposed the PBAS [22] which is based on dynamic thresholding and *a neighboring random rule* to update the background model over time. This method is efficient in coping with background dynamics (e.g., swaying trees, jitter), but objects tend to be rapidly absorbed by backgrounds while shadows and illumination changes can be detected as objets.

Inspired by PBAS, the SuBSENSE algorithm [7] uses *local binary patterns* (LBP) to model the relationship between neighboring pixels along with pixel-level feedback loop for dynamic decision thresholds. This enables to model several types of background dynamics. However, it does not deal efficiently with illumination changes, shadows, and camouflage problems. In [41], *Markov random fields* (MRFs) have been used to model the spatial coherence between pixels for BS. Generally, MRFs provide spatial consistency for BS labeling but are less efficient to handle challenges such dynamic backgrounds, shadows, camera jitter and PTZ effects. In addition, they are computational intensive which is limitation to real-time processing. Generally, region-based approaches allow to deal locally with noisy scenes and mild background dynamics, but are sensitive to sudden illumination and complex dynamics.

2) Brick-level models: In these approaches, 3D spatiotemporal structures named *video bricks* are exploited to build background models for BS. Lin *et al.* [31] have designed a 3D descriptor to deal with complex background scenes by pursuing subspaces within video bricks and using the ARMA (auto regressive moving average model) to separate foregrounds from instable backgrounds. However, the method lacks flexibility to deal with more challenging scenes. In [34], a spatiotemporal saliency algorithm is proposed for foreground detection. This is carried out by combining 3D motion features and dynamic texture models. This method outperforms its predecessors by reducing the average error rate, but at a cost of a huge computational time. Chen *et al.* [9] have used optical flow (OF) to refine and update the noisy background obtained from the GMMs. This enables to deal efficiently with scenes containing stopped objects or objects with slow motion (e.g., stopping cars, person waiting in a queue). However, errors in OF estimation may badly influence the accuracy of the derived BS.

3) Frame-level models: These can be considered as the extreme case of the brick level when the size of bricks are the frames. One of the most used approaches in this category is *eigenvalue decomposition* of spatiotemporal video volumes [36]. It enables implicit encoding of spatial relation between pixels while avoiding tiling effects of block partitioning. Several works in the past have proposed to extract video foregrounds using *low-rank and sparse decomposition* via Robust Principal Component Analysis (RPCA) [11], [49], [18]. Videos are decomposed into two matrices, a low-rank matrix representing the background and a sparse matrix representing the foreground. Other versions of this approach using Higher-order RPCA (HoRPCA) [18], Bayesian Tensor Factorization (BRTF) [64] and RPCA with a dynamic tree-structured foreground [11] have been proposed.

One of the major limitations of RCPA-based approaches is their tendency to recognize foreground objects with slow motion as background. Moreover, they are not efficient in processing videos in an online fashion in addition of being computational demanding. To overcome these drawbacks, some online approaches have been proposed [14], [23], [49]. In [49], an online tensor decomposition of spatiotemporal features is proposed for foreground detection. In the same vein, online tensor subspace learning [23] is used to represent spatial dependencies between pixels for foreground detection. These approaches can deal efficiently with some challenges in an online fashion. However, are not generalizable to all challenges (e.g., complex backgrounds dynamics, camera jitter, local illumination changes, shadows and PTZ effects), while they are computationally expensive.

C. Overview of our contribution

Our goal is to achieve an online BS by a simple yet efficient new procedures for combining spatial and temporal information. In addition to dealing with several complex background dynamics, our approach is less sensitive to shadows, illumination changes, camera jitter and PTZ camera effects. Finally, it is optimized to process videos with near real-time capability. We briefly summarize our contributions as follows:

1) We propose and online approach for foreground segmentation combining temporal and spatial information. Compared to previous spatia-temporal methods, our approach can cope efficiently with several challenges such as cast shadows, illumination changes and complex background dynamics. We also propose several procedures to optimize the computation time of our algorithm. 2) a new scheme is proposed for temporal information modeling by coupling MoGGs and objects/background co-occurrence analysis. This enables to accurately modeling random background dynamics (e.g., fast foreground/background switching, fountains), 3) we model spatial information using inter-frame spatial structure and histogram analysis. Spatial information makes our BS method less sensitive to shadows and illumination changes. We propose also a procedure for adapting the learning rate of the MoGG model to the scene context. This enables, for example, a quick absorption of drastic background changes induced by PTZ operations.

III. Temporal/spatial information modeling

The proposed algorithm is composed of temporal and spatial modules interacting with each other for efficient BS (see Fig. 1). Temporal information is modeled by combining MoGGs and co-occurrence analysis, which allows for an accurate representation of various complex background dynamics. Spatial information is incorporated into the method using correlation analysis and histogram matching which mitigate effects of cast shadows, highlights, illumination changes and PTZ effects. This information is also used to derive an adaptive scheme to estimate the learning rate of the MoGG parameters. This scheme contributes also to accelerate the convergence rate of the background model and prevent it from rapidly absorbing objects.

A. Basic temporal information modeling using MoGGs

The MoGG model has the flexibility to accurately fit different histogram shapes while ensuring robustness to noise and/or outliers which cause heavy-tailed distributions [2]. The one-dimensional generalized Gaussian density (GGD) is defined in $\mathbb{R}$ as follows:

$$p(X|\theta) = K(\lambda, \sigma) \exp\big(- A(\lambda) \, |(X - \mu)/\sigma|^{\lambda} \big), \tag{1}$$

where $\theta = \{\mu, \sigma, \lambda\}$ is the set of GGD parameters, $K(\lambda, \sigma) = \lambda\sqrt{\Gamma(3/\lambda)/\Gamma(1/\lambda)}/(2\sigma\Gamma(1/\lambda))$ and $A(\lambda) = [\Gamma(3/\lambda)/\Gamma(1/\lambda)]^{\lambda/2}$; $\Gamma(.)$ being the gamma function. The parameters μ and σ are the GGD location and scale parameters. The parameter λ controls the kurtosis of the probability density function (pdf) and determines whether its shape is peaked or flat. To model temporal changes in video, we consider the history of each pixel (x, y) at time t as $\{\vec{X}_0, ..., \vec{X}_t\}$. Each vector $\vec{X}_t$ is D-dimensional $\vec{X}_t = (X_{1,t}, ..., X_{D,t}) \in \mathbb{R}^D$ ($D = 3$ for RGB color). Suppose that the history of the pixel at time t is modeled as a mixture of K components where, given that the dimensions of $\vec{X}_t$ are independent in each class, the probability of observing the vector $\vec{X}_t$ is given as [1]:

$$p(\vec{X}_t) = \sum_{i=1}^{K} \omega_{i,t} * \Pi_{d=1}^{D} p(X_{d,t}|\vec{\theta}_{i,d,t}), \tag{2}$$

where $\vec{\theta}_{i,d,t} = (\mu_{i,d,t}, \sigma_{i,d,t}, \lambda_{i,d,t})$ are parameters describing the dimension d of the ith component of the mixture, $\omega_{1,t}$,..., $\omega_{K,t}$ are the weights of components such that $\sum_{i=1}^{K} \omega_{i,t} = 1$ and K is a parameter that represents the maximum number of foreground/background components.

We assume that at frame I_{t+1}, a pixel (x, y) have value $\vec{X}_{t+1}$ and a match is found with one of the components of the mixture (let's say with component k) if we have the following condition:

$$p(\vec{\theta}_{k,t}|\vec{X}_{t+1}) > \tau, \text{ with } \quad k = \arg\max_{i}\{p(\theta_{i,t}|\vec{X}_{t+1})\}, \tag{3}$$

where τ is a given threshold and $p(\theta_{i,t}|\vec{X}_{t+1})$ is the posterior probability of the ith mixture component. If a match is found, all the parameters of the matched component k are updated, whereas only the weight parameters are updated for the other components. In no match is found, a new component of the mixture is created. Note that an online updating method has been proposed in [1] using the Expectation-Maximisation (EM) algorithm. However, the procedure uses Fisher scoring which incurs a huge computation time to calculate the likelihood derivatives. Here, we propose a

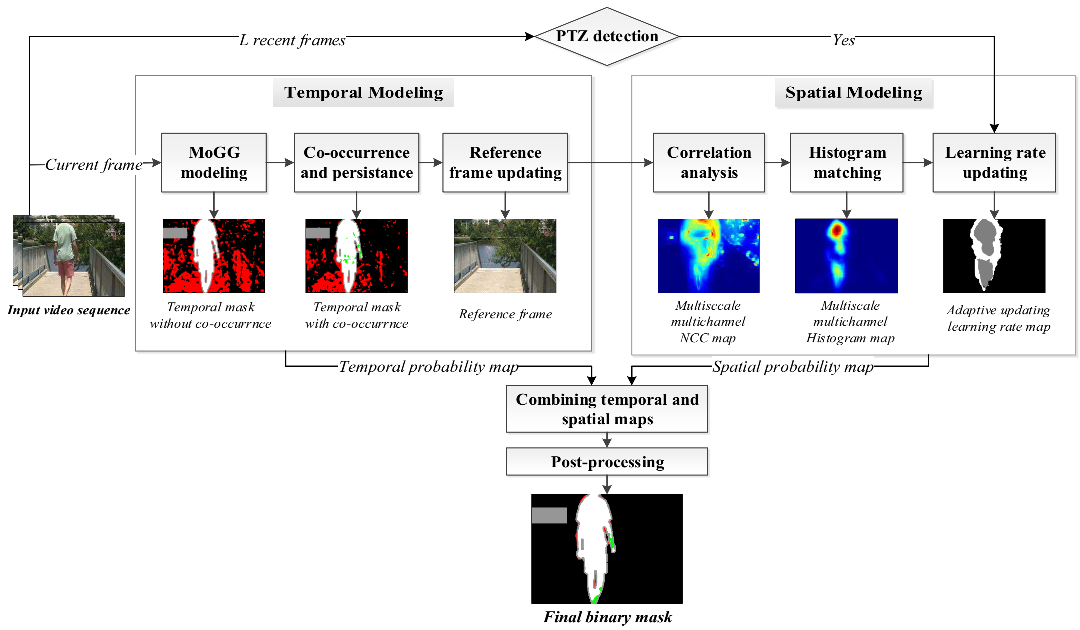


Fig. 1. The proposed algorithm architecture. In the binary masks: white, black, red, green and gray colors represent the true positives (TP), true negatives (TN), false positives (FP), false negatives (FN) and unknown pixels, respectively.

faster procedure based on statistical moments for online updating the MoGG parameters. First, since $\sum_{i=1}^{K} \omega_{i,t} = 1$, the weights are updated as follows [50]:

$$\omega_{i,t+1} = (1-\rho) * \omega_{i,t} + \rho * \delta(i=k), i = 1, ..., K \tag{4}$$

where δ is the delta function and ρ is a learning parameter. After this updating, we normalize all the weights. The entries of the shape parameter vector $\vec{\lambda}_k$ are updated using the following property [46]:

$$\left[\frac{\sigma_{k,d,t}}{\mathbb{E}\big[\mid X_{k,d,t} - \mu_{k,d,t} \mid\big]}\right]^2 = \frac{\Gamma\left(1/\lambda_{k,d,t}\right)\Gamma\left(3/\lambda_{k,d,t}\right)}{\Gamma^2\left(2/\lambda_{k,d,t}\right)}, \tag{5}$$

where $X_{k,d,t}$ are values of the dth dimension of $\vec{X}$ assigned to component k until time t, $\mu_{k,d,t}$ is the location parameter of the same component and $\mathbb{E}\big[\mid X_{k,d,t} - \mu_{k,d,t} \mid\big]$ is the mean of centered absolute values (MAV), given as:

$$\mathbb{E}\big[\mid X_{k,d,t} - \mu_{k,d,t} \mid\big] = \frac{1}{N_k}\sum_{s=1}^{t} \mid X_{k,d,s} - \mu_{k,d,t} \mid, \tag{6}$$

where N_k is the number of data assigned to component k. Using the shorthand $\tilde{X}_{k,d,t}$ to designate $|X_{k,d,t} - \mu_{k,d,t}|$, the MAV is updated online as follows:

$$\mathbb{E}_{(t+1)}\left[\tilde{X}_{k,d,t+1}\right] = (1-\phi) * \mathbb{E}_{(t)}\left[\tilde{X}_{k,d,t}\right] + \phi * \tilde{X}_{k,d,t+1}. \tag{7}$$

For a matched component k, $\vec{\lambda}_k$ can be efficiently updated for each dimension using Eq. (5) via a quick look-up table search [46]. The location parameter of the same component is updated using Eq. (8) as follows [1]:

$$\mu_{k,d,t+1} = \frac{\sum_{s=1}^{t+1} \tilde{X}_{k,d,s}^{(\lambda_{k,d,t+1}-2)} * X_{k,d,s}}{\sum_{s=1}^{t+1} \tilde{X}_{k,d,s}^{(\lambda_{k,d,t+1}-2)}} = \frac{\alpha_{k,d}(t+1)}{\beta_{k,d}(t+1)}, \tag{8}$$

where $\lambda_{k,d,t+1}$ is the shape parameter for dimension d computed at time $t+1$. The terms $\alpha_{k,d}(.)$ and $\beta_{k,d}(.)$ can be updated online using the following equations:

$$\alpha_{k,d}(t+1) = \alpha_{k,d}(t) + X_{k,d,t+1} * \tilde{X}_{k,d,t}^{(\lambda_{k,d,t+1}-2)} \tag{9}$$

$$\beta_{k,d}(t+1) = \beta_{k,d}(t) + \tilde{X}_{k,d,t}^{(\lambda_{k,d,t+1}-2)}. \tag{10}$$

Finally, the scale parameter $\sigma_{k,d,.}$ is updated in frame I_{t+1} using the following online equation:

$$\sigma_{k,d,t+1} = \Big[(1-\phi) * (\sigma_{k,d,t})^{\lambda_{k,d,t}} + \phi * \lambda_{k,d,t+1} * A(\lambda_{k,d,t+1}) * (\tilde{X}_{k,d,t})^{\lambda_{k,d,t+1}}\Big]^{1/\lambda_{k,d,t+1}}, \tag{11}$$

where $A(\lambda)$ is given in Eq. (1). The parameter ρ represents a learning rate in all the above equations where $\rho = \phi/\omega_i$ and ϕ is named the learning factor. This factor is adaptively estimated using the spatial information as explained in section III-D. See Appendix A for a detailed description of the MoGG parameters derivations.

B. Temporal co-occurrence and persistence modeling

In the original GMM-based BS and its variants [5], the mixture components of each pixel are first sorted in the descending order of their weights (i.e., ω parameters). Then, the background model is constituted by the components with the highest weight values. This achieves good results only if background patterns are stable over time. In case of fast intermittent switching of object/background components over time, the performance of the model will decrease. Fig. 2 illustrates this fact by taking two samples in different locations of the image. The video is 'fountain01' from the 'dynamicBackground' category of the CDnet dataset [60]. The first location is illustrated in Fig. 2-(a) where the background (grass) is well separated from the

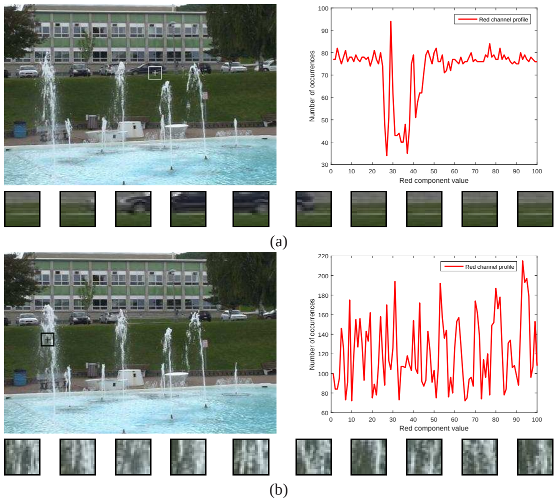


Fig. 2. An illustration for static and dynamic backgrounds. (a) The pixel at the white box illustrates a static background. In the top left: a sample frame from the CDnet dataset [60], 'dynamicBackground' category, 'fountain01' sequence. In the top right: Red channel history of the spotted pixel (frames: 651 to 750). (b) A dynamic background illustrated by the pixel at the black box. Captions are the same as in (a).

object (black car). Indeed, the grass component weight in the mixture overpasses 80%. The second location is illustrated in Fig. 2-(b) where the background contains dynamic random appearances of the ground-grass and the water drop fountain. The rapidly intermittent switching between the two has prevented the background from converging rapidly to two components.

To cope with this problem, we propose to analyse the *co-occurrence* and *persistence* of mixture components for accelerating the convergence of the background model. Let p be a pixel at position (x, y), K is the number of components, and let the set of variables and parameters be defined as follows:

- *The component co-occurrence matrix (CCM)* $\mathbf{A}$: is a $K \times K$ matrix with each element a_{ij}, $1 \leq i, j \leq K$, giving the number of times the pixel p is labelled with components c_i and c_j at time t and $t + 1$, respectively. Let $\mathbf{B} = (\mathbf{A} + \mathbf{A}^T)/2$, with diagonal set to zero.
- *The co-occurrence weights* $\boldsymbol{\eta}$: is a K-element normalized vector constituted of the r_m-th row of the matrix $\mathbf{B}$, where $m = 1, ..., M$.
- *The persistence vector* $\boldsymbol{\upsilon}$: is a K-element vector generated at each pixel with each element υ_j, $j = 1, ..., K$, expressing the duration of the component occurrence without interruption since its activation.
- *The number of dominant components* M: represents the number of dominant background components that interacts with the rest of components which may represent moving patterns of background. It is obvious that $M \ll K$ (usually M is set to 1).
- $R = \{r_1, r_2, ..., r_M\}$: is the set of dominant component indices sorted in descending order of their temporal weights ω.
- *The co-occurrence factor* ν: defines the importance given to the co-occurrence weights $\boldsymbol{\eta}$ versus the temporal weights $\boldsymbol{\omega}$, where $\nu \in [0, 1]$.
- *The set of combined weights* $\boldsymbol{\Pi} = \{\boldsymbol{\pi_1}, \boldsymbol{\pi_2}, ..., \boldsymbol{\pi_M}\}$: is the set of combined temporal/co-occurrence weight vectors, where each vector $\boldsymbol{\pi_m}$ is computed considering the r_m-th line of the co-occurrence matrix, and where $m = 1, ..., M$.
- *The final combined weights* $\boldsymbol{\pi}$: is a K-element vector defined as the average of the M combined weight vectors in the set $\boldsymbol{\Pi}$.

Note that component *persistence* is a complementary concept to the co-occurrence. At a given pixel p, we keep for each mixture component c_j for $j = 1, ..., K$ a count reflecting the number of successive occurrences of c_j in time. In other words, if c_j is matched in two successive frames t and $t+1$, its persistence is incremented by 1. Otherwise, it is reset to 1. It follows that stable components (background or foreground) will tend to have high persistence values.

Algorithm 1 combines the MoGG temporal information with the co-occurrence/persistance information. It starts with updating the CCM and extracting *the co-occurrence weights* $\boldsymbol{\eta}$, then each value η_j is divided by the corresponding persistence value υ_j. The new vector $\boldsymbol{\eta}$ encodes approximate probabilities of the components switching, in a similar way as the Markov chain transition matrix. *The co-occurrence weights* $\boldsymbol{\eta}$ are then combined with the MoGG *temporal weights* $\boldsymbol{\omega}$ to build *the combined weights* $\boldsymbol{\pi_m}$ using the linear formula $\boldsymbol{\pi_m} = \nu * \boldsymbol{\omega} + (1 - \nu) * \boldsymbol{\eta}$, where $\nu \in [0, 1]$ is a parameter that defines the importance given to *the co-occurrence weights* $\boldsymbol{\eta}$ versus *the temporal weights* $\boldsymbol{\omega}$. At the end, *the final combined weights* $\boldsymbol{\pi}$ can be computed as the average of the M combined weight vectors using the formula $\boldsymbol{\pi} = \frac{1}{M} \sum_{m=1}^{M} \boldsymbol{\pi_m}$.

The co-occurrence factor ν can be adjusted according to the scene nature. If the processed scene contains dynamic areas like swaying trees or fountains, then a higher value of ν is preferred, otherwise (i.e. the scene is only composed of stable background/foreground components) the co-occurrence weights have no effect on the computation of the combined weights that will be equal to the temporal weights. In our experiments, the parameter ν is set to 0.5 to carry the general case and provide a good trade-off between the co-occurrence and temporal model.

For illustration, consider the two scenarios presented in Fig. 2. Let us set $K = 7$ for the pixel p (the center of the square) with mixture component labels activated from frame 651 to frame 750. In Fig. 2-(a), the spotted pixel at position $(240, 110)$ represents a stable green grass ground with a black car passing over it. The matched MoGG temporal labels and the CCM $\mathbf{A}$ are given in Tab. I-(a) and Tab. I-(b), respectively. We can observe that the majority of the CCM entries are null and, consequently, all the co-occurrence weights vector $\boldsymbol{\eta}$ in Tab. I-(e) are also null. This is because there is practically one dominant stable background component (the grass) and there is little co-occurrence with the other foreground components (the black car crossing the road).

In Fig. 2-(b), the selected pixel is characterized by a rapidly intermittent switch between the grass ground and water of the fountain. The correspondent timeline labels and CCM are given in Tab. II-(a) and Tab. II-(b), respectively. For example, a_{11} (top left) is the number of times that the pixel p with component c_1 appears after the same component and a_{14} (top middle) is the number of times that the pixel p with component c_4 appears after component c_1. Indeed, the pixel switches rapidly between several components which can explain why all CCM entries are relatively high.

Contrarily to past works based on GMMs [5], our model assigns high weights to components that occur successively in time or have high switching rate with other components. This

Frames	Activated components																			
651 - 670	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
671 - 690	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
691 - 710	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	2	3	3
711 - 731	4	4	3	5	3	5	5	3	3	3	2	6	6	7	3	1	1	1	1	1
731 - 750	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

(a)

650	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

(b)

729	1	0	0	0	0	0
0	1	1	0	0	1	0
1	1	3	1	2	0	0
0	0	1	1	0	0	0
0	0	2	0	1	0	0
0	0	0	0	0	1	1
0	0	1	0	0	0	0

(c)

c_1	0.9964
c_2	0.0006
c_3	0.0019
c_4	0.0003
c_5	0.0006
c_6	0.0003
c_7	0.0000

(d)

0.0000
0.0000
0.0000
0.0000
0.0000
0.0000
0.0000

(e)

0.9964
0.0006
0.0019
0.0003
0.0006
0.0003
0.0000

(f)

TABLE I

CO-OCCURRENCE MODEL RELATED TO THE EXAMPLE AT FIG. 2-(A). (A) MOGG ACTIVATED COMPONENT LABELS. (B) CCM AT FRAME #651. (C) CCM AT FRAME #750. (D) MOGG TEMPORAL WEIGHTS $\boldsymbol{\omega}$. (E) CO-OCCURRENCE WEIGHTS $\boldsymbol{\eta}$. (F) COMBINED MOGG AND CO-OCCURRENCE WEIGHTS $\boldsymbol{\pi}$.

Frames	Activated components																			
651 - 670	1	1	1	1	2	6	3	7	1	1	1	1	1	1	1	1	1	1	2	1
671 - 690	3	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1
691 - 710	1	1	1	1	1	1	1	3	1	1	1	1	1	1	1	1	1	1	1	1
711 - 731	1	1	1	1	1	1	1	1	1	3	1	1	1	1	1	1	1	1	1	1
731 - 750	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1

(a)

504	9	9	13	7	7	1
8	0	0	0	0	0	0
9	0	0	0	0	0	0
14	0	0	3	1	0	0
6	0	0	2	1	0	0
7	0	0	0	0	2	0
0	1	0	0	0	0	0

(a)

586	13	12	13	7	7	1
11	0	0	0	0	1	0
12	0	0	0	0	0	1
14	0	0	3	1	0	0
6	0	0	2	1	0	0
7	0	1	0	0	2	0
1	1	0	0	0	0	0

(b)

c_1	0.9322
c_2	0.0338
c_3	0.0165
c_4	0.0051
c_5	0.0024
c_6	0.0094
c_7	0.0006

(c)

0.0000
0.2353
0.2353
0.2647
0.1275
0.1373
0.0000

(d)

0.4661
0.1346
0.1259
0.1349
0.0649
0.0733
0.0003

(e)

TABLE II

CO-OCCURRENCE MODEL RELATED TO THE EXAMPLE AT FIG. 2-(B). CAPTIONS AS IN TAB. I.

Algorithm 1: Compute the combined weights $\boldsymbol{\pi}$.

Data: MoGG models, co-occurrence matrices (CCM) $\mathbf{A}$, persistence vectors $\boldsymbol{\upsilon}$, number of dominant components M, and co-occurrence factor ν.
Result: Component combined weights $\boldsymbol{\pi}$.
for *each pixel* p **do**
  Let $\mathbf{A}$ *be the CCM and* c_{old}, c_{new} *be the old and new activated components, resp.*;
  $\mathbf{A}(c_{old}, c_{new}) \leftarrow \mathbf{A}(c_{old}, c_{new}) + 1$; `// update the CCM`
  if $c_{old} = c_{new}$ **then**
    $\upsilon(c_{new}) \leftarrow \upsilon(c_{new}) + 1$;
  else
    $\upsilon(c) \leftarrow 1 \quad with \quad c \neq c_{new}$;
  end
  $\mathbf{B} \leftarrow (\mathbf{A} + \mathbf{A}^T)/2$;
  $diag(\mathbf{B}) \leftarrow 0$; `// reset diagonal of B to 0`
  for *each component index* $m = 1 : M$ *in* R **do**
    $\boldsymbol{\eta} \leftarrow \mathbf{B}(r, :)$ with $r \leftarrow R(i)$;
    $\boldsymbol{\eta} \leftarrow \boldsymbol{\eta}/\boldsymbol{\upsilon}$;`// divide the co-occurrence weights by the persistence`
    $\boldsymbol{\eta} \leftarrow \boldsymbol{\eta}/(\sum_j \eta_j)$; `// normalize η`
    $\boldsymbol{\pi_m} \leftarrow \nu \times \boldsymbol{\omega} + (1-\nu) \times \boldsymbol{\eta}$;`// compute the combined temporal/co-occurrence weights`
    $\boldsymbol{\pi_m} \leftarrow \boldsymbol{\pi_m}/(\sum_j \pi_{m,j})$; `// normalize weights`
  end
  $\boldsymbol{\pi} \leftarrow \frac{1}{M}\sum_{m=1}^{M} \boldsymbol{\pi_m}$; `// final weights`
end

allows to reduce significantly false positives caused by background dynamics, such as fast swaying tree leaves, fountains, camera jitter, etc. The detailed procedure for updating the CCM and combining the MoGG and co-occurrence weights is outlined in Algorithm 1.

C. Spatial information modeling

Spatial information is added to our approach using local structure (or texture) and color distribution. This information is relatively stable under soft shadows, and illumination changes and will enforce our BS to overcome these challenges.

1) Correlation analysis: The spatial structure conformity with the background is done by multi-scale correlation analysis between patches. We recall that the normalized cross correlation (NCC) between two vectors $\vec{v}_1$ and $\vec{v}_2$ is defined as:

$$NCC(\vec{v}_1, \vec{v}_2) = \frac{\vec{v}_1 \cdot \vec{v}_2}{\parallel \vec{v}_1 \parallel \parallel \vec{v}_2 \parallel} \tag{12}$$

where $\parallel \vec{v} \parallel = \sqrt{\vec{v} \cdot \vec{v}}$ is the norm of $\vec{v}$. The NCC is invariant to linear scaling of the form $\vec{v}' = \gamma \vec{v}$, where $\gamma \in \mathbb{R}^*$. For our BS algorithm, for each pixel we approximate the current background reference by the mean of components with M highest mixture weights $\{\omega_{\{1\},t+1}, ..., \omega_{\{M\},t+1}\}$, where M is the number of dominant background components. The set of resulting reference frames $\{I_1, ..., I_M\}$ for the spatial module is build with local value for each pixel determined as follows:

$$I_m(x, y) = \mu_{\{m\},t+1}(x, y), m \in \{1, ..., M\}, \tag{13}$$

where $\mu_{\{m\},t+1}(x, y)$ is the mean parameter of the GGD corresponding to the weight $\omega_{\{m\},t+1}$.

Using the correlation between the reference frame I_m and the current frame I_{t+1}, we can compare the local structure between the current and the reference frames. We then derive an approximation of the spatial foreground/background probabilities. For more reliable estimation of these probabilities, we use multiple window sizes surrounding each pixel. That is, for each reference frame $I_m, m \in \{1, ..., M\}$, S correlation maps are computed: NCC_1, NCC_2,..., NCC_S for square blocks of size $N_1 \times N_1$, $N_2 \times N_2$,..., $N_S \times N_S$, respectively, where $N_1 < N_2 < ... < N_S$ (typically $S = 3$). These maps are obtained by factorizing correlations over color channels as:

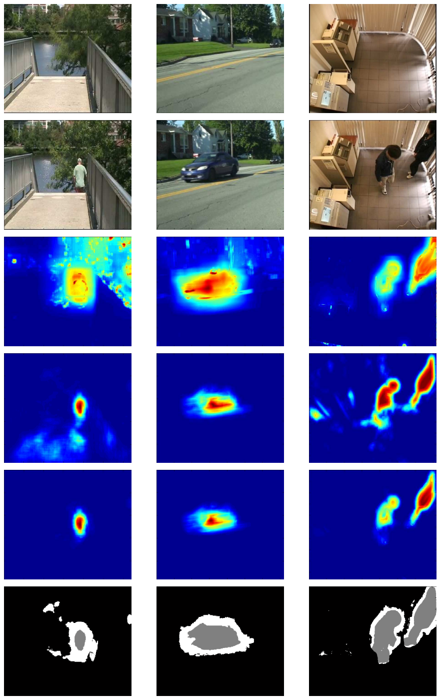
$$NCC_{m,j} = \prod_{d=1}^{D} NCC_{m,j,d}, \tag{14}$$

where $NCC_{m,j,d}$ corresponds to the correlation calculated with reference I_m using scale j and color channel d. Then, the maximum correlation among the reference frames is retained:

$$NCC_f = \max_{m=1..M} \Big(\max_{j=1..S} NCC_{m,j} \Big). \tag{15}$$

To reduce the computational cost of computing the correlation maps, the integral image is used [54]. In fact, template matching can be efficiently obtained in the d-th channel in the reference frame I_d and the corresponding channel in the current frame $I_{d,t+1}$ by computing the integral images of I_d^2, $I_{d,t+1}^2$, and $I_d * I_{d,t+1}$ images, then Eq. (12) can be computed at all image locations using simple arithmetic operations. The final correlation maps are then computed using Eqs. (14) and (15).

Finally, the spatial foreground and background probabilities are approximated using the lenient functions: $p_{s,f}(\vec{X}_{t+1}) \simeq$



(a) (b) (c)

Fig. 3. Spatial maps obtained for sample frames. Rows from top to down represent the reference frame, current frame, NCC map, histogram map, discrimination map and learning factor map (the black, white and gray colors represent ϕ_{low}, ϕ_{avg} and ϕ_{high} respectively). (a) frame #2700 from 'overpass', (b) frame #850 from the 'bungalows' sequence and (c) frame #1080 from the 'copyMachine' sequence.

$\exp(-f(NCC_f))$ and $p_{s,b}(\vec{X}_{t+1}) \simeq 1 - \exp(-f(NCC_f))$, where f is a linear function defined as $f(x) = w_1 x + w_2$ and w_1, w_2 are constants controlling the sensitivity of the probability to the spatial correlation.

Fig. 3 shows the NCC_f map obtained for a sample of frames from the Change Detection dataset [60]. The reference and original frames are shown in the first and second rows, respectively. The third row shows the obtained NCC_f map where darker regions are pixel surroundings which are close to those of the reference frame. By opposite, brighter regions are pixel surroundings with high foreground spatial probability.

2) Histogram matching: Spatial information can also be exploited through local color distribution. This can be useful when a video contains dynamic backgrounds (i.e., waving trees, water fountain, camera jitters, etc.) where the local structure of the background may slightly change but not the color distribution. Suppose that we have a reference image I at the fame I_{t+1}. Let R and R_{t+1} be the regions centered around a pixel (x, y) in frames I and I_{t+1}, respectively, H and H_{t+1} their respective histograms and N_{bins} is the number of bins in the histograms H_t and H_{t+1}. We use the Bhattacharyya distance $d(H_t, H_{t+1})$ to compare H_t and H_{t+1} as follows:

$$d(H_t, H_{t+1}) = 1 - \sum_{i=1}^{N_{bins}} \sqrt{H_t(i) * H_{t+1}(i)}; \qquad (16)$$

We compute the histograms for each reference frame $I_m, m \in \{1, ..., M\}$ in Eq. (13) at different scales using window sizes $W_1 < W_2 < ... < W_S$ (typically $S = 3$) and all D color channels using [39]. The histogram distance map $HIST_f$ given by Eq. (17), where $HIST_{m,s,d}$ corresponds to the distance map calculated with reference I_m at scale s and color channel d.

$$HIST_f = \prod_{s=1}^{S} \left(\max_{m=1..M} \left(\max_{d=1..D} HIST_{m,s,d} \right) \right). \qquad (17)$$

Finally, we build a discrimination map S_ϕ by combining the histogram final distance map $HIST_f$ and the correlation map NCC_f as shown in the Eq. (18).

$$S_\phi = \exp(-f(NCC_f)) * \big(1 - \exp(-g(HIST_f))\big), \qquad (18)$$

where f and g are two linear functions defined $f(x) = w_1 x + w_2$ and $g(x) = w_3 x + w_4$, with w_1 to w_4 set experimentally. Basically, the map S_ϕ enables to discriminate between changes due to illumination changes to those caused by jitter, background dynamics and the presence of foreground objects.

D. Adaptive learning rate for MoGG modeling

A bi-level thresholding is carried out on the discrimination map S_ϕ using two thresholds T_1 and T_2 to obtain the learning factor used by each MoGG temporal model. Consequently, the learning factor map should contain three levels of learning factors: $\phi_{low} < \phi_{avg} < \phi_{high}$ that correspond to high plausibility of foreground, unknown and high plausibility background, respectively. The dynamic estimation of the learning factor allows the temporal MoGG update procedure to assign a low learning factor ϕ_{low} to stopped objects or objects with slow motion. Background regions will be assigned a high learning factor ϕ_{high} that enables those regions to be quickly integrated into the background model. The third value ϕ_{avg} is assigned to pixels not identified strongly as either foreground or background according to the discrimination map S_ϕ. The learning parameter ρ of Eq. (4) is updated for each pixel using the equation $\rho = \phi/\omega_i$.

Fig. 3 shows NCC maps, histogram matching and the resulting learning factor for the sample frames. We can observe that the correlation analysis (CA) and the histogram matching (HM) are complementary in their nature. For example, in the 'overpass' frame, CA detects the tree leaves as false positives. On the other hand, HM failed to remove the person shadow and illumination changes that caused false detections. However, we can observe that most of these false detections are removed by combining the two spatial maps and, therefore, an appropriate learning factor map has been generated. Finally, the stopped object challenge given by the waiting woman in the 'copyMachine' sequence can not be integrated into the background model due to the low learning factor ϕ_{low} assigned to the woman pixels.

E. Detection of PTZ camera effect scenarios

Several works have been proposed to deal with PTZ camera effects [15], [61]. To detect the presence of PTZ camera effect in a given sequence, a global-level processing is performed based on the displacement estimation between two successive frames using cross-correlation. At each time t, both the previous and current frames are compared each other using a two-step cross-correlation algorithm. In the first step, we compute the correlation coefficients between overlapping patches of the two frames. In the second step, we find the best-match positions using the maximum of cross-correlation values obtained for all possible displacement in the search matrix. Finally, PTZ effects are detected using the average of displacements calculated in a temporal window size L, which is formulated as:

$$d_{ptz} = \frac{1}{L} \sum_{i=t-L}^{t} \|\vec{d_i}\| \tag{19}$$

where $\vec{d_i}$ is the displacement vector computed between the two frames I_{i-1} and I_i.

The presence of a PTZ effect is decided by a threshold test on the displacement average d_{ptz}. If d_{ptz} overpasses a given threshold ϵ then a PTZ scenario is started. Consequently, the learning rate is set to a high value α_{ptz} for the next N_{ptz} frames to allow the new background to be quickly absorbed into the background model.

F. The overall background subtraction algorithm

Suppose that at time t, the model in Eq. (2) has generated k_f and k_b components associated with the foreground (f) and background (b), respectively, where $k_f + k_b = K$. The foreground and background probabilities are given by $p_{t,f}(\vec{X}_t)$ and $p_{t,b}(\vec{X}_t)$, respectively.

Firstly, the presence of PTZ camera effect is checked as explained in Section III-E. If a PTZ scenario is started, then a high value is assigned to the learning rate. Otherwise, the learning rate is updated as explained in Section III-D. At each pixel in frame I_{t+1}, we may have one of the following scenarios:

1) If the vector $\vec{X}_{t+1}$ is matched with one of the mixture components, the matched component parameters ($\omega_{k,t+1}$, $\mu_{k,t+1}$, $\sigma_{k,t+1}$ and $\lambda_{k,t+1}$) are updated using Eqs. (4) to (11) and only the weight value of the unmatched components ($\omega_{\overline{k},t+1}$) is updated using Eq. (4). The CCM and persistence values are updated as detailed in Section III-B. Finally, the matched component's label (b:background or f:foreground) is assigned to the pixel.

2) If no match is found, a new component $K+1$ is created for the mixture model with parameters set as follows: $\omega_{K+1,t+1} = \alpha$, $\mu_{K+1,d,t+1} = \vec{X}_{t+1}$, $\sigma_{K+1,d,t+1} = \sigma_0$ and $\lambda_{K+1,d,t+1} = \lambda_0$, where σ_0 and λ_0 are initial scale and shape parameters, respectively. The CCM and persistence entries that correspond to this component are reset to zero.

Next, the mixture components are sorted in descending order of temporal co-occurrence weights ($\pi_{i,t+1}$) computed by Algorithm 1 and the new background temporal model $p_{t+1,b}(.)$ is formed using the first B largest mixture components, where

$$B = \arg\min_b \left(\sum_{i=1}^{b} \pi_{i,t+1} > T \right). \tag{20}$$

The threshold T is the minimum portion of data considered to belong to the dynamic background (typically $T = 0.8$). At this step, the reference frame used by the spatial module is constructed by taking the mean parameter of the first component among the B components resulting in Eq. (20) as the pixel value (see Eq. (13)).

Next, based on the two temporal and spatial probabilities (i.e. $p_{t,f}$ and $p_{s,f}$, respectively), the current pixel will be assigned label f if:

$$\mathcal{S}_{fcm} * p_{s,f}(\vec{X}_{t+1}) + (1 - \mathcal{S}_{fcm}) * p_{t,f}(\vec{X}_{t+1}) \geq \delta \tag{21}$$

where δ is a threshold and $\mathcal{S}_{fcm}$ (spatial foreground coherence map) is a smoothed map of the correlation map $p_{s,f}$ using an average filter of 5×5 pixels size. Otherwise, it is assigned label b (i.e., a potential shadow or highlight).

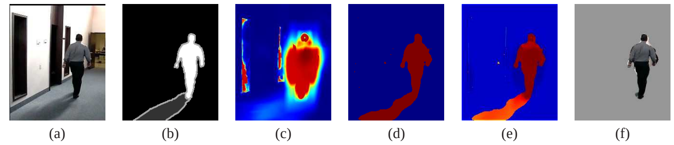


Fig. 4. Shadow removing using correlation analysis (Sample frame #1200 from the 'CDnet' dataset, 'shadow' category, 'cubicle' sequence). (a) Original frame. (b) Ground truth frame. (c) Spatial correlation map. (d) Temporal map. (e) Temporal and spatial combination map. (f) Final detection.

Module	Required parameters
MoGG	$K = 7$, $\tau = 0.002$, $\sigma_0 = 20.0$, $\lambda_0 = 1.0$, $\phi_{low} = 10^{-5}$, $\phi_{avg} = 10^{-3}$, $\phi_{high} = 0.05$
Co-occurrence	$M = 1$, $\nu = 0.5$, $T = 0.80$
NCC analysis	$N_1 = 11$, $N_2 = 35$, $N_3 = 65$, $w_1 = 1.10$, $w_2 = 0.82$, $\delta = 0.9$
Histogram matching	$W_1 = 13$, $W_2 = 33$, $W_3 = 53$, $N_{bins} = 16$, $w_3 = 0.25$, $w_4 = 0.01$, $T_1 = 0.2^3$, $T_2 = 0.5^3$
PTZ detection	$L = 10$, $\epsilon = 6$, $N_{ptz} = 15$, $\alpha_{ptz} = 0.1$

TABLE III

PARAMETER SETTING OF THE PROPOSED ALGORITHM USED IN OUR EXPERIMENTATIONS.

Finally, a post-processing step is performed through the binary masks generated after the temporal and spatial combination. The "salt and pepper noise" is a common problem that arise in BS. First, we apply the median filter to reduce this type of noise. Then, a morphological correction is applied to smooth silhouettes and fill their "internal holes". This allows also to remove small wrongly detected artifacts from the resulting binary masks.

The combination of temporal and spatial models is demonstrated in Fig. 4, where the correlation analysis is used to remove casting shadows. We can see that the shadow cast by the walking person is wrongly classified as foreground by the temporal model. However, by using the spatial information, the shadow is detected and removed. On the other hand, the temporal model helps to detect precisely the walking person silhouette.

IV. EXPERIMENTAL RESULTS

We conducted experiments using three different benchmark datasets: the CDnet dataset [60], the SABS dataset [6] and the MSVS [4] dataset. To evaluate the merit of our approach, internal comparisons are done between the temporal and spatial modules of the proposed approach. Moreover, the proposed approach results are compared with those produced by new state-of-the-art methods. The choice of optimal parameters is critical to the evaluation task, therefore, preliminary experiments have been conducted to adjust the best set of parameters. Tab. III shows the optimal parameters selected for the proposed algorithm.

A. Used evaluation datasets

We run our algorithm on the following datasets:

1) **The Change Detection Dataset (CDnet)**: This dataset has been proposed recently [60] to address the shortcomings of previous datasets regarding challenges and ground truth availability. It provides 53 videos that have been acquired in different scenarios: baseline, dynamic backgrounds, camera jitter, shadows, intermittent object motion, thermal, bad weather, low frame-rate, night, PTZ camera motion and air turbulence. They

are grouped into 11 categories according to the type of challenge each video exhibits.

2) **The Stuttgart Artificial BS Dataset (SABS)**: This dataset proposed in [6] contains synthetic videos for pixel-wise evaluation of BS methods. It includes 9 realistic scenarios: basic, dynamic background, bootstrapping, darkening, light switch, noisy night, camouflage, no Camouflage and Video compression. Each video sequence exhibits one or more challenges such as shadows, waving trees and traffic lights. High-quality ground truth annotation is provided as color-coded foreground masks for every frame of each test video.

3) **The Multispectral Video Sequences Dataset (MSVS)**: The MSVS dataset [4] is the first BS dataset that uses multispectral band (MSB) video data. It is composed of 5 multispectral video sequences containing between 250 and 2300 frames of the size of 658×491 for each video frame. Each video sequence is composed of 7 spectral bands, six in the visible spectrum and one in the near infrared (NI). This video dataset represents different challenges such as gradual illumination changes, shadows, and camouflage effects. The video sequences and their ground truth data are available online.

To conduct a quantitative comparison between the proposed model and state-of-the-art approaches, we use the evaluation metrics provided by the CDnet dataset [60]. The seven metrics used are Recall (Re), Specificity (Sp), False Positive Rate (FPR), False Negative Rate (FNR), Percentage of Wrong Classifications (PWC), Precision (Pr) and F-measure (F).

B. Quantitative evaluation of the temporal/spatial modules

We conduct a separate quantitative evaluation of the proposed temporal and spatial modules to illustrate the improvement that can be achieved by the integration of each module. The bar chart of the Fig. 5 highlights the results obtained by the application of the proposed modules separately for the ’shadow’ and ’dynamicBackground’ categories from the CDnet [56] dataset along with five RGB and MSB video sequences from the MSVS dataset [4]. We note that the videos in the same category may include different kinds of challenges. As shown in the Fig. 5, three modules have been evaluated, (1) the temporal model without co-occurrence analysis (*MoGG*), (2) the temporal model with co-occurrence analysis (*MoGG+CooC*), and (3) the overall approach including the combination of temporal co-occurrence information with spatial analysis (*MoGG+CooC+Spatial*).

Fig. 5 reports that the co-occurrence information improves the precision for the ’dynamicBackground’ category by removing false detections results from the rapidly switching background that characterizes videos in this category. However, the use of MoGGs with co-occurrence gives similar results in terms of F-measure compared to using only the temporal MoGGs for the ’shadow’ category. We can also observe that the use of spatial information has improved both recall and precision metrics in the two categories. Considering the MSVS dataset, the experimental evaluations given in the Fig. 5 show that the use of co-occurrence analysis increases the performance of the temporal model. Furthermore, the results given by the temporal modules are significantly improved by integrating the spatial modules. This fact can be observed for both RGB and multispectral (MSB) video sequences.

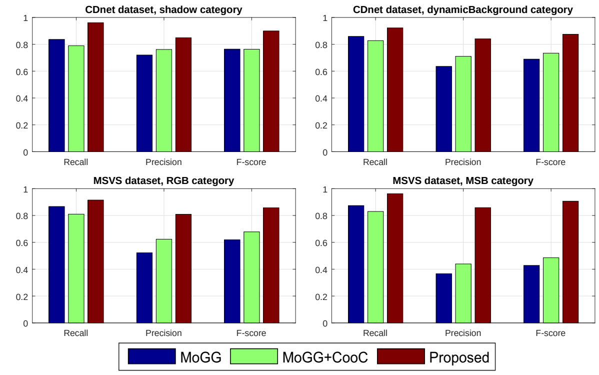


Fig. 5. Comparison of evaluation metrics obtained by the different proposed modules for the CDnet and the MSVS dataset.

The observed performance improvement has been carried out thanks to the combination of temporal and spatial modules that cooperate each other to cope with several challenges. Firstly, the MoGGs modeling has been improved by co-occurrence and adaptive updating of the learning rate to detect and remove false positives in background dynamics scenarios such as noise and camera jitter. Secondly, soft changes in illumination and shadow are absorbed to the background by the MoGG updating mechanism along with the multi-scale correlation analysis. Thirdly, the convergence of the background model is accelerated using an adaptive learning rate which is updated using the correlation and histogram spatial information.

C. Overall evaluation of the proposed method

1) The CDnet Dataset: Fig. 6 contains the performance metrics obtained by the application of the proposed approach for the 53 videos of the CDnet dataset (The overall results for the CDnet dataset are given in Tab. VII. The graphics in Fig. 6 show that the proposed approach gives competitive results for several categories such as ’shadow’, ’dynamicBackground’, ’cameraJitter’, and ’PTZ’. We can note that the majority of videos show F-Measure metrics above 80% which indicates that the proposed approach is efficient in dealing with shadows and dynamic backgrounds. However, one can observe that the proposed approach gives relatively low scores regarding some categories such as ’intermittentObjectMotion’ and ’nightVideos’. In fact, most of methods in the literature fail to give satisfactory results for these categories for several reasons. For videos containing intermittent objects, it is hard for some videos to build an object-free background that enables good object detections. In the ’parking’ video, for example, this difficulty has prevented detecting the car in the majority of video frames which caused a great number of false negatives (Re=0.06). For night videos, erroneous detections can be caused by strong illuminations that saturate pixels (e.g., car headlights) or in videos containing low contrast between objects and the background.

For the ’PTZ’ category, the proposed algorithm gives good results for three sequences, namely: the ’continuousPan’, the ’intermittentPan’, and the ’twoPositionPTZCam’ sequences. However, the proposed technique fails to detect the PTZ scenario for the ’zoomInZoomOut’ sequence. This is because, in this video, the displacement between two consecutive frames is very small and; therefore, the PTZ detection and processing can not be started throughout all the frames of this sequence.

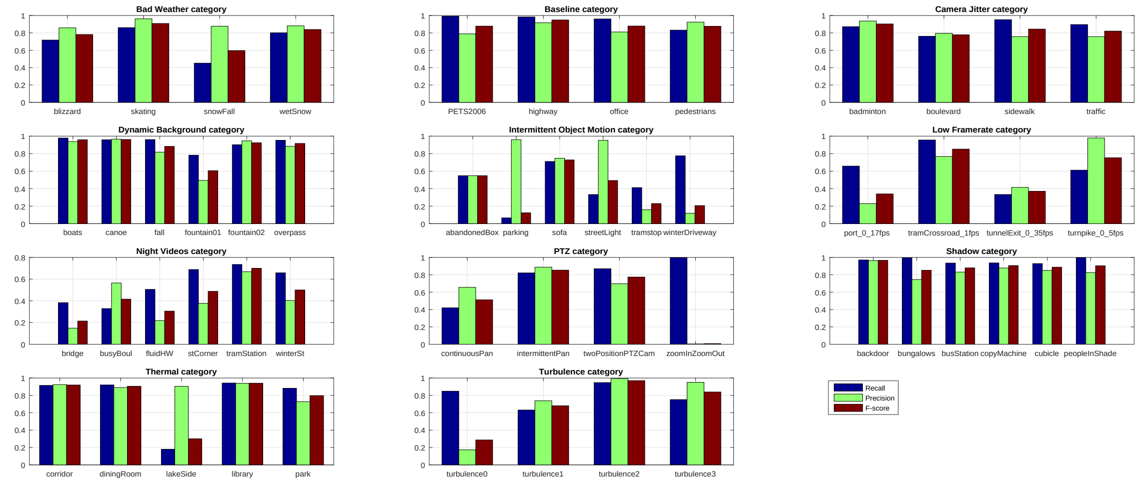


Fig. 6. BS scores obtained by the proposed algorithm on all CDnet videos.

To show the merit of our algorithm to deal with the shadow, dynamic background, camera jitter, and PTZ challenges, comparison tests have been conducted on all the videos of of the CDnet dataset. The graphic in Fig. 7 shows a comparison evaluation between category average values of the recall, precision and F-measure metrics obtained by a set of BS methods. The set of compared methods includes six state-of-the-art methods, namely: SuBSENSE (Self-Balanced SENsitivity SEgmenter) [7], BinWang [55], RMoG (Region based Mixture of Gaussians) [53], GMM (Gaussian Mixture Model) by Grimson et *al.* [50], GMM (Gaussian Mixture Model) by Zivkovic [65], KDE (Kernel Density Estimation) [12], and the proposed approach.

From Fig. 7, we can observe that for the ’shadow’ and ’cameraJitter’ categories, the proposed approach gives the best results and outperforms most of the compared methods in terms of recall and F-measure. However, for the ’dynamicBackground’ category, the proposed method can be ranked second after the FTSG method [56] which was ranked first at the CDnet 2014 overall challenges. For the ’PTZ’ category, our method gives the highest precision and F-measure metrics along with good results in terms of the recall metric. This is can be done due to the global cross-correlation technique that has been added to deal with PTZ camera sequences by finding large displacement between consecutive frames.

Tab. IV shows the results obtained by the application of the proposed algorithm on all the CDnet dataset categories as well as the overall results computed according to the CDnet evaluation methodology [60]. To make a comparison with a method combining temporal and spatial information, Tab. V shows results obtained by application of a patch-based approach proposed in [42]. From Tabs. IV and V, we can observe that our method achieves better results in terms of overall F-measure and overall precision metrics than [42]. However, the patch-based approach surpasses our method in only the Recall metric. This is due to the fact that [42] does not deal explicitly with background subtraction challenges such as illumination changes, shadow, dynamic background and PTZ challenges.

Fig. 8 shows a sample of foreground masks generated by each compared BS method. The original frames and ground truth masks are displayed in the first and second rows, respectively. The third row shows the foreground masks given by our proposed method. The rest of BS methods are shown in the other rows. Columns of Fig. 8 represent 6 sample sequences from the studied categories. These sequences are as follows: ’traffic’ and ’sidewalk’ from the ’cameraJitter’ category, ’canoe’ and ’fountain01’ from the ’dynamicBackground’ category and ’busStation’ and ’cubicle’ from the ’shadow’ category.

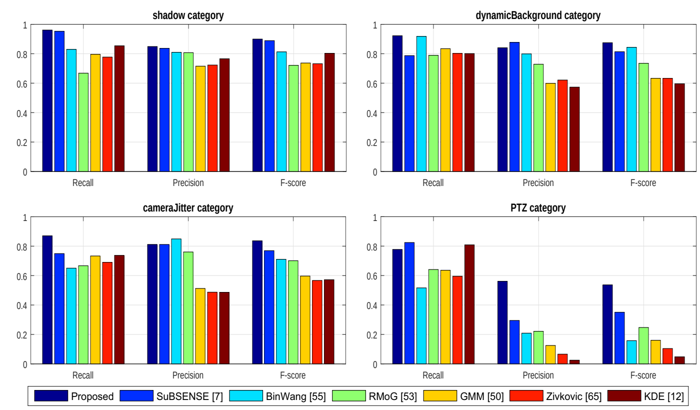


Fig. 7. Evaluation metrics obtained by state-of-the-art as well as proposed method for the ’shadow’, ’dynamicBackground’, ’cameraJitter’ and ’PTZ’ categories from CDnet dataset.

Category/Metric	Re	Sp	FPR	FNR	PWC	Pr	F
badWeather	0.7079	0.9987	0.0013	0.2921	0.5504	0.8947	0.7815
baseline	0.9419	0.9934	0.0066	0.0581	0.7592	0.8600	0.8956
cameraJitter	0.8704	0.9903	0.0097	0.1296	1.5086	0.8118	0.8365
dyn. Back.	0.9224	0.9987	0.0013	0.0776	0.1868	0.8408	0.8749
int.Obj.Mot.	0.4744	0.9077	0.0923	0.5256	11.8726	0.5810	0.3885
lowFramerate	0.6396	0.9942	0.0058	0.3604	1.7770	0.5977	0.5785
nightVideos	0.5501	0.9812	0.0188	0.4499	2.8090	0.3969	0.4372
PTZ	0.6396	0.9941	0.0058	0.3603	1.7770	0.5977	0.5785
shadow	0.9610	0.9908	0.0092	0.0390	1.0220	0.8490	0.8997
thermal	0.7681	0.9930	0.0070	0.2319	1.3989	0.8771	0.7727
turbulence	0.7949	0.9977	0.0023	0.2051	0.3710	0.7136	0.6943
Overall	0.7643	0.9728	0.0271	0.2356	3.3484	0.7258	0.7001

TABLE IV
EVALUATION METRICS OBTAINED BY APPLICATION OF THE PROPOSED METHOD FOR ALL CDNET CATEGORIES AS WELL AS THE OVERALL RESULTS.

The videos in the first and second column of Fig. 8 are characterized by unstable cameras. We can observe that our method significantly avoids false positives caused by camera jitter and efficiently detects the car in the ’traffic’ sequence

Category/Metric	Re	Sp	FPR	FNR	PWC	Pr	F
badWeather	0.3939	0.9972	0.0028	0.6061	1.1915	0.8088	0.4557
baseline	0.9742	0.9966	0.0034	0.0258	0.4305	0.9064	0.9384
cameraJitter	0.8480	0.9856	0.0144	0.1520	2.0182	0.7203	0.7784
dyn. Back.	0.8982	0.9927	0.0073	0.1018	0.8253	0.6294	0.6823
int.Obj.Mot.	0.7444	0.8469	0.1531	0.2556	14.4944	0.4493	0.4955
lowFramerate	0.9177	0.9822	0.0178	0.0823	1.9655	0.5266	0.5887
nightVideos	0.8174	0.9600	0.0400	0.1826	4.2120	0.3871	0.4933
PTZ	0.7887	0.7613	0.2387	0.2113	23.8913	0.0491	0.0896
shadow	0.9792	0.9879	0.0121	0.0208	1.2570	0.7856	0.8669
thermal	0.3169	0.9920	0.0080	0.6831	6.1843	0.7035	0.3957
turbulence	0.8081	0.9992	0.0008	0.1919	0.2262	0.7483	0.7680
Overall	0.7715	0.9547	0.0453	0.2285	5.1542	0.6104	0.5957

TABLE V

EVALUATION METRICS OBTAINED BY THE APPLICATION OF THE REDDY METHOD [42] FOR ALL CDNET CATEGORIES AS WELL AS THE OVERALL RESULTS.

as well as the waiting person in the 'sidewalk' video. The dynamic background challenge is presented in columns 3 and 4. The 'fountain01' sequence contains a fountain and cars moving over the dynamic background. The 'canoe' sequence represents a water rippling scene. We can note that most of the false positives are eliminated by the proposed algorithm. The fifth column shows a sample frame from the 'busStation' sequence which consists of persons waiting in a bus station. This sequence is characterized by a hard shadow cast on the ground by the walking persons. Our approach detects accurately the walking man and separates a significant amount of shadow from the ground. The last column shows a sample frame from the 'cubicle' sequence. This video contains a person walking through a cubicle corridor. This sequence is characterized by a hard shadow cast on the ground by the walking person as well as some highlights on the cubicle walls. We can note that the proposed method detects the person and avoids false detections due to shadows. These sample frames clearly show the advantage of combining spatial and temporal information to deal with challenges such as casting shadows, illumination changes, dynamic backgrounds and camera jitter.

2) The SABS Dataset: To demonstrate the accuracy of the proposed model, experiments are conducted on the SABS dataset using the proposed method compared to the nine state-of-the-art methods cited in the SABS dataset website [1] which include: ViBe [3], SOBS [33], Zivkovic [65], and GMM [50]. In addition, we add two spatiotemporal BS approaches, namely: Spatiotemporal [9] and a recently published method: SuBSENSE [7].

The chart in Fig. 9 presents compared quantitative results in terms of the maximal F-measure as cited in [6]. We can observe that our method gives competitive results for almost all scenarios. More precisely, (in terms of F-measure metric) the proposed method outperforms the compared ones for 5 sequences out of 9 which are: the Basic, Dynamic Background, Bootstrapping, Camouflage and H.264 (40kbps). However, the proposed method presents the second or the third best F-measure evaluation for the 4 remaining sequences. Nonetheless, the average F-measure computed for all sequences is higher than the compared methods.

Fig. 10 shows some frames from the SABS dataset, the associated ground truth as well as the foreground masks obtained by the compared methods. It can be seen that the proposed model has effectively discriminated between backgrounds and moving objects. Indeed, thanks to adding multi-scale spatial

[1] http://www.vis.uni-stuttgart.de/~hoeferbn/bse/

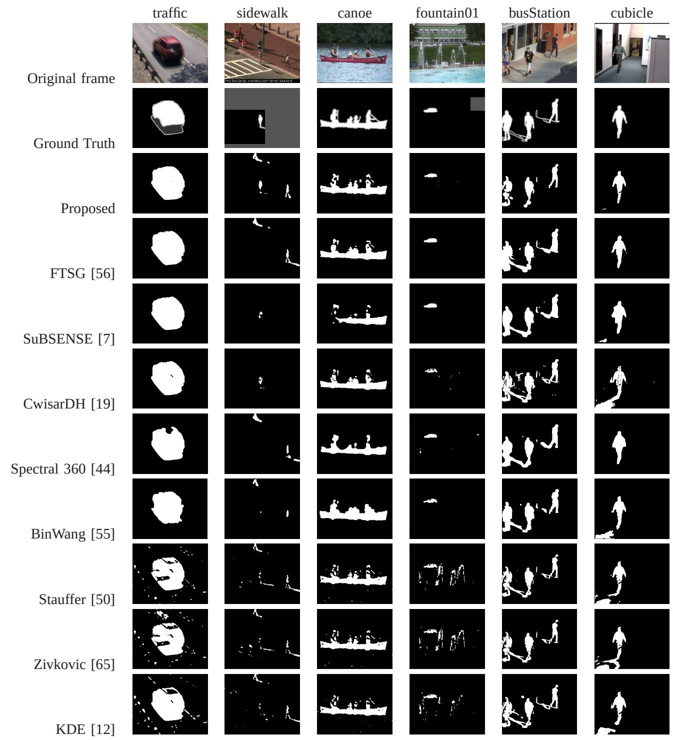


Fig. 8. BS masks obtained for sample frames from the CDnet dataset [60] by application of different compared methods.

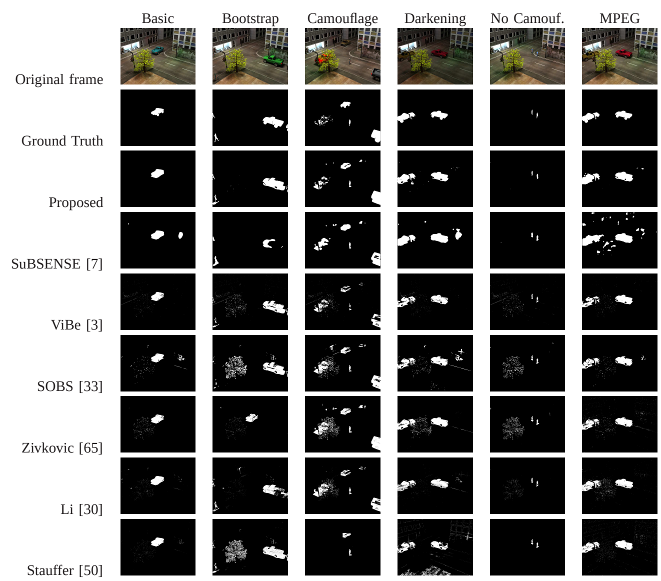


Fig. 10. BS masks obtained for sample frames from the SABS dataset [6] by application of different compared methods.

information, the proposed model has been able to reduce false positives generated by the illumination changes cast on the wall by the traffic lights. On the other hand, parasites produced by the waving tree, the Gaussian noise or due to compression artifacts are absorbed by the temporal co-occurrence module which has considerably improved precision.

3) The MSVS Dataset: We evaluate the proposed method on this multispectral dataset. Our purpose is to show the efficiency of the proposed approach by integrating multispectral information. In fact, the multispectral data can be handled in the proposed approach by adjusting the D parameter that appears in the Eq. (2),(14) and (17).

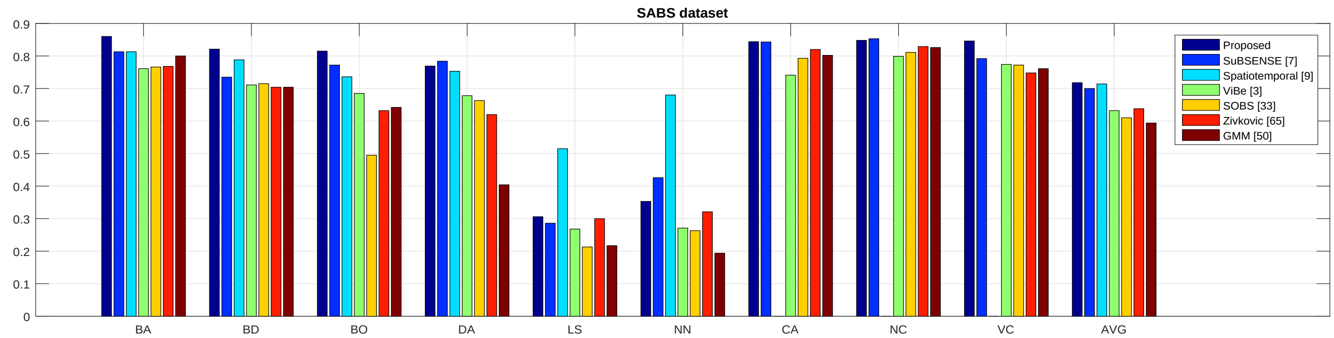


Fig. 9. F-measure metrics obtained by application of the compared methods for the SABS dataset sequences [6]. Each set of bars represents one sequence from the SABS dataset. From left to right: Basic (BA), Dynamic Background (DB), Bootstrap (BO), Darkening (DA), Light Switch (LS), Noisy Night (NN), Camouflage (CA), No Camouflage (NC) and Video Compression (VC) H.264 codec with bitrate 40kbps/s. The last set of bars (AVG) represents the average of F-measure metric computed for each method for all sequences.

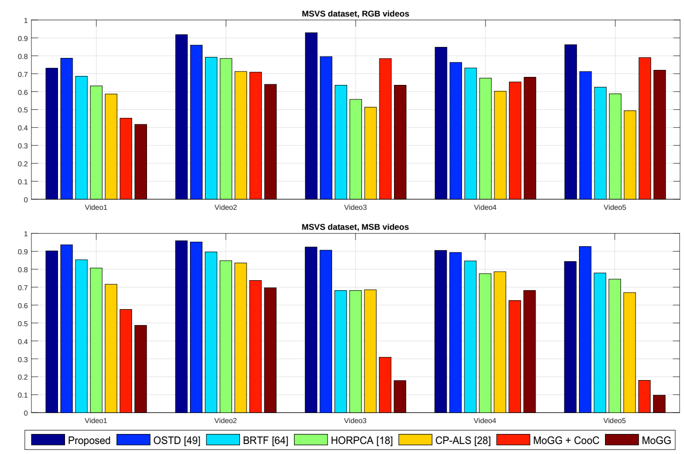


Fig. 11. F-measure metrics obtained by application of the compared methods for the MSVS dataset sequences [4].

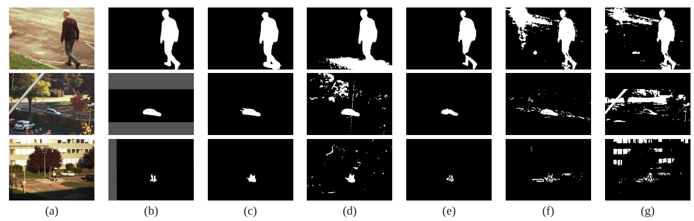


Fig. 12. BS binary masks obtained for sample frames selected from the MSVS dataset [4], [49] by application of the compared methods. (a) Original frame. (b) Ground truth frame. (c) Proposed approach. (d) MOGG. (e) OSTD [49]. (f) BRTF [64]. (g) HORPCA [18].

The graphic in Fig. 11 shows a quantitative comparison between the proposed approach and the state-of-the-art methods where the two types of input data are used: the multispectral bands (MSB) which has 7 channels ($D = 7$), and the usual trichromatic images with the 3 RGB channels ($D = 3$). The F-measure score is computed for each video by comparing our results with 6 other methods such as MoGG, MoGG+CooC, CP-ALS [28], HORPCA [18], BRTF [64], and OSTD [49]. Note that the results for the compared methods are obtained from [49].

Fig. 11 reports that the results obtained by the proposed approach and most of the compared methods for the MSB data are better than those obtained for the RGB ones. This may be due to fact that using the 7 multispectral bands (MSB) is more discriminant than using only 3 RGB channels. In terms of F-measure score, it can be observed from Fig. 11 that the proposed approach outperforms the compared approaches in 7 out of 10 videos and can be ranked secondly after the OSTD [49] approach in the rest of 3 videos.

Fig. 12 represents a visual comparison of background subtraction BS results obtained by the application of the proposed method along with the compared methods over 3 video sequences from the MSVS dataset [4]. It can be seen from Fig. 12 that most of the false positives generated by the state-of-the-art methods can be mitigated by the application of the overall proposed approach. For example, in the first row, most of the walking man shadow and the illumination changes on ground can be removed using the proposed approach. The same remark can also be applied for the false positives caused by the moving tree leaves in the second and third row of this figure.

D. Computational Time

Our tests were implemented using the MATLAB environment with some optimization using MEX C++ subroutines. On a PC with Intel Core i7 2.93 GHz CPU and 16 Go of RAM and MS Windows 7 operating system, the proposed prototype runs at about 5 fps for videos of RGB color frames with a size of 320×240 pixels. Most of the processing time is dedicated to the calculation of NCC and histogram distances along with the updating procedure for the temporal/co-occurrence model.

Tab. VI shows the computational time recorded in CPU time and required by application of each proposed module, the overall proposed algorithm and some compared BS methods. The first row of the Tab. VI represents the input frame size $W \times H \times D$ which correspond to the width, the height and the number of channels of each frame, respectively. The second row of Tab. VI includes the computational time of the four modules constituting our approach, as well as the overall time. The last row includes the CPU time required by application of a set of compared BS methods, namely: KDE [12], GMM [50], OSTD [49], CP-ALS [28], BRTF [64], and HORPCA [18].

V. Conclusion

A statistical approach for video background subtraction (BS) by combining temporal and spatial information is presented. The two types of information are fused in an algorithm that performs efficient BS in the presence of cast shadows, illumination

Method/Frame size	$180 \times 100 \times 1$	$180 \times 100 \times 3$	$320 \times 240 \times 3$	$658 \times 492 \times 3$
Temporal MoGG	0.007	0.027	0.113	0.421
Co-occurrence	0.011	0.011	0.054	0.229
Correlation analysis	0.002	0.005	0.022	0.122
Histogram matching	0.013	0.041	0.144	0.481
Proposed	0.032	0.085	0.333	1.254
KDE [12]	0.001	0.003	0.020	0.050
GMM [50]	0.002	0.007	0.037	0.118
OSTD [49]	0.009	0.038	0.120	0.699
CP-ALS [28]	0.120	0.411	1.290	5.437
BRTF [64]	0.073	0.220	2.250	9.216
HORPCA [18]	0,495	1,487	7.096	43.924

TABLE VI
COMPUTATIONAL TIME IN SECONDS FOR EACH FRAME REQUIRED BY EACH MODULE, THE PROPOSED APPROACH AND SOME STATE-OF-THE-ART METHODS.

changes, complex background dynamics and PTZ effects. Our algorithm achieves accurate foreground detections compared to well-known methods. Future work will address speeding-up the proposed algorithm as well as the analysis of hard shadows and other background subtraction challenging problems such as camouflage and intermittent object motion.

VI. ACKNOWLEDGMENT

This work has been completed thanks to the support of the Natural Sciences and Engineering Research Council of Canada.

APPENDIX A
ONLINE UPDATE EQUATIONS FOR MOGG MODEL PARAMETERS

Following [1], the online estimations given in the Eq. (8) and (11) can be derived as follows:

A. The location parameter μ:

We have the following formula for estimating the location parameter at time $t+1$ [1]:

$$\mu(t+1) = \frac{\sum_{i=1}^{t+1} |x_i - \mu|^{\lambda-2} x_i}{\sum_{i=1}^{t+1} |x_i - \mu|^{\lambda-2}}. \tag{22}$$

Therefore, we define:

$$\alpha(t) = \sum_{i=1}^{t} |x_i - \mu|^{\lambda-2} x_i \tag{23}$$

and

$$\beta(t) = \sum_{i=1}^{t} |x_i - \mu|^{\lambda-2}, \tag{24}$$

and by replacing Eqs. (23) and (24) in Eq. (22), we have

$$\mu(t+1) = \frac{\alpha(t) + |x_{t+1} - \mu|^{\lambda-2} x_{t+1}}{\beta(t) + |x_{t+1} - \mu|^{\lambda-2}} \tag{25}$$

B. The scale parameter σ:

The scale parameter at $t+1$ is defined by the formula:

$$\sigma(t+1) = \left[\frac{\lambda A(\lambda)}{t+1} \sum_{i=1}^{t+1} |x_i - \mu|^{\lambda}\right]^{1/\lambda}. \tag{26}$$

By replacing the inside of $\sigma(t+1)$ with $\sigma(t)$, we get what follows:

$$\sigma(t+1) = \left[\frac{\lambda A(\lambda)}{t+1} \left[\frac{t\sigma(t)^{\lambda}}{\lambda A(\lambda)} + |x_{t+1} - \mu|^{\lambda}\right]\right]^{1/\lambda} \tag{27}$$

Therefore, we obtain:

$$\sigma(t+1) = \left[(1-\phi)\sigma(t)^{\lambda} + \phi\lambda A(\lambda)|x_{t+1} - \mu|^{\lambda}\right]^{1/\lambda} \tag{28}$$

where: $\phi = \frac{1}{1+t}$ is the learning factor.

C. The mean of centered absolute value (MAV):

The mean of centered absolute value of the MoGG distribution (MAV) can be obtained as follows

$$\begin{aligned}
\mathbb{E}_{t+1}\big[\,|\,X\,|\,\big] &= \frac{1}{(t+1)} \sum_{i=1}^{t+1} |\,x_i - \mu\,| \\
&= \frac{1}{(t+1)} \Big(\sum_{i=1}^{t} |\,x_i - \mu\,| + |\,x_{t+1} - \mu\,| \Big) \\
&= \frac{t}{t(t+1)} \sum_{i=1}^{t} |\,x_i - \mu\,| + \frac{1}{t+1}\,|\,x_{t+1} - \mu\,| \\
&= (1-\phi)\mathbb{E}_t\big[\,|\,X\,|\,\big] + \phi\,|\,x_{t+1} - \mu\,|
\end{aligned} \tag{29}$$

where $\phi = \frac{1}{1+t}$ is the learning factor.

Category	Video	Re	Sp	FPR	FNR	PWC	Pr	F
badWeather	blizzard	0.7177	0.9986	0.0014	0.0033	0.0047	0.8584	0.7818
	skating	0.8607	0.9982	0.0018	0.0073	0.0086	0.9620	0.9085
	snowFall	0.4518	0.9995	0.0005	0.0044	0.0048	0.8771	0.5964
	wetSnow	0.8013	0.9986	0.0014	0.0026	0.0039	0.8814	0.8394
baseline	PETS2006	0.9905	0.9965	0.0035	0.0001	0.0036	0.7888	0.8782
	highway	0.9842	0.9944	0.0056	0.0010	0.0063	0.9165	0.9491
	office	0.9608	0.9833	0.0167	0.0029	0.0182	0.8105	0.8792
	pedestrians	0.8320	0.9993	0.0007	0.0017	0.0023	0.9242	0.8757
cameraJitter	badminton	0.8721	0.9979	0.0021	0.0045	0.0064	0.9368	0.9033
	boulevard	0.7610	0.9904	0.0096	0.0118	0.0204	0.7955	0.7779
	sidewalk	0.9524	0.9918	0.0082	0.0013	0.0092	0.7577	0.8440
	traffic	0.8962	0.9809	0.0191	0.0069	0.0244	0.7571	0.8208
dyn.Back.	boats	0.9786	0.9996	0.0004	0.0001	0.0005	0.9386	0.9582
	canoe	0.9583	0.9988	0.0012	0.0015	0.0027	0.9657	0.9620
	fall	0.9608	0.9961	0.0039	0.0007	0.0045	0.8172	0.8832
	fountain01	0.7825	0.9993	0.0007	0.0002	0.0008	0.4940	0.6056
	fountain02	0.9020	0.9999	0.0001	0.0002	0.0003	0.9463	0.9236
	overpass	0.9524	0.9983	0.0017	0.0006	0.0023	0.8833	0.9166
interm.Obj.Mot.	abandonedBox	0.5483	0.9772	0.0228	0.0228	0.0434	0.5484	0.5484
	parking	0.0665	0.9998	0.0002	0.0782	0.0724	0.9593	0.1243
	sofa	0.7108	0.9890	0.0110	0.0132	0.0231	0.7470	0.7284
	streetLight	0.3329	0.9992	0.0008	0.0340	0.0332	0.9527	0.4933
	tramstop	0.4121	0.5246	0.4754	0.1286	0.4956	0.1594	0.2299
	winterDriveway	0.7757	0.9567	0.0433	0.0017	0.0446	0.1191	0.2065
lowFr.rate	port_0_17fps	0.6578	0.9993	0.0007	0.0001	0.0008	0.2293	0.3400
	tramCrossroad_1fps	0.9560	0.9918	0.0082	0.0013	0.0092	0.7680	0.8518
	tunnelExit_0_35fps	0.3334	0.9867	0.0133	0.0188	0.0312	0.4145	0.3696
	turnpike_0_5fps	0.6113	0.9990	0.0010	0.0312	0.0299	0.9791	0.7527
nightVideos	bridgeEntry	0.3835	0.9684	0.0316	0.0089	0.0399	0.1485	0.2141
	busyBoulvard	0.3289	0.9907	0.0093	0.0246	0.0326	0.5645	0.4157
	fluidHighway	0.5057	0.9741	0.0259	0.0071	0.0325	0.2188	0.3054
	streetCornerAtNight	0.6887	0.9943	0.0057	0.0015	0.0072	0.3775	0.4877
	tramStation	0.7353	0.9897	0.0103	0.0075	0.0173	0.6683	0.7002
	winterStreet	0.6583	0.9703	0.0297	0.0104	0.0389	0.4036	0.5004
PTZ	continuousPan	0.4199	0.9986	0.0014	0.0037	0.0050	0.6560	0.5120
	intermittentPan	0.8221	0.9985	0.0015	0.0025	0.0040	0.8879	0.8537
	twoPositionPTZCam	0.8701	0.9942	0.0058	0.0020	0.0077	0.6972	0.7741
	zoomInZoomOut	0.9979	0.4324	0.5676	0.0000	0.5664	0.0037	0.0074
shadow	backdoor	0.9712	0.9992	0.0008	0.0006	0.0013	0.9633	0.9672
	bungalows	0.9954	0.9783	0.0217	0.0003	0.0207	0.7455	0.8525
	busStation	0.9351	0.9927	0.0073	0.0025	0.0094	0.8313	0.8802
	copyMachine	0.9369	0.9903	0.0097	0.0047	0.0134	0.8782	0.9066
	cubicle	0.9292	0.9967	0.0033	0.0014	0.0046	0.8500	0.8878
	peopleInShade	0.9983	0.9874	0.0126	0.0001	0.0120	0.8258	0.9039
thermal	corridor	0.9154	0.9974	0.0026	0.0029	0.0053	0.9244	0.9198
	diningRoom	0.9207	0.9894	0.0106	0.0075	0.0165	0.8906	0.9054
	lakeSide	0.1801	0.9996	0.0004	0.0160	0.0161	0.9041	0.3003
	library	0.9424	0.9853	0.0147	0.0138	0.0229	0.9388	0.9406
	park	0.8821	0.9931	0.0069	0.0024	0.0091	0.7274	0.7973
turbulence	turbulence0	0.8486	0.9923	0.0077	0.0003	0.0079	0.1728	0.2871
	turbulence1	0.6316	0.9991	0.0009	0.0014	0.0023	0.7379	0.6806
	turbulence2	0.9478	1.0000	0.0000	0.0000	0.0000	0.9937	0.9702
	turbulence3	0.7517	0.9994	0.0006	0.0041	0.0046	0.9502	0.8393

TABLE VII
BS SCORES OBTAINED BY THE PROPOSED ALGORITHM ON ALL CDNET VIDEOS.

REFERENCES

[1] M.S. Allili, N. Bouguila and D. Ziou. Finite General Gaussian Mixture Modeling and Application to Image and Video Foreground Segmentation. *J. of Electronic Imaging*, 17(1):013005, 2008.

[2] M.S. Allili, D. Ziou, N. Bouguila and S. Boutemedjet. Image and Video Segmentation by Combining Unsupervised Generalized Gaussian Mixture Modeling and Feature Selection. *IEEE TCSVT*, 20(10): 1373-1377, 2010.

[3] O. Barnich and M.V. Droogenbroeck. Vibe: A Powerful Random Technique to Estimate the Background in Video Sequences. *ICASSP*, 945-948, 2009

[4] Y. Benezeth, D. Sidibé, and J. B. Thomas. BS with Multispectral Video Sequences. *IEEE ICRA*, 1-5, 2014.

[5] T. Bouwmans. Traditional and Recent Approaches in Background Modeling for Foreground Detection: An Overview. *Computer Science Review*, 11(12):31-66, 2014.

[6] S. Brutzer, B. Höferlin and G. Heidemann. Evaluation of BS Techniques for Video Surveillance. *IEEE CVPR*, 1937-1944, 2011.

[7] P.-L. St-Charles, G.-A. Bilodeau and R. Bergevin. *SuBSENSE: A Universal Change Detection Method With Local Adaptive Sensitivity*. *IEEE TIP*, 24(1):359-373, 2015.

[8] L. Chen, J. Shen, B. Ni and L. Shao. Robust video segmentation using dense trajectories. *IEEE Trans. on Multimedia*, 17(12):2225-2234, 2015.

[9] M. Chen, Q. Yang, Q. Li, G. Wang, and M.-H. Yang. Spatiotemporal BS Using Minimum Spanning Tree and Optical Flow. *ECCV*, 521-534, 2014.

[10] B. Dey and M. K. Kundu. Robust BS for Network Surveillance in H.264 Streaming Video. *IEEE TCSVT*, 23(10):1695-1703, 2013.

[11] S. E. Ebadi and E. Izquierdo. Foreground Segmentation via Dynamic Tree-Structured Sparse RPCA. *ECCV*, 314329, 2016.

[12] A. Elgammal, D. Harwood and L. Davis. Non-parametric Model for BS. *ECCV*, 751-767, 2000.

[13] T. Elguebaly and N. Bouguila. BS Using Finite Mixtures of Asymmetric Gaussian Distributions and Shadow Detection. *MVA*, 25(5): 1145-1162, 2014.

[14] J. Feng, H. Xu and S. Yan. Online robust PCA via stochastic optimization. *NIPS*, 404-412, 2013.

[15] A. Ferone and L. Maddalena. Neural background subtraction for PTZ cameras. *IEEE Trans. on Sys., Man, and Cyb.*, 44(5):571-579, 2014.

[16] W. T. Freeman, K. Tanaka, J. Ohta and K. Kyuma. Computer Vision for Computer Games. *IEEE FG*, 100-105, 1996.

[17] H. Fu, D. Xu, B. Zhang, S. Lin and R. K. Ward. Object-Based Multiple Foreground Video Co-Segmentation via Multi-State Selection Graph. *IEEE TIP*. 24(11):3415-3424, 2015.

[18] D. Goldfarb and Z. Qin. Robust low-rank tensor recovery: Models and algorithms. *SIMAX*, 35(1):225-253, 2014.

[19] M. De Gregorio and M. Giordano. Change Detection with Weightless Neural Networks. *IEEE Workshop on CD*, 409-413, 2014.

[20] M. Grundmann, V. Kwatra, M. Han and I. Essa. Efficient hierarchical graph-based video segmentation. *IEEE CVPR*, 2141-2148, 2010.

[21] M. Heikkila and M. Pietika. A Texture-Based Method for Modeling the Background and Detecting Moving Objects. *IEEE TPAMI*, 28(4):657-662, 2006.

[22] M. Hofmann, P. Tiefenbacher and G. Rigoll. Background Segmentation with Feedback: The Pixel-Based Adaptive Segmenter. *IEEE CVPR Workshop*, 38-43, 2012.

[23] W. Hu, X. Li, X. Zhang, X. Shi, S. Maybank and Z. Zhang. Incremental tensor subspace learning and its applications to foreground segmentation and tracking. *Int. J. Comput. Vis.*, 91(3):303-327, 2011.

[24] P. Jodoin, M. Mignotte and J. Konrad, Statistical BS Using Spatial Cues. *IEEE TCSVT*, 17(12):1758-1763, 2007.

[25] P. Kaewtrakulpong and R. Bowden. An Improved Adaptive Background Mixture Model for Realtime Tracking with Shadow Detection. *European Workshop on AVBSS*, 1-5, 2001.

[26] J. Kato, T. Watanabe, S. Joga, J. Rittscher, and A. Blake. An HMM-based Segmentation Method for Traffic Monitoring Movies. *IEEE TPAMI*, 24(9):1291-1296, 2002.

[27] K. Kim, T. Chalidabhongse, D. Harwood and L. Davis. Real-Time Foreground-Background Segmentation Using Codebook Model. *Real-Time Imaging*, 11(3):172-185, 2005.

[28] T. G. Kolda and B.W. Bader. Tensor decompositions and applications. *SIAM Review*, 2008.

[29] Y. J. Lee, J. Kim and K. Grauman. Key-Segments for Video Object Segmentation. *IEEE ICCV*, 1995-2002, 2011.

[30] L. Li, W. Huang, I. Gu and Q. Tian. Foreground Object Detection From Videos Containing Complex Background. *ACM Int'l Conf. on Multimedia*, 2-10, 2003.

[31] L. Lin, Y. Xu, X. Liang and J. Lai. Complex Background Subtraction by Pursuing Dynamic Spatio-Temporal Models. *IEEE TIP*, 23(7):3191-3202, 2014.

[32] T. Ma and L. J. Latecki. Maximum Weight Cliques with Mutex Constraints for Video object Segmentation. *CVPR*, 670-677, 2012.

[33] L. Maddalena and A. Petrosino. A Self-Organizing Approach to BS for Visual Surveillance Applications. *IEEE TIP*, 17(7):1168-1177, 2008.

[34] V. Mahadevan and N. Vasconcelos. Spatiotemporal Saliency in Dynamic Scenes. *IEEE TPAMI* 32(1):171-177, 2010.

[35] J. M. McHugh, J. Konrad, V. Saligrama and P.-M. Jodoin. Foreground-Adaptive BS. *IEEE SPL*, 16(5):390-393, 2009.

[36] N. Oliver, B. Rosario and A. Pentland. A Bayesian Computer Vision System for Modeling Human Interactions. *IEEE TPAMI*, 22(8):831-843, 2000.

[37] A. Papazoglou and V. Ferrari. Fast Object Segmentation in Unconstrained Video. *IEEE ICCV*, 1777-1784, 2013.

[38] M. Piccardi. BS Techniques: A Review. *IEEE ICSMC*, 3099-3104, 2004.

[39] F. Porikli. Integral Histogram: a Fast Way to Extract Histograms in Cartesian Spaces. *IEEE CVPR*,829-836, 2005.

[40] A. Prati, I. Mikic, M. M. Trivedi and R. Cucchiara. Detecting Moving Shadows: Algorithms and Evaluation. *IIEEE TPAMI*, 25(7):918-923, 2003.

[41] V. Reddy, C. Sanderson and B. C. Lovell. A Low-Complexity Algorithm for Static Background Estimation from Cluttered Image Sequences in Surveillance Contexts. *J. Image and Video Processing*, Article ID 164956, 1-14, 2011.

[42] V. Reddy, C. Sanderson and B. C. Lovell. Improved Foreground Detection via Block-Based Classifier Cascade With Probabilistic Decision Integration. *IEEE TCSVT*, 23(1):83-93, 2013.

[43] J. Rittscher, J. Kato, S. Joga, and A. Blake. A Probabilistic Background Model for Tracking. *European Conf. on Computer Vision*, 336-350, 2000.

[44] M. Sedky, M. Moniri and C. C. Chibelushi. Spectral-360: A physical-Based Technique for Change Detection. *IEEE Workshop on CD*, 405-408, 2014.

[45] M. Shah, J. D. Deng and B. J. Woodford. Video Background Modeling: Recent Approaches, Issues and our Proposed Techniques. *MVA*, 25(5):1105-1119, 2014.

[46] K. Sharifi and A. Leon-Garcia. Estimation of shape parameter for generalized Gaussian distibution in subband decomposition of video. *IEEE TCSVT*, 5(1):52-56, 1995.

[47] Y. Sheikh, O. Javed and T. Kanade. Background Subtraction for Freely Moving Cameras. *IEEE ICCV*, 1219-1225, 2009.

[48] J. Shen, Y. Du and X. Li. Interactive segmentation using constrained Laplacian optimization. *IEEE TCSVT*, 24(7):1088-1100, 2014.

[49] A. Sobral, S. Javed, S. K. Jung, T. Bouwmans, E. Zahzah. Online Stochastic Tensor Decomposition for BS in Multispectral Video Sequences. *IEEE ICCV*, 2015.

[50] C. Stauffer and W. E. L. Grimson, Learning Patterns of Activity Using Real-Time Tracking. *IEEE TPAMI*, 22(8):747-757, 2000.

[51] B. Stenger, V. Ramesh, N. Paragios, F. Coetzee, and J. Bouhman. Topology free hidden markov models: Application to background modeling. *IEEE ICCV*, 294 - 301 , 2001.

[52] P. Turaga, R. Chellappa, V. S. Subrahmanian and O. Udrea. Machine Recognition of Human Activities: A Survey. *IEEE TCSVT*, 18(11):1473-1488, 2008.

[53] S. Varadarajan, P. Miller and H. Zhou. Spatial mixture of Gaussians for dynamic background modelling. *AVSS*, 63-68, 2013.

[54] P. Viola and M. J. Jones. Robust Real-Time Face Detection. *Int'l J. of Computer Vision*, 57(2):137-154, 2004.

[55] B. Wang and P. Dudek. A Fast Self-tuning BS Algorithm. *IEEE Workshop on CD*, 395-398, 2014.

[56] R. Wang, F. Bunyak, G. Seetharaman and K. Palaniappan. Static and Moving Object Detection Using Flux Tensor with Split Gaussian Models. *IEEE Workshop on CD*, 420-424, 2014.

[57] W. Wang, J. Shen, L. Shao. Consistent video saliency using local gradient flow optimization and global refinement. *IEEE TIP*, 24(11):4185-4196, 2015.

[58] W. Wang, J. Shen, X. Li and F. Porikli. Robust video object co-segmentation. *IEEE TIP*, 24(10):3137-3148, 2015.

[59] W. Wang, J. Shen and F. Porikli. Saliency-aware geodesic video object segmentation. *IEEE CVPR*, 3395-3402, 2015.

[60] Y. Wang, P.-M. Jodoin, F. Porikli, J. Konrad, Y. Benezeth and P. Ishwar. CDnet 2014: An Expanded Change Detection Benchmark Dataset. *IEEE Workshop on CD*, 387-394, 2014.

[61] K. Xue. Y. Liu. G. Ogunmakin. J. Chen and J. Zhang. Panoramic Gaussian Mixture Model and large-scale range background substraction method for PTZ camera-based surveillance systems. *MVA*. 24(3):477-492, 2013.

[62] D. Zamalieva and A. Yilmaz. Background Subtraction for the Moving Camera: A Geometric Approach. *CVIU*, 127:73-85, 2014.

[63] D. Zhang, O. Javed and M. Shah. Video Object Segmentation through Spatially Accurate and Temporally Dense Extraction of Primary Object Regions. *IEEE CVPR*, 628-635, 2013.

[64] Q. Zhao, Z. L. Zhou, A. Cichocki and S. Amari. Robust bayesian tensor factorization for incomplete multiway data. *CoRR*, abs/1410.2386, 2014.

[65] Z. Zivkovic. Improved Adaptive Gaussian Mixture Model for BS. *IEEE ICPR*, 28-31, 2004.

Aïssa Boulmerka Aïssa Boulmerka received the B Eng and MSc degrees in computer science from the École nationale Supérieure en Informatique (ESI), Algeria in 2004 and 2009, respectively. In 2016, he received the PhD degree in computer science from ESI/Algeria. Since 2009, he has been a lecturer of computer science at the University Center of Mila, Algeria. His primary research interests include statistical models applied to image/video segmentation and recognition.

Mohand Saïd Allili Mohand Saïd Allili received his MSc and PhD degrees in computer science from the University of Sherbrooke, Quebec, Canada, in 2004 and 2008, respectively. Since June 2008, he has been a Professor of computer science in the Department of Computer Science and Engineering at the University of Quebec in Outaouais (UQO) in Canada. He received the best PhD thesis award in engineering and natural sciences from the University of Sherbrooke for the year 2008. Two of his papers have received best student paper and best vision paper awards at the conferences CRV 2007 and CRV 2010, respectively. His main research interests include computer vision and graphics, image processing, pattern recognition and machine learning.