

A model change detection approach to dynamic scene modeling

Seon Joo Kim[†] Gianfranco Doretto[‡] Jens Rittscher[‡] Peter Tu[‡] Nils Krahnstoever[‡] Marc Pollefeys^{†§}

[†]Department of Computer Science, University of North Carolina, Chapel Hill, USA

Email: {sjkim,marc}@cs.unc.edu

[‡]Department of Computer Science, ETH, Zürich, Switzerland

Email: marc.pollefeys@inf.ethz.ch

[§]Visualization and Computer Vision Lab, GE Global Research, Niskayuna, USA

Email: {doretto,rittische,tu,krahnsto}@research.ge.com

Abstract—In this work we propose a dynamic scene model to provide information about the presence of salient motion in the scene, and that could be used for focusing the attention of a pan/tilt/zoom camera, or for background modeling purposes. Rather than proposing a set of saliency detectors, we define what we mean by salient motion, and propose a precise model for it. Detecting salient motion becomes equivalent to detecting a model change. We derive optimal online procedures to solve this problem, which enable a very fast implementation. Promising results show that our model can effectively detect salient motion even in severely cluttered scenes, and while a camera is panning and tilting.

Keywords—dynamic scene modeling, PTZ camera, focus-of-attention, background modeling, sequential generalized likelihood ratio, model change detection, linear dynamical systems, dynamic textures.

I. INTRODUCTION

In current video surveillance applications pan/tilt/zoom (PTZ) cameras are used for the manual monitoring of large areas such as parking lots, runways of airports, or perimeters of large sites. By using an interactive controller such cameras are utilized to capture high-resolution images of people, vehicles, and other objects of interest. When PTZ cameras are not used interactively, they can be programmed to move along predefined trajectories. Best of all would be if they could automatically detect, follow, and record video snippets of events that the system considers to be *salient activities*. To enable this, we should design algorithms that allow PTZ cameras for the localization of these activities in space and time.

In [1] a static master camera that monitors a wide area is used to detect people in the field of view. This information is then used to automatically aim a PTZ camera towards the identified target. Although this is a viable approach, it requires two cameras, which is not a cost-effective solution for many practical applications. In this paper we develop a dynamic background model that could be used to focus the attention of a PTZ camera without the need for a master camera.

In [2], a model is proposed for the automatic focus-of-attention of a PTZ camera. The camera moves to a series of random PTZ values where it remains static enough

time to build a motion history image of that location. A navigation algorithm decides how to fuse this information with an activity map of the entire “spherical” field of view of the PTZ camera, and decides how to move the camera to a location with salient activity. In general, one would like to have the PTZ camera always in motion while an activity map is being updated. To this end, one needs to build a background model for a moving camera, which is continuously updated, and which provides information to update the activity map.

This is challenging because the background needs to be updated while factoring out the motion noise due to the lack of a perfect registration of the images with respect to the model. The noise is generated by extrinsic calibration errors, and camera vibrations. Only limited work has been done in this area. [3] proposed to extend the Gaussian mixture model of [4] to deal with PTZ camera motion by including mixtures of neighbors to deal with inaccuracies in the registration. More recently, [5] proposed to use feature matching to compute the registration of a new image with a reference model image (see [5] for a few more works in this area). The biggest drawback of these approaches is the fact that they have been tested only with scenes where the background was mostly static or slowly varying. A more general scenario is given by a highly dynamic background, which significantly complicates the problem by adding a great deal of motion clutter. We refer all the sources of motion unrelated to the salient activities as *nuisance motion*.

In this paper we address the problem of detecting the *salient motion* related to a salient activity embedded in a dynamic scene imaged by a panning and titling camera. We do so by proposing a generative model for nuisance motion, and salient motion is detected as a deviation from this model. We provide a precise definition for nuisance and salient motion, which is based on the concept of *stationarity* of stochastic processes (Section II). We formulate the salient motion detection problem as a *model change detection problem*, which we propose to solve optimally, online, by applying the generalized likelihood ratio (GLR) test [6] (Section V). To improve efficiency, we decompose the nuisance motion model into a hierarchy of three models

(Section III). The first one aims at dealing with portions of the scene that are perfectly static. The second one still deals with portions of static scenes, but registration errors are taken into account. The third one deals with highly dynamic portions of the scene. Section IV presents an efficient and recursive parameter estimation procedure for each model that is also adaptive, because it gives more importance to recent measurements according to a forgetting factor, and allows the detection system for adjusting to slow variations of the visual process, favoring the reduction of false alarms. Section VI illustrates results that show the promise of this approach for dealing with highly dynamic scenes.

Related work: There is a large body of work on modeling backgrounds for the purpose of detecting foreground events. Most of it is limited to the case of static or slowly varying background, and does not deal very well with strong dynamic background motion. Relevant works include [4], [7], [8], and the reader can refer to [9] for a recent comparison.

Dynamic backgrounds are considered in [10]. The framework has a number of limitations because not all the parameters are updated online, and the detection approach is somewhat *ad-hoc*. [11] also deals with dynamic backgrounds, only that the images are modeled as a whole, and this implies the camera to be static.

Besides background modeling, other methods for detecting salient activities include [12], where they combine feature detectors for color, intensity, orientation, flicker, and motion into a measure of surprise. [13] allows detecting surprising events in video as events that cannot be correlated with a database of normal events. [14] proposed a saliency criterion that detects motions that tend to exhibit a consistent direction over time by analyzing the optical flow.

II. SALIENT AND NUISANCE MOTION

Let us assume that $\{I_t(\mathbf{x})\}_{1 \leq t \leq T}$ is a sequence of images acquired by a PTZ camera undergoing a rotational motion $g(t) \in SO(3)$. The image at time t is such that the image coordinate $\mathbf{x} \in \Omega \subset \mathbb{R}^2$, and $I_t : \Omega \rightarrow \mathbb{R}_+$. We also assume that the PTZ camera center is fixed. With $\Lambda_t : \mathbb{S}^2 \rightarrow \mathbb{R}_+$ we indicate the spherical image that is seen from the center of the PTZ camera at time t . With $\Lambda_t(\mathbf{x})$ we indicate the spherical image parameterized according to the cartesian domain where the image I_t is defined. With these assumptions we have that

$$I_t(\mathbf{x}) = \chi_{g(t)}(\mathbf{x})\Lambda_t(\mathbf{x}), \quad (1)$$

where $\chi_{g(t)}(\mathbf{x}) = 1$ if $\mathbf{x} \in \Omega$, and 0 otherwise, is an indicator function that selects the appropriate portion of the spherical image that corresponds to I_t .

We view the sequence of images $\{I_t\}_{1 \leq t \leq T}$ as a truncated realization, in space and time, of a stochastic process defined on the unit sphere, given by Λ_t . We say that the sequence $\{I_t\}$ contains only *nuisance motion* if there exists a time

*stationary stochastic process*¹ $\tilde{\Lambda}_t$, possibly different from Λ_t , that admits the sequence $\{I_t\}$ as a realization.

If we restrict $\tilde{\Lambda}_t$ to be a second-order stationary process, then it can be modeled as a dynamic texture [15], only defined on a unit sphere, i.e.

$$\begin{cases} x_{t+1} = Ax_t + v_t, & v_t \stackrel{IID}{\sim} \mathcal{N}(0, Q); \\ \tilde{\Lambda}_t = \tilde{\Lambda}_0 + \tilde{C}x_t + \tilde{w}_t, & \tilde{w}_t \stackrel{IID}{\sim} \mathcal{N}(0, \tilde{R}), \end{cases} \quad (2)$$

where $\tilde{\Lambda}_0$ is the mean spherical image, $A \in \mathbb{R}^{n \times n}$ and \tilde{C} are matrices that encode the dynamics and the appearance of $\tilde{\Lambda}_t$, whereas v_t , and \tilde{w}_t are a driving noise process, and a measurement noise process, with corresponding variances given by Q and \tilde{R} . From Equation (1) we obtain the model for a sequence $\{I_t\}$ containing only nuisance motion, which is given by

$$\begin{cases} x_{t+1} = Ax_t + v_t, & v_t \stackrel{IID}{\sim} \mathcal{N}(0, Q); \\ I_t = I_{0,t} + \check{C}_t x_t + w_t, & w_t \stackrel{IID}{\sim} \mathcal{N}(0, \check{R}), \end{cases} \quad (3)$$

where $I_{0,t}(\mathbf{x}) \doteq \chi_{g(t)}(\mathbf{x})\tilde{\Lambda}_0(\mathbf{x})$ is the mean image, $\check{C}_t(\mathbf{x}) \doteq \chi_{g(t)}(\mathbf{x})\tilde{C}(\mathbf{x})$ is an appropriate selection of the rows of \tilde{C} , and describes the appearance of I_t , and $w_t(\mathbf{x}) \doteq \chi_{g(t)}(\mathbf{x})\tilde{w}_t(\mathbf{x})$ represents the corresponding measurement noise. Its covariance \check{R} does not vary over time because the measurement noise is equally distributed over the unit sphere.

Note that, if the sequence $\{I_t\}$ contains *salient motion*, then $\tilde{\Lambda}_t$ is a non-stationary process, and all the parameters in Equation (2), $\Lambda_{0,t}$, A_t , \check{C}_t , Q_t , \check{R}_t , are time-variant, which means that all the parameters in Equation (3), $I_{0,t}$, A_t , \check{C}_t , Q_t , \check{R}_t , are *time-variant* too. Moreover, if $\{I_t\}$ contains nuisance motion, and the PTZ camera does not move (i.e. $g(t)$ is constant), then all the parameters in Equation (3), I_0 , A , \check{C} , Q , \check{R} , are *time-invariant*.

What we just discussed provides us with the basic principle to detect salient motion. Since the motion of the PTZ camera, $g(t)$, is known, we can register the images in the spherical domain. Here time-invariance of the model means nuisance motion, whereas time-variance means salient motion. In practice, we project the portion of interest of the spherical domain onto a plane. This means that rather than computing de-rotations, we compute warpings according to homographies $\{H_t\}$, corresponding to $g(t)$.

If we indicate with $\{I_t(H_t\mathbf{x})\}$ the warped sequence of images, supposed to be defined on a common domain $D = \bigcup_{t=1}^T H_t^{-1}\Omega \subset \mathbb{R}^2$, in order to localize the detection of salient motion, we divide D into a number of N , possibly overlapping, rectangular domains $\{D_i\}$, such that $D = \bigcup_{i=1}^N D_i$, and indicate with $\{y_t^{(i)}\}$, the sequence of warped image measurements defined over D_i .

¹A stochastic process is stationary (of order k) if the joint statistics (up to order k) are time-invariant. For instance, a process $\{\Lambda_t\}$ is second-order stationary if its mean $\bar{\Lambda} \doteq E[\Lambda_t]$ is constant and its covariance $E[(\Lambda_{t_1} - \bar{\Lambda})(\Lambda_{t_2} - \bar{\Lambda})]$ only depends upon $t_2 - t_1$.

III. HIERARCHY OF MODELS

In this section we focus the attention on modeling $\{y_t\}$, defined on D_i , where we have dropped the superscript to simplify the notation. We assume the presence of nuisance motion. Note that every D_i will correspond to a rectified spherical region identifiable by the indicator function χ_{D_i} , such that $\chi_{D_i}(\mathbf{x}) = 1$ if $\mathbf{x} \in D_i$, and 0 otherwise, and with which $\tilde{\Lambda}_t$ can be related to y_t according to $y_t(\mathbf{x}) = \chi_{D_i}(\mathbf{x})\tilde{\Lambda}_t(\mathbf{x})$. Therefore, the model for y_t follows immediately from Equation (2). However, since there are registration errors due to imprecisions with which the motion $g(t)$ is known, we augment the model by assuming that the measurement y_t needs to be shifted by a small planar translation \mathbf{T}_t . This leads to the following model:

$$\begin{cases} x_{t+1} = Ax_t + v_t, & v_t \stackrel{IID}{\sim} \mathcal{N}(0, Q) \\ y_t(\mathbf{x} + \mathbf{T}_t) = y_0(\mathbf{x}) + C(\mathbf{x})x_t + w_t(\mathbf{x}), \end{cases} \quad (4)$$

where $y_0(\mathbf{x}) \doteq \chi_{D_i}(\mathbf{x})\tilde{\Lambda}_0(\mathbf{x})$, $C(\mathbf{x}) \doteq \chi_{D_i}(\mathbf{x})\tilde{C}(\mathbf{x})$, and $w_t(\mathbf{x}) \doteq \chi_{D_i}(\mathbf{x})\tilde{w}_t(\mathbf{x}) \stackrel{IID}{\sim} \mathcal{N}(0, R)$, which is a special case of the model of [16].

Model (4) represents the complete generative model of nuisance motion. However, we observe that if the region D_i would correspond to a static portion of the scene, detection could be performed with a simplified model, which would quickly confirm that no salient motion was observed. Should a detection occur, another procedure would verify whether the motion could be modeled with the next model in the hierarchy, and so on. This would lead to an improved computational performance, and justifies why we decompose (4) into a hierarchy of three models.

Model M_1 : Static background could be represented by the following simplification of model (4), given by

$$\begin{cases} x_{t+1} = 0 \\ y_t = y_{01} + w_t, & w_t \stackrel{IID}{\sim} \mathcal{N}(0, R_1), \end{cases} \quad (5)$$

where y_{01} and R_1 are mean and covariance of the image measurements y_t . The model detects image motion that cannot be modeled as image measurement noise.

Model M_2 : This model adds to M_1 the possibility to compensate for registration errors, due to imprecise calibration and sensor vibrations, which could cause false alarms,

$$\begin{cases} x_{t+1} = 0 \\ y_t(\mathbf{x} + \mathbf{T}_t) = y_0(\mathbf{x}) + w_t(\mathbf{x}), & w_t \stackrel{IID}{\sim} \mathcal{N}(0, R_2). \end{cases} \quad (6)$$

Note that $y_0 \neq y_{01}$, and $R_2 \neq R_1$, because they are computed after the registration error compensation.

Model M_3 : This model, given by Equation (4), adds to M_2 the possibility to capture the statistics, up to the second order, of y_t . Anything not captured by (4) should be detected as salient motion.

IV. ADAPTIVE ONLINE PARAMETER ESTIMATION

We derive a procedure to estimate model parameters adaptively and online to enable realtime performance.

Model M_1 : The estimation of the mean y_{01} is updated by an exponential moving average, which means that

$$\hat{y}_{01}(t) = \lambda \hat{y}_{01}(t-1) + (1-\lambda)y_t, \quad (7)$$

where λ is a *forgetting factor* that exponentially weights the measurements by giving more importance to the recent ones, enabling the adaptability of the model to the current situation.

The covariance R_1 of model (5), is computed with the exponentially weighted version of the sample covariance estimator, which gives the following unbiased update equation

$$\hat{R}_1(t) = \lambda \hat{R}_1(t-1) + \frac{(1-\lambda^2)}{2\lambda} (y_t - y_{01})(y_t - y_{01})^T. \quad (8)$$

Model M_2 : For model (6) we estimate each parameter separately. \mathbf{T}_t is estimated by solving the following template matching problem:

$$\hat{\mathbf{T}}_t = \arg \min_{\mathbf{T}} \int_{D_i} |y_t(\mathbf{x} + \mathbf{T}) - y_0(\mathbf{x})|^2 dx + \nu \|\mathbf{T}\|^2, \quad (9)$$

where the second term of the cost function is a prior that limits the translation length, as we are interested in correcting small registration errors. We solve (9) efficiently with the inverse compositional algorithm [17]. After estimating \mathbf{T}_t , we compute $y_t(\mathbf{x} + \mathbf{T}_t)$ and update y_0 and R_2 with equations analogous to Equation (7) and Equation (8), respectively.

Model M_3 : After computing \mathbf{T}_t , $y_t(\mathbf{x} + \mathbf{T}_t)$, and y_0 as explained before, one can update C with the quantity $\rho_t(\mathbf{x}) \doteq y_t(\mathbf{x} + \mathbf{T}_t) - y_0(\mathbf{x})$. This can be done in a recursive fashion by using a specialization of the subspace update algorithm described in [18]. This procedure is very fast, numerically stable, and allows for the insertion of the forgetting factor λ , which has the same meaning of the one in Equation (7). Algorithm 1 describes the procedure.

After the update of C the state x_t can be computed as $C^T \rho_t(\mathbf{x} + \mathbf{T}_t)$. Given x_t , it is possible to update the state matrix A . This is done with a variant of the recursive least squares algorithm [19], which leads to the following update equations

$$L(t) = \frac{x_{t-1}^T \Sigma(t-1)}{\lambda + x_{t-1}^T \Sigma(t-1) x_{t-1}}, \quad (10)$$

$$\hat{A}(t) = \hat{A}(t-1) + (x_t - \hat{A}(t-1)x_{t-1})L(t), \quad (11)$$

$$\Sigma(t) = \frac{1}{\lambda} \Sigma(t-1)(I - x_{t-1}L(t)), \quad (12)$$

where $L(t)$ is the Kalman gain, $\Sigma(t)$ is a covariance matrix, and I here is the identity matrix. Finally, from $x_t - Ax_{t-1}$ one can update Q , and from $y_t(\mathbf{x} + \mathbf{T}_t) - y_0(\mathbf{x}) - C(\mathbf{x})x_t$ one can update R with an exponentially weighted sample covariance estimation, like it is done in Equation (8).

Algorithm 1 Recursive update of C .

Input: $C(t-1)$, $S(t-1)$, ρ_t , λ .**Output:** $\hat{C}(t)$, $\hat{S}(t)$.

- 1: $r_y \leftarrow C(t-1)^T \rho_t$
 - 2: $p_y \leftarrow \rho_t - C(t-1)r_y$
 - 3: $C_e \leftarrow \begin{bmatrix} C(t-1) & \frac{p_y}{\|p_y\|} \end{bmatrix}$
 - 4: $S_e \leftarrow \begin{bmatrix} \lambda S(t-1) & r_y \\ 0 & p_y \end{bmatrix}$
 - 5: Compute $\tilde{U}\tilde{S}\tilde{V}'$, the SVD of S_e .
 - 6: **return** $\hat{C}(t) = C_e\tilde{U}_{:,1:n}$, and $\hat{S}(t) = \tilde{S}_{1:n,1:n}$
-

V. DETECTION OF SALIENT MOTION

So far we have introduced a background model, which represents image measurements in every domain D_i by three parametric models $M_1(\theta_1)$, with parameter $\theta_1 \doteq \{y_{01}, R_1\}$, $M_2(\theta_2)$, with parameter $\theta_2 \doteq \{y_0, R_2\}$, and $M_3(\theta_3)$, with parameter $\theta_3 \doteq \{y_0, A, C, Q, R\}$.

Each of the three models will operate individually in the same way. In particular, in presence of nuisance motion a model $M(\theta)$ is time-invariant, which means that the parameter does not change over time, $\theta \doteq \theta_0$. In presence of salient motion the model is time-variant, which means that the parameter is changing, $\theta \doteq \theta(t)$. If at time t_0 we observe the presence of salient motion, this means that we switched from a model $M(\theta_0)$, $t < t_0$, to a model $M(\theta(t))$, $t \geq t_0$. Therefore, given the set of measurements y_1, \dots, y_t , we need to be able to discriminate between the hypotheses

$$\mathbf{H}_0 : \{y_t \text{ comes from } M(\theta), \text{ with } \theta = \theta_0\}, \quad (13)$$

$$\mathbf{H}_1 : \{y_t \text{ comes from } M(\theta), \text{ with } \theta \neq \theta_0\}. \quad (14)$$

Since in \mathbf{H}_1 the parameter θ is unknown, what we need to solve is a so called *composite hypothesis testing* problem. What we have just described is known as the *model change detection problem* [6].

M_1 , M_2 , M_3 will work together in this way. M_1 models the nuisance motion given by image measurement noise of a static background, and will be the first to perform a detection. If \mathbf{H}_0 holds, then all the parameters are updated with the new measurements. If \mathbf{H}_1 holds, then M_2 performs a detection. For M_2 the nuisance motion includes also registration errors. If \mathbf{H}_0 holds, then the parameters are updated. If \mathbf{H}_1 holds, then M_3 performs a detection. For M_3 the nuisance motion includes also any dynamic visual process that is second-order stationary. If \mathbf{H}_0 holds, then the parameters are updated. If \mathbf{H}_1 holds, then the observed motion is not due to image measurement noise, it is not due to registration errors, and it does not have a stationary dynamics. Therefore, it is salient motion.

Detecting changes of M_1 : From (5) it follows that $\{y_t\}_{t \geq 1}$ is such that $y_t \stackrel{IID}{\sim} \mathcal{N}(y_{01}, R_1)$. Being able to detect a change of θ_1 in principle means being able to detect a

change of y_{01} , or a change of R_1 , or a change of both. However, M_1 represents a static portion of the background, and R_1 should account for image measurement noise, and should not vary. Therefore, we focus on detecting changes of the mean y_{01} .

The composite hypothesis testing problem of detecting changes of the mean y_{01} can be solved optimally, online, using the *sequential generalized likelihood ratio* (GLR) test [6]. The decision rule chooses \mathbf{H}_0 , if $h_t < \gamma$, and chooses \mathbf{H}_1 , if $h_t \geq \gamma$. γ is a convenient threshold, and h_t is a test statistic (which one can prove to be sufficient), given by $h_t = \max_{1 \leq j \leq t} S_j^t$, where

$$S_j^t = \ln \frac{\sup_{\{\theta | \theta \neq \theta_0\}} \prod_{i=j}^t p_\theta(y_i)}{\prod_{i=j}^t p_{\theta_0}(y_i)}, \quad (15)$$

and S_j^t is the logarithm of a GLR, as one may notice that it is the maximization of the likelihood ratio with respect to all the possible instances of θ after the change. The first detection is performed at time $t_a = \min\{t \geq 1 | h_t \geq \gamma\}$, which is called time of alarm.

Equation (15) reduces to the following expression: $S_j^t = \frac{t-j+1}{2} (\chi_j^t)^2$, where $\chi_j^t = [(\bar{y}_j^t - y_{01})' R_1^{-1} (\bar{y}_j^t - y_{01})]^{1/2}$, and $\bar{y}_j^t = \frac{1}{t-j+1} \sum_{i=j}^t y_t$. These equations can be written in a recursive fashion. In particular, since the parameters are learned adaptively, it is easy to derive their recursive and adaptive counterparts, which include the usual forgetting factor λ .

Note that the sequential GLR does not take per-frame decisions independently, but time-integration is an important aspect of this approach. More precisely, the algorithm compares, through S_j^t , the parameter θ_0 , estimated in the growing time window $[1, t]$, to values of $\theta \neq \theta_0$, estimated in all possible time windows $\{[j, t] | 1 \leq j \leq t\}$. This is the optimal result of a formal problem statement. The actual implementation computes the test statistics recursively, adaptively, and the parameter comparison is limited to the J most recent time windows $\{[j, t] | t - J + 1 \leq j \leq t\}$, where J is a tuning parameter.

Detecting changes of M_2 : From (6) it follows that $\{y_t\}_{t \geq 1}$ is such that $y_t(\mathbf{x} + \mathbf{T}_t) \stackrel{IID}{\sim} \mathcal{N}(y_0, R_2)$. M_2 represents a static portion of the background, after the elimination of registration errors, and R_2 should account for image measurement noise, and residual registration errors, and should not vary. Therefore, we focus on detecting changes of the mean y_0 . This can be done by using exactly the same procedure applied for detecting changes of M_1 , only with the new parameters y_0 , and R_2 .

Detecting changes of M_3 : From (4) it follows that the process $\{y_t(\mathbf{x} + \mathbf{T}_t)\}_{t \geq 1}$ is modeled as the output of a linear dynamical system with parameters θ_3 . A change of y_0 would be produced by *additive changes* in the state transition equation and/or in the observation equation of (4). Changes of C , Q , R affect mostly the covariance of the



Figure 1. **Additive Changes.** Left: Simulation of model (4) after parameter estimation. Right: Simulation of model (4) after blending y_0 with the image of a boat.

process (also the spectrum), and they are called *changes in variance*. Changes of A directly affect the spectrum of the process, and they are called *spectral changes*, [6].

Detecting a change of M_3 means being able to detect additive, variance, and spectral changes. [20] studies a wide class of spectral changes and changes in variance for the purpose of synthesizing dynamic textures with new appearance and dynamics. If we consider a sequence with ocean waves, these changes may vary the speed of the wave motion, or may make the waves appear bigger or smaller. These are not the types of salient motion we would like to detect. On the other hand, Figure 1 shows a synthetic experiment where we fit model (4), which is a dynamic texture, to the ocean wave sequence, and produce an additive change to the estimated model by substituting the mean image y_0 with a new image that replaces some pixels with the image of a boat. The simulation of this new model clearly shows the boat, which represents a change we would definitely want to detect, and this is why we concentrate on detecting additive changes.

Additive changes of M_3 are modeled as follows

$$\begin{cases} x_{t+1} = Ax_t + v_t + \Upsilon_{x_t, t_0}, \\ \tilde{y}_t = y_0 + Cx_t + w_t + \Upsilon_{y_t, t_0}, \end{cases} \quad (16)$$

where $\tilde{y}_t \doteq y_t(\mathbf{x} + \mathbf{T}_t)$. Υ_{x_t, t_0} and Υ_{y_t, t_0} are the dynamic profiles of the changes. They are unknown a priori, and they are zero when $t \leq t_0$, where t_0 is the time of change.

One can prove that the innovation process $\{\varepsilon_t\}$, computable by applying a Kalman filter to the observations $\{\tilde{y}_t\}$, is distributed according to

$$\varepsilon_t \stackrel{IID}{\sim} \begin{cases} \mathcal{N}(0, \Sigma_{\varepsilon_t}), & \text{for } t < t_0; \\ \mathcal{N}(\Upsilon_{t, t_0}, \Sigma_{\varepsilon_t}), & \text{otherwise}; \end{cases} \quad (17)$$

where Υ_{t, t_0} , which is unknown, is the dynamic profile of the innovation, corresponding to Υ_{x_t, t_0} and Υ_{y_t, t_0} [6]. In this way, all the information of an additive change is carried by the innovation, which is an IID process distributed according to model (17). Therefore, deciding for \mathbf{H}_0 , or \mathbf{H}_1 can be done optimally by applying the sequential GLR to the process $\{\varepsilon_t\}$.

VI. EXPERIMENTS

We tested our C++ implementation of the detection algorithm on two video sequences, both acquired with a

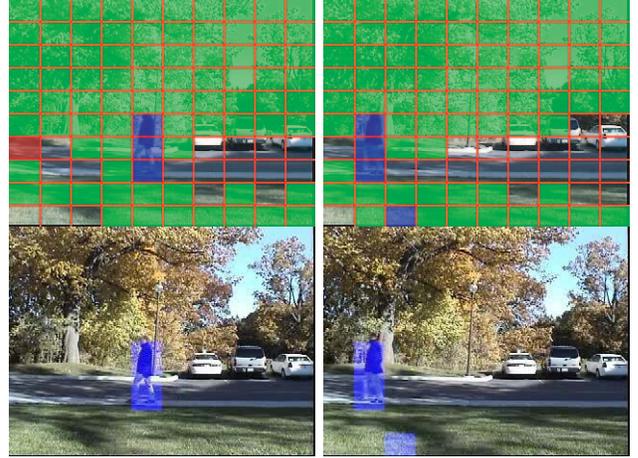


Figure 2. **Outdoor Sequence.** Top row: Color-coded output of the detection procedures associated to the models M_1 , M_2 , and M_3 . A transparent block means measurement noise. A red block means registration errors. A green block means stationary dynamic motion. A blue block means non-stationary motion, i.e. salient motion. Bottom row: salient motion detection results.

SONY PTZ camera. The outdoor sequence was acquired while keeping the PTZ camera static. The sequence shows a remarkable amount of motion clutter due to the strong wind that is moving the trees, which in turn are continuously changing the shadow patterns of the scene. The salient motion that the algorithm should detect is given by a person walking by. The indoor sequence was acquired while the PTZ camera was panning. The motion clutter of the scene is produced artificially by a fan that is moving the leaves of a couple of plants, and by a water fountain. They both produce stationary motion. The salient motion that should be detected is given by a fluffy dog walking by. The motion $g(t)$ of the PTZ camera is known because we stored the PTZ values along with the video data. Also, the camera was calibrated beforehand, and a look-up table is used to associate different PTZ values with a particular homography.

Figure 2 shows the detection results for the outdoor sequence. The top two images show that the image domain was divided into small rectangular domains, or blocks, $\{D_i\}$, which in this case are non-overlapping. Their dimension is of 20×15 pixels. The state dimension n was set to 8. All the forgetting factors λ were set to 0.99. The models had identical settings for each of the blocks. After the first 50 frames the background models were learnt, and ready to perform stable detections. In the top row, each domain is color-coded to indicate the last model in the hierarchy of M_1 , M_2 , and M_3 , that established a detection. More precisely, a *transparent* block indicates no detections. This means that M_1 established that the current image data at that block was only affected by measurement noise. A *red* block indicates that M_1 performed a detection, and M_2 established that the current image data was also affected by registration errors.



Figure 3. **Indoor Sequence.** Top row: Color-coded output of the detection procedures associated to the models M_1 , M_2 , and M_3 . See Figure 2 for color descriptions. Bottom row: salient motion detection results. The images on the left contain some false alarms, which could easily be removed by a temporal filter.

A *green* block indicates that M_2 performed a detection, and M_3 established that the current image data was dynamic, but stationary. A *blue* block indicates that M_3 detected that the current image data was dynamic and non-stationary, and therefore it is salient motion. The bottom row of Figure 2 shows only the salient motion detections. The model shows to be effective in filtering out all the stationary motion of the scene, which current background subtraction methods still struggle with. The salient motion, due to the walking pedestrian, is detected reliably.

Figure 3 shows the results on the indoor sequence. This sequence contains areas that are perfectly static, and areas with stationary dynamic motion. Since the camera is panning, compared to the outdoor sequence, we see more red blocks, which means that registration errors needed to be corrected. Even in this case, the salient motion is revealed. The results on both sequences show a small number of false alarms, which could easily be removed by a temporal filter.

VII. CONCLUSIONS

This work presents an approach for detecting salient motion in severely cluttered scenes as viewed from a panning and tilting camera, where no current methods have shown to be effective. The framework provides definitions, precise models, and techniques that we demonstrated can filter out nuisance motion. The recursive procedures for adaptive, online model estimation and detection, enable a realtime implementation of the approach. Future work will be devoted to the inclusion of zoom variations, and to a thorough comparison of the method against the state-of-the-art background models for static cameras.

REFERENCES

[1] X. Zhou, R. Collins, T. Kanade, and P. Metes, "A master-slave system to acquire biometric imagery of humans at distance," in *ACM IWVS*, 2003, pp. 113–120.

[2] J. W. Davis, A. M. Morison, and D. D. Woods, "An adaptive focus-of-attention model for video surveillance and monitoring," *MVA*, vol. 18, no. 1, pp. 41–64, 2007.

[3] E. Hayman and J. O. Eklundh, "Statistical background subtraction for a mobile observer," in *ICCV*, 2003, pp. 67–74.

[4] C. Stauffer and W. E. L. Gimson, "Learning patterns of activity using real-time tracking," *IEEE TPAMI*, vol. 22, no. 8, pp. 747–757, 2000.

[5] Y. Zhao, M. Casares, and S. Velipasalar, "Continuous background update and object detection with non-static cameras," in *AVSS*, Sep. 1–3, 2008, pp. 309–316.

[6] M. Basseville and I. Nikiforov, *Detection of abrupt changes: theory and application*, T. Kailath, Ed. Prentice Hall, 1993.

[7] A. M. Elgammal, D. Harwood, and L. S. Davis, "Non-parametric model for background subtraction," in *ECCV*, 2000, pp. 751–767.

[8] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati, "Detecting moving objects, ghosts, and shadows in video streams," *IEEE TPAMI*, vol. 25, no. 10, pp. 1337–1342, Oct. 2003.

[9] D. H. Parks and S. S. Fels, "Evaluation of background subtraction algorithms with post-processing," in *AVSS*, Sep. 2008, pp. 192–199.

[10] A. Mittal, A. Monnet, and N. Paragios, "Scene modeling and change detection in dynamic scenes: A subspace approach," *CVIU*, vol. 113, no. 1, pp. 63–79, 2009.

[11] J. Zhong and S. Sclaroff, "Segmenting foreground objects from a dynamic textured background via a robust Kalman filter," in *ICCV*, 2003, pp. 44–50.

[12] L. Itti and P. Baldi, "A principled approach to detecting surprising events in video," in *CVPR*, vol. 1, 2005, pp. 631–637.

[13] O. Boiman and M. Irani, "Detecting irregularities in images and in video," in *ICCV*, vol. 1, 2005, pp. 462–469.

[14] L. Wixson, "Detecting salient motion by accumulating directionally-consistent flow," *IEEE TPAMI*, vol. 22, no. 8, pp. 774–780, 2000.

[15] G. Doretto, A. Chiuso, Y. N. Wu, and S. Soatto, "Dynamic textures," *IJCV*, vol. 51, no. 2, pp. 91–109, 2003.

[16] G. Doretto and S. Soatto, "Dynamic shape and appearance models," *IEEE TPAMI*, vol. 28, no. 12, pp. 2006–2019, 2006.

[17] S. Baker and I. Matthews, "Lucas-Kanade 20 years on: a unifying framework," *IJCV*, vol. 56, no. 3, pp. 221–255, 2004.

[18] A. Levy and M. Lindenbaum, "Sequential Karhunen-Loeve basis extraction and its application to images," *IEEE Trans. on Image Processing*, vol. 2, pp. 456–460, 1998.

[19] L. Ljung, *System identification: theory for the user*, 2nd ed. Prentice-Hall, Inc., 1999.

[20] G. Doretto and S. Soatto, "Editable dynamic textures," in *CVPR*, vol. 2, 2003, pp. 137–142.