Scene Modeling and Change Detection in Dynamic Scenes: A Subspace Approach

Anurag Mittal

Indian Institute of Technology Madras, Chennai, INDIA

Antoine Monnet

Siemens Corporate Research, Princeton, NJ 08540

Nikos Paragios

Ecole Centrale de Paris, Grande Voie des Vignes, 92 295 Chatenay-Malabry, FRANCE

Abstract

Background modeling and subtraction are core components in video processing. To this end, one aims to recover and continuously update a representation of the scene that is compared with the current input to perform subtraction. Most of the existing methods treat each pixel independently and attempt to model the background perturbation through statistical modeling such as a mixture of Gaussians. While such methods have satisfactory performance in many scenarios, they do not model the relationships and correlation amongst nearby pixels. Such correlation between pixels exists both in space and across time especially when the scene consists of image structures moving across space. Waving trees, beach, escalators and natural scenes with rain or snow are examples of such scenes. In this paper, we propose a method for differentiating between image structures and motion that are persistent and repeated from those that are "new". Towards capturing the appearance characteristics of such scenes, we propose the use of an appropriate subspace created from image structures. Furthermore, the dynamical characteristics are captured by the use of a prediction mechanism in such subspace. Since the model must adapt to long-term changes in the background, an incremental method for fast online adaptation of the model parameters is proposed. Given such adaptive models, robust and meaningful measures for detection that consider both structural and motion changes are considered. Promising experimental results that include qualitative and quantitative comparisons with existing background modeling/subtraction techniques demonstrate the very promising performance of the proposed framework when dealing with complex backgrounds.

Key words: Scene Analysis, Background Subtraction, Time Series, Principal Component Analysis.

1 Introduction

The proliferation of cheap sensors and increased computational power has made the acquisition and processing of video information more feasible. Real-time video analysis tasks such as object detection and tracking can now be efficiently performed on standard PC's for a variety of applications such as: Industrial automation, transportation, automotive, security & safety, and communications. The use of stationary cameras is rather common in such applications.

Background subtraction is a core component in many of such applications where the objective is to separate the new objects from the repetitive parts of the scene. The information provided by such a module can be considered as a valuable low-level visual cue to perform high-level tasks of object analysis, like object detection, tracking, classification and event analysis ([45,33,20,24,6,2,53,7,42]).

The task of background modeling and subtraction consists of recovering and continuously updating a representation of the scene that is compared with the current input to perform detection. Methods for such modeling of the background may be classified into two categories: *predictive* and *non-predictive*. *Predictive* methods attempt to model the scene as a time series and develop a dynamical model to determine the current input based on past observations. The magnitude of the deviation between the predicted and actual observation can then be used as a measure of change. The second class of methods (which we call *non-predictive density-based* methods) neglect the particular order of the input observations and attempt to build a probabilistic representation (i.e. a *p.d.f.*) of the data at a given point in the scene [2] . A new observation can then be classified as background or foreground based on the probability that this observation belongs to the background.

1.1 Non-Predictive Density-Based Methods

Various methods have been proposed in the literature for developing a pixel-level statistical representation of the scene. The simplest model keeps a single background image that refers to the "empty" scene. Several authors discuss methods

Email addresses: amittal@cs.iitm.ernet.in (Anurag Mittal), nikos.paragios@ecp.fr (Nikos Paragios).

[1] Most of this work was done while the authors were with the Real-Time Vision and Modeling Department at Siemens Corporate Research, Princeton, NJ 08540.

[2] In a more general sense, non-predictive methods may be considered as a subset of predictive methods since they also retain some temporal information. However, since the temporal relationship between consecutive observations is mostly lost and such methods are unable to model short-term relationships in fast changing periodic and persistent signals such as a sinusoid, we have classified these methods in a separate category.

to perform illumination-invariant change detection using such background representation ([23,38,18]). Along this direction, a more advanced technique consists of a running average of the intensity, which would be a computationally efficient approach towards providing a rough description of the static scene in the absence of any moving objects. Variations of this method include - taking the median of the observed values, calculating spatially weighted values in order to reduce the effect of outliers and keeping the maximum, minimum and largest consecutive values [21]. Such methods do not explicitly model the background versus foreground, and are therefore not very effective for scenes where many moving objects are present and acquisition of a background image ”free” of foreground objects is not easy.

A static scene may be reasonably modeled with a single Normal distribution if the noise is modeled as being zero mean Normally distributed[59]. This can be used to classify a pixel as belonging to the foreground or background. Such decisions can also then be used to update the mean μ and the covariance matrix Σ of the background Gaussian incrementally.

Friedman et. al.[14] use a mixture of three Gaussians to model the visual properties of the background and foreground in traffic surveillance applications. Three hypotheses are considered - road, shadow and vehicles. The EM algorithm is a near-optimal method for simultaneously recovering both the parameters of the individual models and the classification of the data into different groups (since this is a chicken-and-egg problem). However, due to the computational complexity of the algorithm and real-time update requirements of the traffic surveillance problem, an incremental EM algorithm was considered to learn and update the model parameters efficiently. The background (i.e. road), however, is still modeled by a single Gaussian in this case.

Stauffer and Grimson ([20,52]) extended this idea by using multiple Gaussians to model the scene. Such an approach is capable of dealing with multiple hypotheses for the background and can be useful in scenes such as waving trees, beaches, escalators, rain and snow. In order to improve the efficiency of the method compared to the EM algorithm, they propose a simple exponential update scheme for the mixture model. Mittal and Huttenlocher [34] introduce a modification of this scheme by proposing the use of constant weighting along with exponential weighting and specify methods for selection of the scheme to be used at a particular time. The mixture-of-Gaussians method is quite popular and was to be the basis for a large number of related techniques ([19,34,57,26,22]). Gao et. al. [16] present a statistical characterization of the error associated with this algorithm.

Parametric methods are a reasonable compromise between low complexity and a reasonable approximation of the visual properties of the scene when the statistics of such a scene obeys the general assumptions imposed by the selected model. When these assumptions fail, non-parametric approaches are more suitable. A popular non-parametric approach is to use kernels ([48,56]). In this method, a kernel

is created around each of the previous samples and the density is estimated using an average over the kernels. While different kernels can be considered, the Normal kernel was proposed by Elgammal et. al. ([12,11]). The advantage of such approach is its ability to handle arbitrary shapes of the density function. However, it is computationally expensive, both in terms of memory requirements and running time.

Treating background subtraction as a state identification problem, some authors have utilized Hidden Markov Models (HMM) in order to reason about state changes either at the pixel level[47], global level [54], or some combination of the two[57]. In ([61,25]), edge features have been utilized in order to detect the objects at their boundaries. Furthermore, Eveland et. al. [13] propose a background model for stereo images.

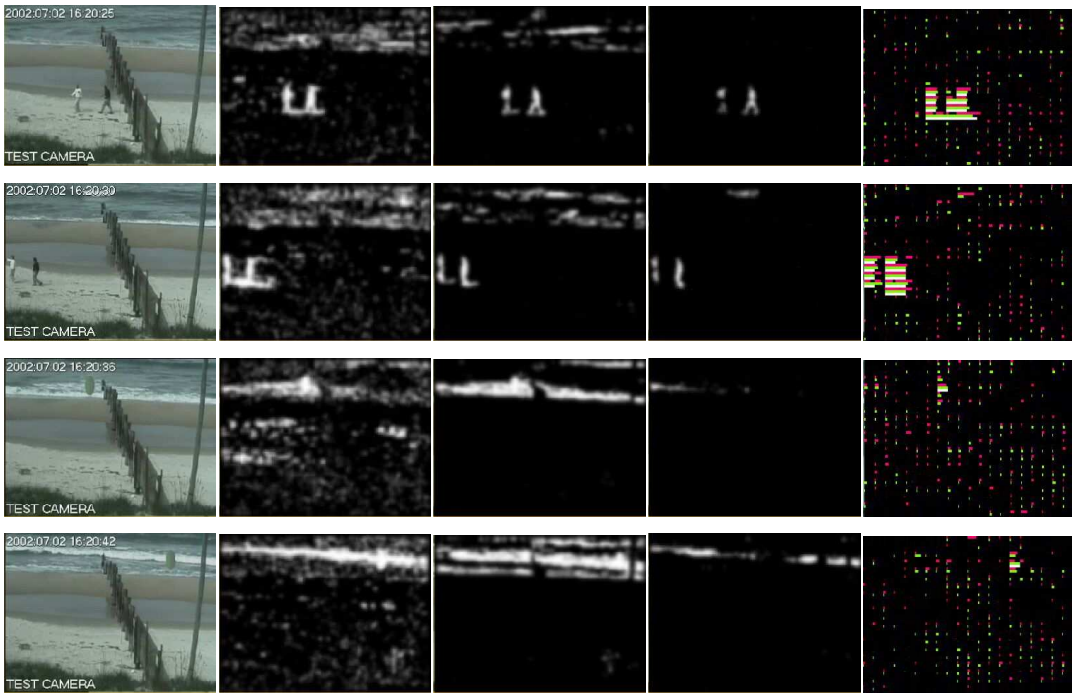
Last but not least, Oliver et. al. [40] model the background at the image level. Treating images as vectors, the means and variances are collected and *Principal Component Analysis (PCA)* over the difference from the mean is performed. Detection is recovered by projection of the input images onto the subspace of basis vectors and thresholding the Euclidean distance between the input image and the projected image (DFFS).

Although a number of non-predictive methods include some form of gradual forgetting for the past observations, most of the temporal information in the data is lost. Such an outcome does not affect the detection results when the scene is static or almost static and the changes in the appearance of the background are rather gradual. In scenes where there is a more drastic change in the background and the observed input is periodic and/or persistent, however, such temporal information is critical. For instance, sinusoidal data cannot be modeled well by such methods. For such scenes, more complex *predictive* methods that are able to capture such short-term temporal relationships are required.

1.2 Predictive Methods

The central idea behind *predictive* methods is to model the scene as a time series and to develop a dynamical model to recover the current input based on past observations. Methods of varying complexity have been considered in the past towards such an objective.

A Kalman-filter at the pixel level is the most popular dynamical model that has been considered in the literature ([46,28–30]). Within such an approach, the objective is to determine the current state of a system governed by a linear process. The estimation process is recursive: the previous *a posteriori* estimates are used to predict the new *a priori* estimates, while the current measurement is used to correct the estimates to obtain current *a posteriori* estimates.



(a) (b) (c) (d) (e)

Fig. 1. (a) Original Images. Detection result using (b) a mixture of Gaussians model [53], (c) a non-parametric model [12], and (d) the non-parametric model with low detection threshold. (e) Our method, where the presence of white in a block denotes detection (for full explanation of the color code used, kindly refer to Fig. 6)

Koller et. al. [30] use a simple state model that refers to the 1D value of the background intensity. The state is updated differently depending on whether it is hypothesized to be part of the background or not. Toyama et. al. [55] use a simpler version of the Kalman filter called *Weiner filter* that operates directly on the data rather than recursively capturing the essence of past observations in a state vector. Such a filter was able to capture simple repetitive behaviors (like flickering of the screen). In their system, integration of such a pixel-level method with region and frame-level information led to a promising solution.

Kalman filters that use more complex state vectors often include higher order motion characteristics such as velocity and acceleration and are be able to capture more complex dynamic behavior. However, even such extensions of the linear-filter driven methods suffer from various limitations: (i) restriction of these filters to linear models, and (ii) lack of the ability to capture complex relationships between neighboring pixels.

Fig. 1. demonstrates the strengths and the limitations of the state-of-the-art methods for a complex scene that is dynamic and exhibits non-stationary properties in time. Examples of scenes with such complex and dynamic behavior include ocean waves, waving trees, rain and moving clouds. Modeling such scenes is a topic that

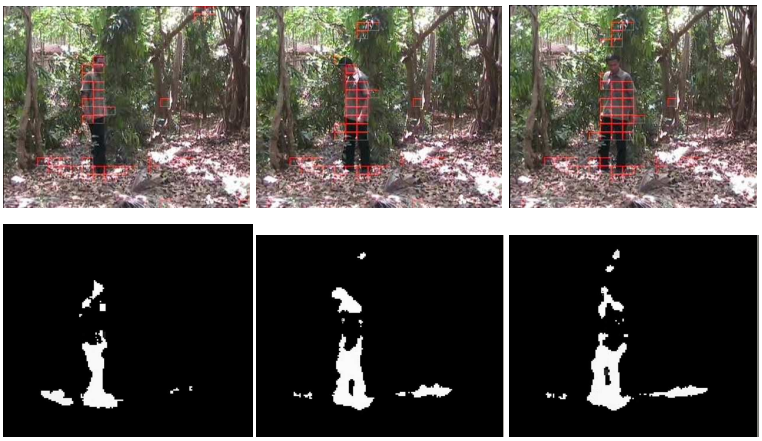


Fig. 2. Comparison of our method with an optical flow-based method of [35] on frames where the color matches the background and the object is stationary. Top Row: Detection using our method, shown by red squares around the detected blocks. Bottom Row: Results using [35].

has drawn attention recently. In [50,49], an approach that is based on modeling the co-occurrence of neighboring image patterns was proposed, while in [63], a Kalman filter is considered in a subspace [63]. Furthermore, the VANDAL project at the Washington University at St. Louis ([60,44,43]) has considered raw spatial and temporal derivatives and developed methods to deal with such scenes by clustering the data in such space. The main idea in this work is that persistent motion in a certain direction will create cylinders in this space that pass through the origin, and different directions will create cylinders in different orientations. In another related work by us [35], we have looked at the use of optical flow as an additional feature at each point in order to model the dynamics of the scene. While such a method can be effective modeling the dynamics of individual points, it cannot handle changes in the structure of the scene (spatial relationships). Hence, the current method, that handles both the structure as well as the temporal relationships can be considered as a more powerful and generic approach, although the modeling approximations needed to make the method real-time can have some adverse effect on the performance. In Fig. 2, we show how our method outperforms the method in [35] when color of the foreground matches with the background and the object is stationary. The performance of the two methods on the scene in Fig. 1 was quite similar.

In this paper, we extend the scope of predictive methods with the objective of handling scenes that exhibit more complex spatio-temporal pattern of change of the observation space. To this end, we present a predictive method based on a subspace analysis of the signal. The method is able to capture:

- long term dynamical characteristics of the scene, and
- temporal and structural relationships between different pixels

where detection is based on measures that are adaptive to the variation present in

the scene.

The main contribution of our approach is the use of the concept of dynamic series to model repetitive scenes. Towards addressing the real-time demand as well as the ill-posedeness of the problem, we consider a subspace approach. This has been initially proposed in a very different context to model and generate dynamic textures[10]. Our approach first assumes an evolving base, both in terms of the number as well as the base of retained eigen vectors. This is done through an incremental approach which to the best in our knowledge has been never considered before in the contest of background modeling. Then, a dynamical model in such a subspace is utilized to perform prediction. This dynamical model is also updated using an incremental approach for efficiency purposes. Furthermore, we introduce two novel measures to determine the appropriateness of the prediction mode with respect to the observation. One measure mostly encodes changes in appearance while the second encodes changes in the dynamics of the scene. Qualitative and Quantitative comparisons with existing methods validate the advantages of the proposed approach when dealing with dynamic backgrounds.

The reminder of this paper is organized as follows. The next section describes the feature space and the prediction model that is utilized to approximate the dynamic behavior of the scene. Initial and incremental construction of the utilized models is considered in Section 3. Detection measures are introduced in section 4. Finally, we conclude the paper with implementation details and experimental evaluation in section 5.

2 Scene Modeling

Let $\{\mathbf{I}(t)\}_{t=1\ldots T}$ be a given set of images. The central idea behind our approach is to generate a prediction mechanism that can determine the current frame using the latest k observed images. Such an objective can be defined mathematically as follows:

$$\mathbf{I}_{pred}(t) = f(\mathbf{I}(t-1), \mathbf{I}(t-2), \ldots, \mathbf{I}(t-k)) \tag{1}$$

where f, a k-th order function is to be determined. One possibility is to model such prediction mechanism via a multi-variate time series in the space of input images. However, it can be claimed that the modeling in such high-dimensional space is rather complex, contains redundancies and is not in a form that can be used directly for efficient prediction. This limitation can be addressed through the reduction of the dimensionality of the feature space according to some filter operators.

2.1 Feature space

Let $\{\phi_i\}_{i=1}^n$, be a filter bank and $\mathbf{s}_i(t) = \phi_i(\mathbf{I}(t))$, the output of the convolution between the operator ϕ_i and the image $\mathbf{I}(t)$. The outcome of such a convolution process can be combined into a vector that represents the current state $\mathbf{s}(t)$ of the system.

$$\mathbf{s}^T(t) = [s_1(t), \ldots, s_n(t)]$$

Examples of such filter operators include wavelet, gabor or anisotropic non-linear filters. Within the proposed framework, we adopt linear filters due to computationally efficient techniques for their implementation and their low complexity. Moreover, such filters are able to capture a significant amount of variations in real scenes.

Principal component analysis [3] ([27,1,9]) refers to a linear transformation of variables that retains - for a given number n of operators - the largest amount of variation within the training data. In a prediction mechanism, such a module can retain and recover in an incremental manner the core variations of the observed data.

The estimation of such operators will be addressed in the next section. In order to facilitate the introduction of the method, one can consider them known: $\{\phi_i = \mathbf{b}_i\}_{i=1}^n$, where $\mathbf{b}_i$ are the set of basis vectors. These can be considered to produce the state vector $\mathbf{s}(t)$:

$$\begin{aligned}\mathbf{s}(t) &= [\mathbf{b}_1^T \cdot \tilde{\mathbf{I}}(t), \mathbf{b}_2^T \cdot \tilde{\mathbf{I}}(t), \ldots, \mathbf{b}_n^T \cdot \tilde{\mathbf{I}}(t)]^T \\ &= \mathbf{B}^T \cdot \tilde{\mathbf{I}}(t)\end{aligned}$$

where $\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3, \ldots, \mathbf{b}_n]$ denotes the matrix of basis vectors, and $\tilde{\mathbf{I}}(t) = \mathbf{I}(t) - \bar{\mathbf{I}}$ denotes the mean ($\bar{\mathbf{I}}$) subtracted input.

2.2 Prediction mechanism

Based on the predictive model that was earlier introduced (Equation 1), one can define a similar concept in the state space:

$$\mathbf{s}_{pred}(t) = f(\mathbf{s}(t-1), \mathbf{s}(t-2), \ldots, \mathbf{s}(t-k)) \tag{2}$$

One can consider various forms (linear or non-linear) for the prediction function f. Non-linear mechanisms involve higher sophistication and can capture more complicated structures. However, the estimation of such functions is computationally expensive, suffers from instability and their incremental update that is crucial within the considered application is a challenging task.

[3] Also known as the Karhunen-Loeve expansion [32].

Linear models are a good compromise between low complexity and a fairly good approximation of the observed structure. Auto-regressive models of a certain order k can be considered to approximate and predict the actual observation based on the latest k feature vectors. In fact, it can be easily shown that due to the use of linear filters, a system governed by an auto-regressive model at the image level will lead to an auto-regressive system at the state level. Using such prediction model, the predicted state can be written as:

$$\begin{aligned}\mathbf{s}_{pred}(t) &= f(\mathbf{s}(t-1), (\mathbf{s}(t-2), \ldots, (\mathbf{s}(t-k)) \\ &= \sum_{i=1}^{k} \mathbf{A}_i \mathbf{s}(t-i)\end{aligned}$$

where $\mathbf{A}$ is an $n \times n$ prediction matrix. The prediction in the image space can then be reconstructed using the basis vectors:

$$\tilde{\mathbf{I}}_{pred}(t) = \mathbf{B} \cdot \mathbf{s}_{pred}(t)$$

Thus, the unknown variables of our scene model consist of the basis vectors and the auto-regressive matrices.

Visual appearance of indoor and outdoor scenes evolves over time. Global and local illumination changes, position of the light sources and tidal changes are examples of such dynamic behavior. One can account for such changes by continuously updating both the basis vectors and the prediction model according to the changes in the observed scene. In such an update, the discriminability of the model has to be preserved in order to perform accurate detection.

3 Model Estimation

Model estimation consists of determining the appropriate filters for state transformation and the parameters of the prediction model. Ideally, these estimates are to be recovered simultaneously. While asymptotically optimal solutions for the joint optimization problem do exist in the system identification literature ([31,41]), such estimation is computationally intensive both in terms of storage requirements and running time and is thus not amenable to a real-time solution for the types of high-dimensional problems that are to be dealt with in this paper. Therefore, we will estimate the two models separately.

In order to determine the filter bank for constructing the state space, we consider the use of principal basis vectors that are the linear operators that capture the most amount of variation in the training data. The estimation of such basis vectors from the observed data can be performed through the singular value decomposition. Within the proposed framework, one has to initiate the process using a certain

number of frames, and then continuously update the model parameters in order to capture the dynamic behavior of the scene. Therefore, two different learning mechanisms are used. The first, known as batch Principal Component Analysis (PCA) aims to recover the initial basis vectors. Subsequently, the incremental Principal Component Analysis (IPCA) mechanism continuously updates the basis vectors as the scene changes gradually.

3.1 Estimation of Basis Vectors

3.1.1 Batch PCA

Let $\{\mathbf{I}(t)\}_{t=1...T}$ be a column vector representation of the previous T observations. We assume that the dimensionality of this vector is d. One can estimate the mean vector $\bar{\mathbf{I}}$ and subtract it from the input to obtain zero mean vectors $\{\tilde{\mathbf{I}}(t)\}$. Given the set of training examples and the mean vector, one can define the $d \times d$ covariance matrix as:

$$\Sigma_{\tilde{\mathbf{I}}} = E\{\tilde{\mathbf{I}}(t)\tilde{\mathbf{I}}^T(t)\}$$

It is well known that the principal orthogonal directions of maximum variation for $\tilde{\mathbf{I}}(t)$ are the eigenvectors of $\Sigma_{\tilde{\mathbf{I}}}$[27]. Therefore, one can assume that the use of such vectors is an appropriate selection for the filter bank. Approximating the covariance matrix with the sample covariance matrix $\tilde{\mathbf{I}}_T\tilde{\mathbf{I}}_T^T$, where $\tilde{\mathbf{I}}_T$ is the matrix formed by concatenating the set of images $\{\tilde{\mathbf{I}}(t)\}_{t=1...T}$, one can compute such eigenvectors using the SVD of $\tilde{\mathbf{I}}_T$: $\tilde{\mathbf{I}}_T = \mathbf{U}D\mathbf{V}^T$. The basis vectors and the corresponding variance in the direction of such basis vectors can be obtained from the matrices U and D respectively.

The variance information can further be used to determine the number n of basis vectors required to retain a certain percentage of the variance in the data. Varying the number of vectors is important since the dynamic of the scene can change over time (for instance, high tide vs. low tide) and different parts of the scene also exhibit different characteristics. A high number of vectors for almost static scenes would be wasteful of resources while highly dynamic parts need a high number of vectors for proper modeling. Examples [4] of retained eigenvectors are shown in Fig. 3. Information related to their magnitude and number are given in Fig. 4.

3.1.2 Incremental PCA

The batch method is computationally inefficient and cannot be performed at each frame. A fast incremental method is an attractive alternative where the current estimate of the basis vectors is updated according to the new observation, while the effect of the previous observations is exponentially reduced. Several methods for

[4] The image is divided into equal size blocks to reduce complexity.

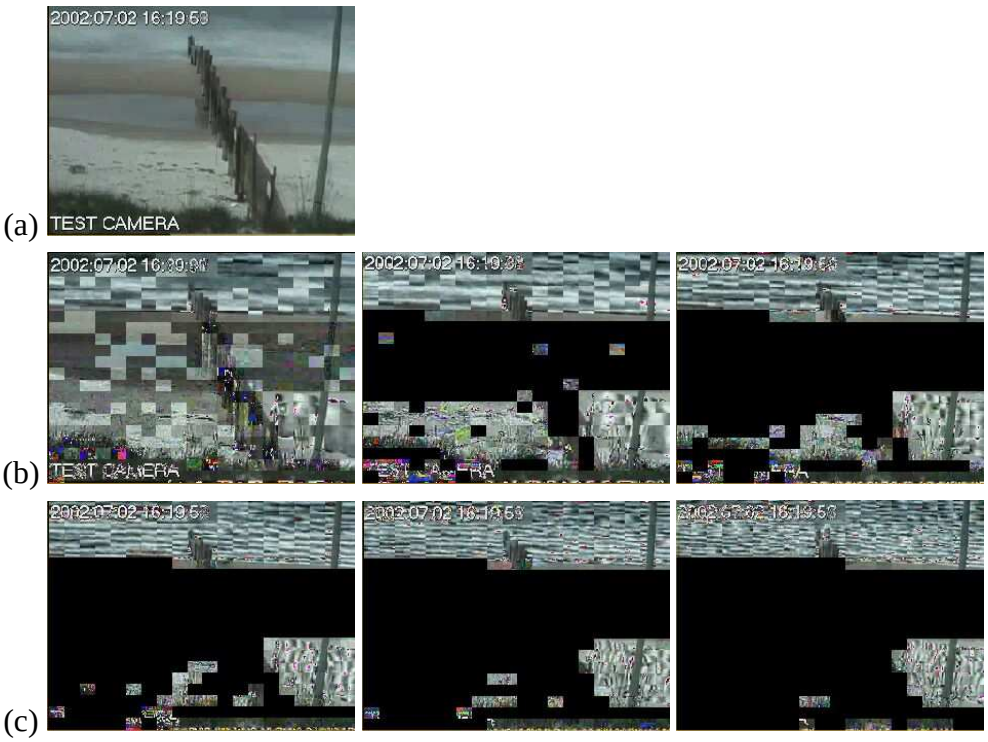


Fig. 3. Basis Vectors: (a) mean, (b,c) mean + constant $\times$(6 principal) basis vectors. Insignificant basis vectors are represented with dark color.

incremental PCA (IPCA) ([58,39,3]) can be considered. We adapt the method developed by Weng et. al.[58] to suit our application. The advantage of such method is that it does not require the computation of the covariance matrix. This is critical for applications where the size d of the input vectors is significant leading to a large $d \times d$ covariance matrix. The method is based on the statistical concept of efficient estimate and it was shown that it has better convergence properties than rest of the covariance-free methods.

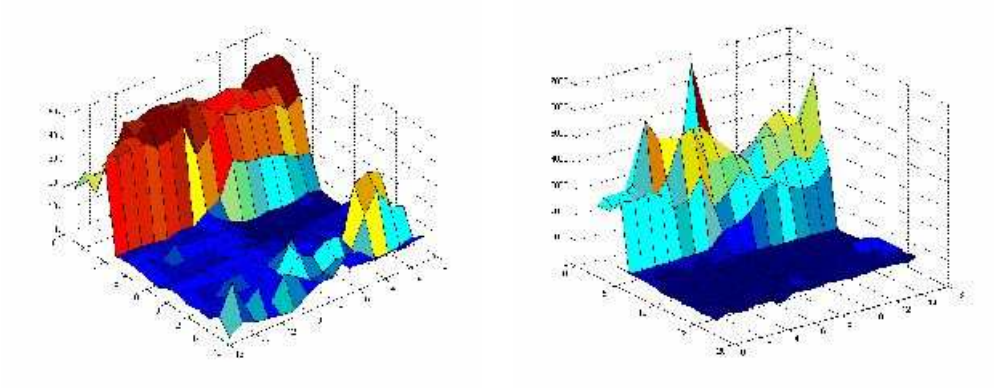
We recall that the objective of incremental update is to efficiently recover valid estimates for the mean of the data and the basis vectors over time. While update of the mean is trivial, the case of basis vectors is more complex.

Amnesic Mean
Let $\mathbf{I}_1, \ldots \mathbf{I}_t$ be the previous t observations. An amnesic mean can be computed from them that exponentially reduces the effect of past observations:

$$\bar{\mathbf{I}}_{t+1} = \left(\frac{t-l}{t+1}\right)\bar{\mathbf{I}}_t + \left(\frac{1+l}{t+1}\right)\mathbf{I}_{t+1}$$

where l is called the *amnesic* parameter that determines the rate of decay of the previous samples. If l is a fixed multiple of t ($l = \lambda t$), one obtains exponential decay.

Update of the Basis Vectors



(a) (b)

Fig. 4. (a) Number of retained eigenvectors, (b) Magnitude of the largest eigenvalue.

Let $\{\mathbf{b}_1, \ldots, \mathbf{b}_n\}$ be the current set of estimates of the basis vectors. For reasons that become apparent later, these vectors are not normalized, although they are orthogonal [5]. Now, suppose we observe a new image $\mathbf{I}(t+1)$ and subtract the mean $\bar{\mathbf{I}}$ to obtain $\tilde{\mathbf{I}}(t+1)$. Then, we update the first basis vector $\mathbf{b}_1$ by essentially "pulling" it in the direction of $\tilde{\mathbf{I}}(t+1)$ by an amount equal to the projection of $\tilde{\mathbf{I}}(t+1)$ onto the unit vector along $\mathbf{b}_1$:

$$\mathbf{b}_1' = \left(\frac{t-l}{t+1}\right) \cdot \mathbf{b}_1 + \left(\frac{l+1}{t+1}\right) \left(\frac{\mathbf{b}_1 \cdot \tilde{\mathbf{I}}(\mathbf{t+1})}{\|\tilde{\mathbf{I}}(\mathbf{t+1})\| \|\mathbf{b_1}\|}\right) \cdot \tilde{\mathbf{I}}(t+1)$$

Here, $\frac{\mathbf{b}_1 \cdot \tilde{\mathbf{I}}(\mathbf{t+1})}{\|\mathbf{b_1}\|}$ is the projection of $\tilde{\mathbf{I}}(t+1)$ in the direction of $\mathbf{b}_1$, and $\frac{\tilde{\mathbf{I}}(t+1)}{\|\tilde{\mathbf{I}}(t+1)\|}$ is the unit vector in the direction of $\tilde{\mathbf{I}}(t+1)$.

Next, we compute the residue $\mathbf{R}_1$ of $\tilde{\mathbf{I}}(t+1)$ on $\mathbf{b}_1$:

$$\begin{aligned} \mathbf{R}_1 &= \tilde{\mathbf{I}}(t+1) - Proj_{\mathbf{b}_1}(\tilde{\mathbf{I}}(t+1)) \\ &= \tilde{\mathbf{I}}(t+1) - \left(\frac{\mathbf{b}_1 \cdot \tilde{\mathbf{I}}(t+1)}{\|\mathbf{b}_1\|^2}\right) \cdot \mathbf{b}_1 \end{aligned}$$

This residue is perpendicular to $\mathbf{b}_1$ and is used to "pull" $\mathbf{b}_2$ in the direction of $\mathbf{R}_1$ by an amount equal to the projection of $\mathbf{R}_1$ onto the unit vector along $\mathbf{b}_2$:

$$\mathbf{b}_2' = \left(\frac{t-l}{t+1}\right) \cdot \mathbf{b}_2 + \left(\frac{l+1}{t+1}\right) \left(\frac{\mathbf{b}_2 \cdot \mathbf{R}_1}{\|\mathbf{R}_1\| \|\mathbf{b}_2\|}\right) \cdot \mathbf{R}_1$$

The residue $\mathbf{R}_2$ is calculated similarly:

$$\begin{aligned} \mathbf{R}_2 &= \mathbf{R}_1 - Proj_{\mathbf{b}_2}(\mathbf{R}_1) \\ &= \mathbf{R}_1 - \left(\frac{\mathbf{b}_2 \cdot \mathbf{R}_1}{\|\mathbf{b}_2\|^2}\right) \cdot \mathbf{b}_2 \end{aligned}$$

[5] However, they can be normalized for use in the background model.

This residue is perpendicular to the *span* of $< \mathbf{b}_1\mathbf{b}_2 >$. This procedure is repeated for each subsequent basis vector such that the basis vector $\mathbf{b}_j$ is pulled towards $\tilde{\mathbf{I}}(t+1)$ in a direction perpendicular to the span of $< \mathbf{b}_1 \ldots \mathbf{b}_{j-1} >$.

Zhang and Weng [58,62] have proved that for the stationary case where the scene characteristics don't change over time, $\mathbf{b}_i \rightarrow \pm\lambda_i \mathbf{e}_i$ as $t \rightarrow \infty$. Here, λ_i is the i-th largest eigenvalue of the covariance matrix $\Sigma_{\tilde{\mathbf{I}}}$, and $\mathbf{e}_i$ is the corresponding eigenvector. Note that the obtained vector has a scale of λ_i and is not a unit vector. Therefore, in our application we store these unnormalized vectors. The magnitude yields the eigenvalue and the normalization yields the eigenvector at any given time. Also important to note is that the estimated eigenvectors may not be perpendicular to each other in the intermediate stages but do converge to perpendicular vectors over time.

3.2 Estimation of the predictive model

3.2.1 Batch Update

As stated earlier, we will use a linear auto-regressive model to model the transformation of states:

$$\mathbf{s}_{pred}(t) = \sum_{i=1}^{k} \mathbf{A}_i \mathbf{s}(t-i) + \mathbf{z}(t)$$

for a k-th order auto-regressive model, where $\mathbf{z}(t) \approx WN(0, \Sigma_{pred})$ is the prediction error.

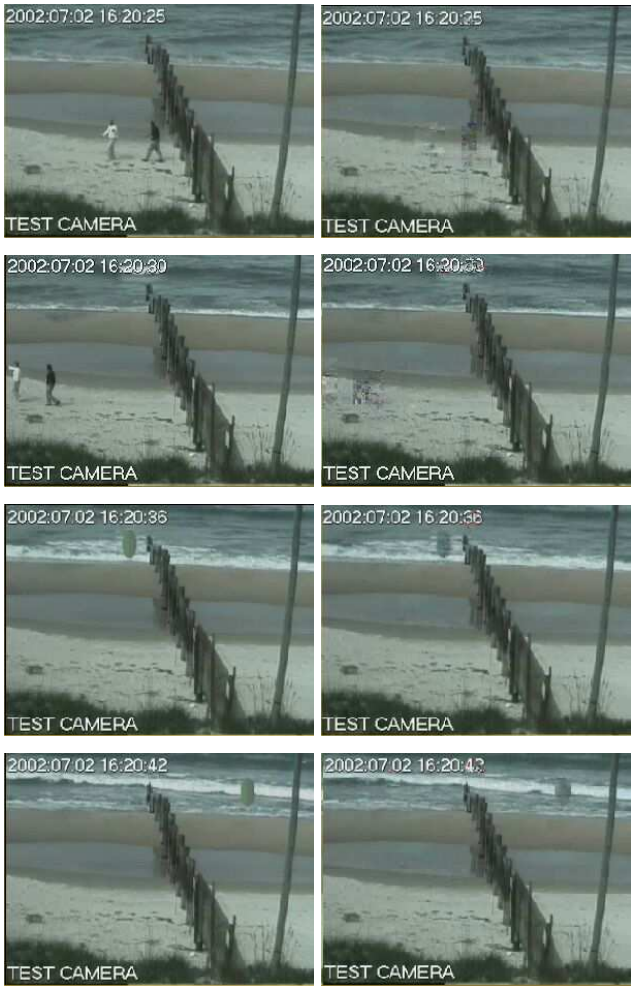
We assume that the noise $\mathbf{z}(t)$ has a normal distribution, is independently and identically distributed (i.i.d.), and has a diagonal covariance matrix. Under this assumption, the maximum likelihood solution for the parameters $\mathbf{A}$ can be computed[4] by minimizing the least squares error $\|\mathbf{s}(t) - \sum_{i=1}^{k} \mathbf{A}_i \mathbf{s}(t-i)\|_2^2$. Such minimization is readily obtained by the well-known method of normal equations. For instance, for $k = 1$, if two matrices $\mathbf{S2}$ and $\mathbf{S1}$ are formed by concatenating the previously observed state vectors:

$$\mathbf{S2} = [\mathbf{s}(2), \mathbf{s}(3), \ldots, \mathbf{s}(t)],\ \mathbf{S1} = [\mathbf{s}(1), \mathbf{s}(2), \ldots, \mathbf{s}(t-1)]$$

then, the solution may be obtained simply as:

$$\mathbf{A_1} = \mathbf{S2} \cdot \mathbf{S1}^T \cdot (\mathbf{S1} \cdot \mathbf{S1}^T)^{-1} \tag{3}$$

While one can claim certain limitations introduced by such a simple noise model, we have found that such a simplification was possible within the considered application without affecting the overall performance of the system. However, for handling more complex noise processes, one may consider more advanced techniques available in the multivariate time series literature [4]. Such techniques are



(a) (b)

Fig. 5. (a) Input signal, (b) Prediction.

quite computation-intensive, however, and thus quite unsuitable for our real-time application.

3.2.2 Incremental Update

Similar to the basis vector case, estimating the auto-regressive model at each time step is a time-consuming process. Incremental methods to update the $\mathbf{A}$ matrices could be considered to address this limitation[5]. To this end, we utilize the concept of storing the sufficient statistics for the problem and use a simple exponential forgetting scheme for the purpose. Consider the case of $k = 1$ described in Equation 3. In this equation, there are essentially two terms: $\mathbf{S2S1}^T$, and $\mathbf{S1S1}^T$. Each of the terms in these two matrices is formed as a sum of the product of two components

of the state vector. For instance, the i,j-th term of $\mathbf{S2S1}^T$ is given by:

$$s2s1_{i,j} = \sum_{k=1}^{t-1} s_i(k+1)s_j(k)$$

In order to obtain exponential forgetting of the prior observations, one can incrementally update such components using an update parameter λ:

$$s2s1_{i,j}(t+1) = (1-\lambda)s2s1_{i,j}(t) + \lambda s_i(t+1)s_j(t)$$

Using such incremental updates for the two matrices, one can readily compute the new least-squares solution. One may note that although such solution was possible for the incremental update of the basis vectors, the size of the original vectors prohibits the maintenance and inversion of such matrices at each time step. However, in the state space, the dimensionality of the problem is greatly reduced leading to the feasibility of such solution. If computational time is limited, it is also possible to stagger the updates to the model both in terms of time (i.e. skipping some frames) and space (i.e. updating only some blocks in each frame).

4 Detection

The simplest mechanism to perform detection is through a comparison between the prediction and the actual observation. Under the assumption that the auto-regressive model was built using background samples, such technique will provide poor prediction for new objects while being able to capture the background. Efficiency and robustness, however, dictate that this comparison be performed in the state space and that the statistics of the training data be taken into consideration in the comparison.

Two types of changes in the signal may be considered for detection: (1) "structural" change in the appearance of pixel intensities in a given region, and (2) change in the motion characteristics of the signal. Measures are developed in order to detect each of these changes.

4.1 Structural Change

In order to develop the approach for estimating structural change in the signal, we begin by reviewing some concepts in Principal Component Analysis and its relationship to density estimation in a multi-dimensional space. The principal component analysis decomposes the vector space $\mathbb{R}^d$ into two mutually exclusive and complementary subspaces: the principal subspace $F = \{b_i\}_{i=1}^n$ containing the first

n principal components and its orthogonal complement $\bar{F} = \{b_i\}_{i=n+1}^{d}$. Then, using the definition $\mathbf{s} = \mathbf{B}^T \cdot \tilde{\mathbf{I}}$, $\mathbf{B}$ being the matrix of basis vectors, the residual reconstruction error for an input vector $\tilde{\mathbf{I}}(t)$ is defined as [15]:

$$\epsilon^2(\tilde{\mathbf{I}}) = \sum_{i=n+1}^{d} s_i^2 = \|\tilde{\mathbf{I}}\|^2 - \sum_{i=1}^{n} s_i^2$$

This is the component of the L_2 norm of $\tilde{\mathbf{I}}(t)$ in the orthogonal subspace $\bar{F}$ and is referred to as the "distance-from-feature-space" (DFFS) ([36,37]). This is easily computed from the first n principal components and the L_2-norm of $\tilde{\mathbf{I}}$.

Let us assume a Gaussian model for the density in high-dimensional space. More complicated models for the density, like mixture-of-Gaussians, or non-parametric approaches can also be considered and easily integrated by explicitly building a background model on the state space. However, for simplicity and ease of use, we will restrict ourselves to Gaussian densities in this paper. If we assume that the mean $\bar{\mathbf{I}}$ and covariance Σ of the distribution has been estimated robustly, the likelihood of an input $\mathbf{I}$ to belong to the background class Ω is given by:

$$p(\mathbf{I}|\Omega) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} exp\left(-\frac{1}{2}(\mathbf{I}-\bar{\mathbf{I}})^T\Sigma^{-1}(\mathbf{I}-\bar{\mathbf{I}})\right)$$

The sufficient statistic for characterizing this likelihood is the *Mahalanobis distance*:

$$d(\mathbf{I}) = \tilde{I}^T\Sigma^{-1}\tilde{I}$$

where $\tilde{\mathbf{I}} = \mathbf{I} - \bar{\mathbf{I}}$. Utilizing the eigenvalue decomposition of Σ: $\Sigma = \mathbf{B}\Lambda\mathbf{B}^T$, we can rewrite:

$$d(\mathbf{I}) = \mathbf{s}^T\Lambda^{-1}\mathbf{s}$$

since $\mathbf{B}^T\tilde{\mathbf{I}} = \mathbf{s}$. Since Λ is diagonal, one can rewrite this further as:

$$d(\mathbf{I}) = \sum_{i=1}^{d} \frac{s_i^2}{\lambda_i}$$

where λ_i is the i-th eigenvalue. If we seek to estimate $\tilde{d}(\mathbf{I})$ using only the n principal projections, one can formulate an optimum estimator for $\tilde{d}(\mathbf{I})$ as follows:

$$\begin{aligned}\tilde{d}(\mathbf{I}) &= \sum_{i=1}^{n} \frac{s_i^2}{\lambda_i} + \frac{1}{\rho}\left[\sum_{i=n+1}^{d} s_i^2\right] \\ &= \sum_{i=1}^{n} \frac{s_i^2}{\lambda_i} + \frac{1}{\rho}\epsilon^2(\tilde{\mathbf{I}})\end{aligned} \tag{4}$$

where $\epsilon^2(\tilde{\mathbf{I}})$ is the DFFS defined above and can be computed using the first n principal components. Moghaddam et. al. [37] have shown that an optimal ρ in terms

of a suitable error measure based on the Kullback-Leibler divergence [8] is:

$$\rho^* = \frac{1}{d-n} \sum_{i=n+1}^{d} \lambda_i$$

We propose the use of $\tilde{d}(\mathbf{I})$ as the first detection measure r_1. Such measure can be determined by utilizing only the first n principal components. It is an optimum measure for estimating the distance from the Gaussian density represented by the principal component analysis such that the covariances of the data are properly taken into account while estimating the difference. High values of such distance measure have a simple interpretation: the original vector is not close to the training data, and thus corresponds to a new object in the scene. In other words, this is a measure of change of the structure of the block appearance. Such case can occur either because of changes in the appearance of the scene (color), or because of structural changes. Therefore, such technique can better detect objects than the pixel-based background subtraction techniques that consider each pixel individually without consideration of the relationships among them. On the other hand, the drawback is that the boundary of the objects in not delineated exactly.

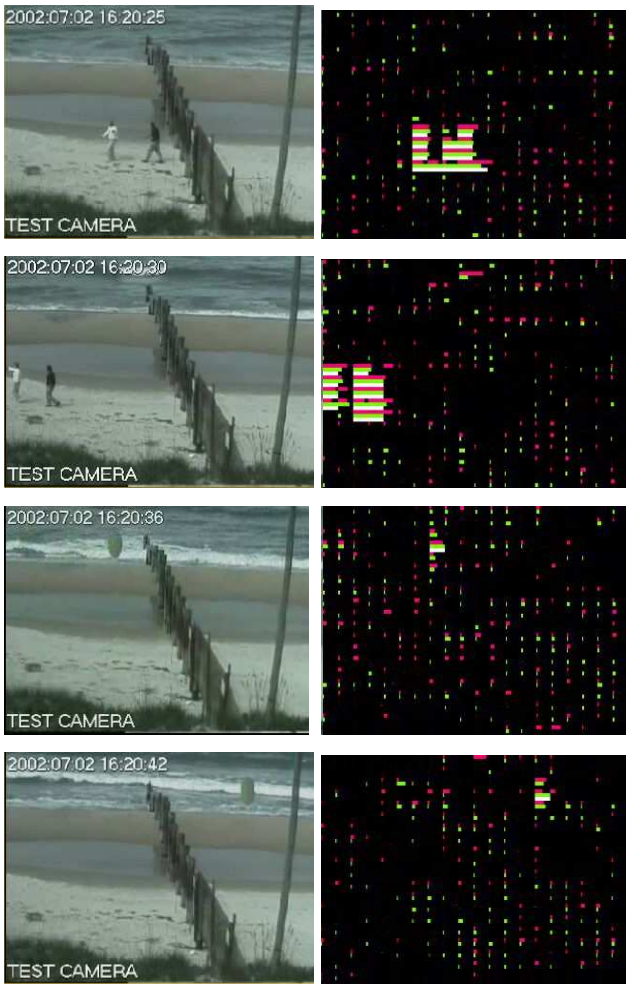
4.2 Change in Motion Characteristics

While the measure r_1 can account for changes of appearance in the structural sense, it would fail to capture changes in the temporal domain. This can occur when temporal information appears in a different order than the one for the background. To this end, one can consider the *SSD* (Sum of squared differences) error between the input and predicted image, which can be expressed as the square of the L_2 norm of the difference between the vectorized input and predicted images: $\|\mathbf{I} - \mathbf{I}_{pred}\|_2^2$. Since any vector $\mathbf{I}$ may be written in terms of its components along the basis vectors, $\mathbf{I} = \sum_{i=1}^{d} s_i \mathbf{B}_i$, we may write:

$$\mathbf{I} - \mathbf{I}^{pred} = \sum_{i=1}^{d} s_i \mathbf{B}_i - \sum_{i=1}^{d} s_i^{pred} \mathbf{B}_i = \sum_{i=1}^{d} (s_i - s_i^{pred}) \mathbf{B}_i$$

Therefore, the norm of this vector may be computed thus:

$$\begin{aligned} \|\mathbf{I} - \mathbf{I}_{pred}\|_2^2 &= \|\sum_{i=1}^{d} (s_i - s_i^{pred}) \mathbf{B}_i\|_2^2 \\ &= \sum_{i=1}^{d} (s_i - s_i^{pred})^2 \\ &= \sum_{i=1}^{n} (s_i - s_i^{pred})^2 + \sum_{i=n+1}^{d} (s_i)^2 \end{aligned}$$



(a) (b)

Fig. 6. (a) Input frames, (b) Detection Components. In each block, green represents r_1, pink shows r_2 and white represents detection by combining r_1 and r_2. r_1 and r_2 are scaled with respect to their threshold values, the "bar" occupying the whole width the measure equals or exceeds this threshold. Thus, if either the red or green bar is "full", it denotes detection. In that case, we also make the third "white" bar full (the white bar can be only full or empty, depending on whether there is a detection or not).

since the prediction is made from only the first n components, and therefore $s_i^{pred} = 0, i = n+1 \ldots d$. Recalling the definition of $\epsilon^2(\tilde{\mathbf{I}})$, we obtain:

$$\|\mathbf{I} - \mathbf{I}_{pred}\|_2^2 = \sum_{i=1}^{n}(s_i - s_i^{pred})^2 + \epsilon^2(\tilde{\mathbf{I}})$$

Again, this quantity may be computed from only the first n components. Since the effect of the second term has already been captured in r_1, we define

$$r_2(t) = \sum_{i=1}^{n}(s_i - s_i^{pred})^2 = \|\mathbf{s} - \mathbf{s}^{pred}\|_2^2$$

where the state vectors are considered only up to the n principal components. Since this is simply the least squares prediction error at the state level, one may further normalize this error by the covariance of the observed prediction error at the state level. Thus, the measure $r_2(t)$ may alternately be defined using the mahalanobis distance as follows:

$$r_2(t) = (\mathbf{s} - \mathbf{s}^{pred})^T \Sigma_{pred}^{-1} (\mathbf{s} - \mathbf{s}^{pred})$$

Such a measure captures the change in the motion characteristics at a structural level. Objects following motion trajectories different than the learned ones will trigger high values for r_2. Thus, such a metric is an additional cue for detection based on structural motion that has not been considered in traditional background adaptation methods ([20,12]). Fusion of the two metrics is performed by triggering a detection if either of them triggers a detection.

In the design of r_2, we have neglected the term ϵ^2. Due to this, r_2 only performs the extra function of detecting changes in the subspace of basis vectors. When the current input does not project onto the subspace, r_1 will trigger an alarm, but it is quite possible that r_2 will not trigger since the projections onto the original subspace may be low in magnitude. On the other hand, only r_2 is flagged when the structural appearance is similar, but the objects move differently. For many of the examples shown in the paper, one may notice that the detection happens due to only one of the two measures. This is not a drawback, however, since the two measures should be taken in conjunction with each other and not separately.

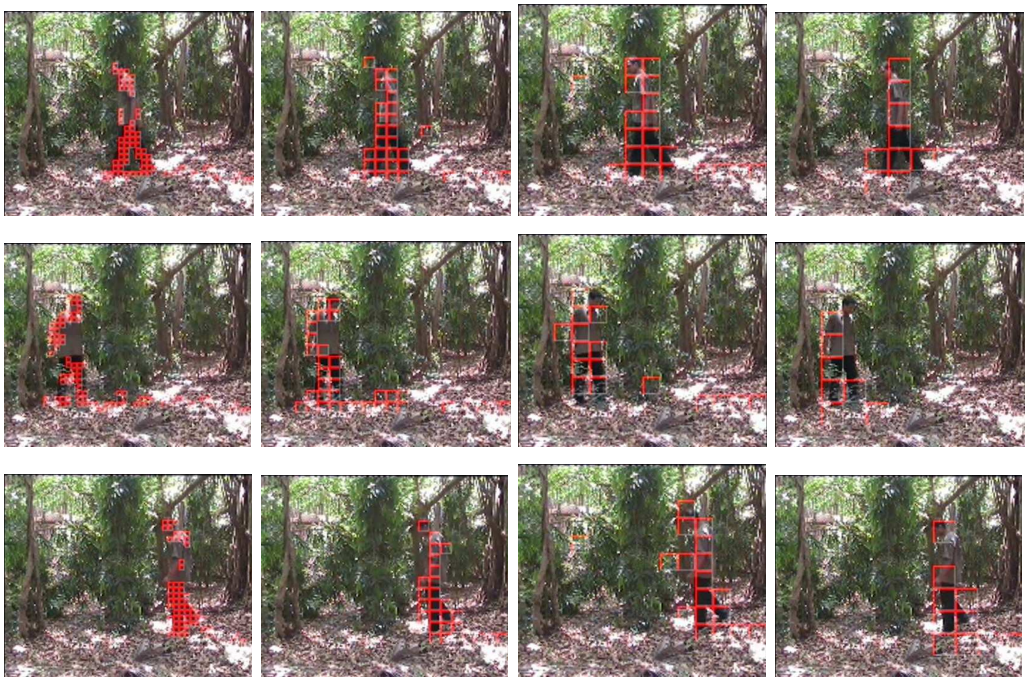
An example of the detection mechanism is shown in Fig. 6. In order to represent the two-dimensional feature space, a color representation was considered where green corresponds to r_1 and pink to r_2. In each block, the length of the color vectors corresponds to the magnitude of the detection measures while detection is represented by white color.

5 Implementation and Experiments

5.1 Implementation Details

Real-time processing is a common requirement of video surveillance. In particular, when dealing with techniques that address background adaptation, such requirement is strictly enforced. Furthermore, changes of the background structure should lead to an updated model in order to preserve satisfactory detection rates.

Computing the basis components for large vectors is a time consuming operation. Optimal algorithms for singular value decomposition of an $m \times n$ matrix take



(a) (b) (c) (d)

Fig. 7. Detection using different block sizes: (a) 4x4 (b) 8x8 (c) 12x12 and (d) 16x16. While one gets better detection using higher block sizes, there is a loss of precision/object localization. Also, one may not get detection in a large block if the portion of the object in the block is too small (this can be seen in some blocks in the results in the last column.

$O(m^2n + n^3)$) time[17]. A simple way to deal with such complexity is by considering the process at a block level. To this end, we divide the image into blocks and run the algorithm independently on each block. For each of these blocks, the number of components retained is determined dynamically by the singular values which refer to the standard deviation in the direction of basis vectors. A given percentage of the total variation is retained.

Too small a block size leads to loss of detection accuracy while a very large block size leads to high computational cost and loss of precision/localization in the detection of the object. Furthermore, if the portion of the block covered by the object is too small, the object may not be detected. The dependence of performance on the block size is illustrated in Fig. 7. In all of the other experiments in this paper, we have chosen 16x12 as the block size as a reasonable tradeoff point. Such a choice leads to a quasi real-time ($\sim$5fps) implementation for a 340×240 3-band video stream on a 2.2 GHz Pentium IV processor machine, where the input vectors were formed by concatenating the R, G and B values for all the pixels in a block.

5.2 Experimental Results

In order to validate the proposed technique, we conducted experiments on three different types of scenes. First, we show results on the challenging scene of the ocean front. Such scene involves wave motion, blowing grass, long-term changes due to tides, global illumination changes, shadows etc. An assessment on the performance of the existing methods [6] ([52,12]) is shown in Fig. 1. In order to detect an object, the detections at the pixel level were grouped together using simple neighborhood analysis to form larger clusters. While existing techniques were able to cope to some extent with the appearance change of the scene, one can claim that their performance exhibits certain limitations for video-based surveillance systems. The detection of events was either associated with a non-acceptable false alarm rate or the detection was compromised when focus was given to reducing the false alarm rate. On the other hand, our algorithm was able to detect events of interest in the land and simulated events on the ocean front as shown in Fig.s 1 and 6.

The essence of the approach is depicted in Fig. 5 and 6. Observation as well as prediction are presented for comparison. Visually, one can conclude that the prediction is rather close to the actual observation for the background component. On the other hand, prediction quality deteriorates when non-background structures appear in the scene. A more elaborate technique to validate prediction is through the detection process as shown in Fig. 6.

A quantitative evaluation of the method can be considered through the *ROC* (Receiver-Operator Characteristics) curves, where the detection rate (number of correct detected foreground objects/total number of foreground objects) is plotted against the false alarm rate (number of wrong detections/number of frames). Fig. 10.(a) illustrates the *ROC* characteristics of our method for the sequence of the ocean front (Fig. 1). Also shown for comparison purposes are the *ROC* curves for the existing techniques. As can be seen from the plots, there was a substantial improvement in the results as compared to existing methods. Most of this improvement was observed in the region of the ocean front and the blowing grass; the improvement in the static parts of the scene, although significant, was not as marked and the performance of all three methods can be considered satisfactory in these regions.

The second scene we consider is a typical traffic monitoring scene where the trees were blowing due to the wind (Fig. 8). Results comparable to existing methods were obtained for the static parts of the scene (e.g. road). At the same time, false detections in the tree area were significantly reduced as compared to traditional methods. This was achieved without any manual parameter adjustments and objects even behind the trees but visible in spots through the holes in the structure were detected in some cases. These objects would have been impossible to detect with

[6] Using our implementation.

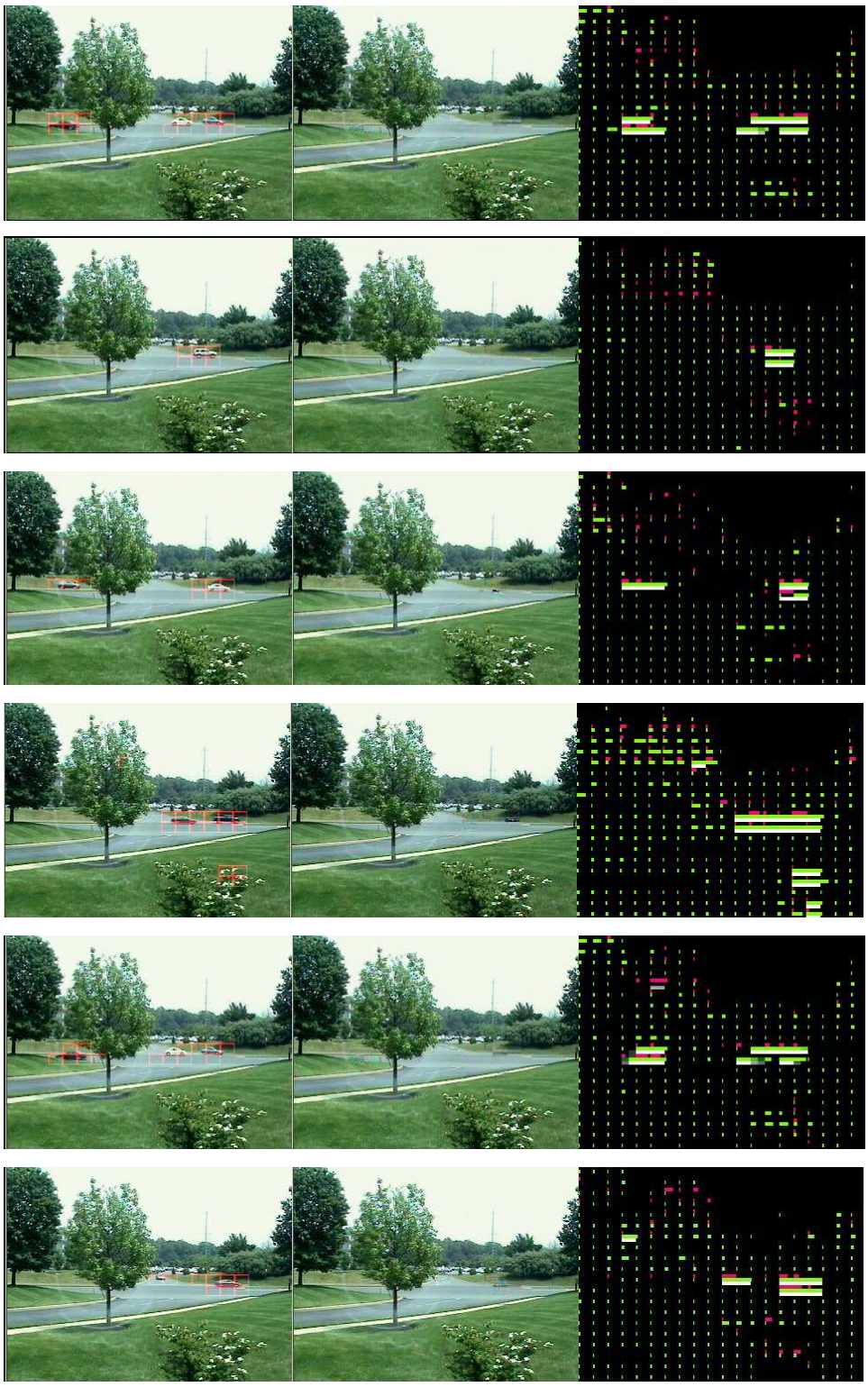


Fig. 8. Some results from a sequence of a road with waving trees. *Left*: Input signal, with detection denoted by red squares around the detected block, *Middle:* Predicted signal, *Right:* Block-wise response of the detection measures. As before, green represents r_1, pink represents r_2 and white represents detection in a block. Note that the algorithm is largely able to handle waving trees (with the exception of very large movements, as in the fourth image) automatically without any parameter adjustments for different blocks. This sequence has been uploaded onto the submission website.

traditional methods that typically have to rely on pixel neighborhood analysis to remove outliers and thus lack the ability to correlate far away pixels. The *ROC* curve for this sequence is shown in Fig. 10.(b).

The third scene is again a traffic monitoring scene where the changes in the observation space occur due to rain and high sensor noise due to low light conditions (Fig. 9). Again, the algorithm was able to adapt to such conditions and detect objects inspite of such variation. Local changes due to rain and sensor noise were handled by the correct modeling of the variance of the data in the subspace. The *ROC* curve for this sequence is shown in Fig. 10.(c). Sequences showing the results for these three scenes have been uploaded to the *PAMI* website.

6 Discussion

In this paper we have proposed a prediction-based on-line method for the modeling of dynamic scenes. The core contribution of our approach is the integration of a powerful set of filter operators with a linear prediction model towards the detection of events in a dynamic scene. Furthermore, we have proposed the use of on-line adaptation techniques to maintain the selection of the best filter operators and the prediction model. Last but not least, appropriate detection measures have been developed that are adaptive to the complexity of the scene.

The approach has been tested and validated using a challenging setting: detection of events on the coast line and the ocean front (Fig. 11,12). Large scale experiments were conducted on a recorded representative video of several hours that involved real events (Fig.s 11,12) as well as simulated ones (Fig. 6). The proposed technique was able to detect such events with a minimal false alarm rate. Detection performance was a function of the complexity of the observed scene. High variation in the observation space reflected to a mechanism with limited discrimination power. The method was able to adapt with global and local illumination changes, weather changes, changes of the natural scene, etc. Validation has been performed by comparing our technique with state-of-the-art methods in background adaptation (Fig.s 1,10). Our method could meet and overcome in some cases the performance of these techniques for stationary scenes, while being able to deal with more complex and evolving natural scenes.

The enhanced performance of the method may be attributed to two factors. First, as opposed to the traditional method of pixel level detection, the consideration of pixels at the block level helps in improving the detection rates since the inference is based on more information and can now take into account the correlation between neighboring pixels. Secondly, the use of a dynamical model helps in handling the dynamic nature of the scenes considered. The drawback of using a block-based approach, however, is that the boundary of the objects cannot be delineated exactly.

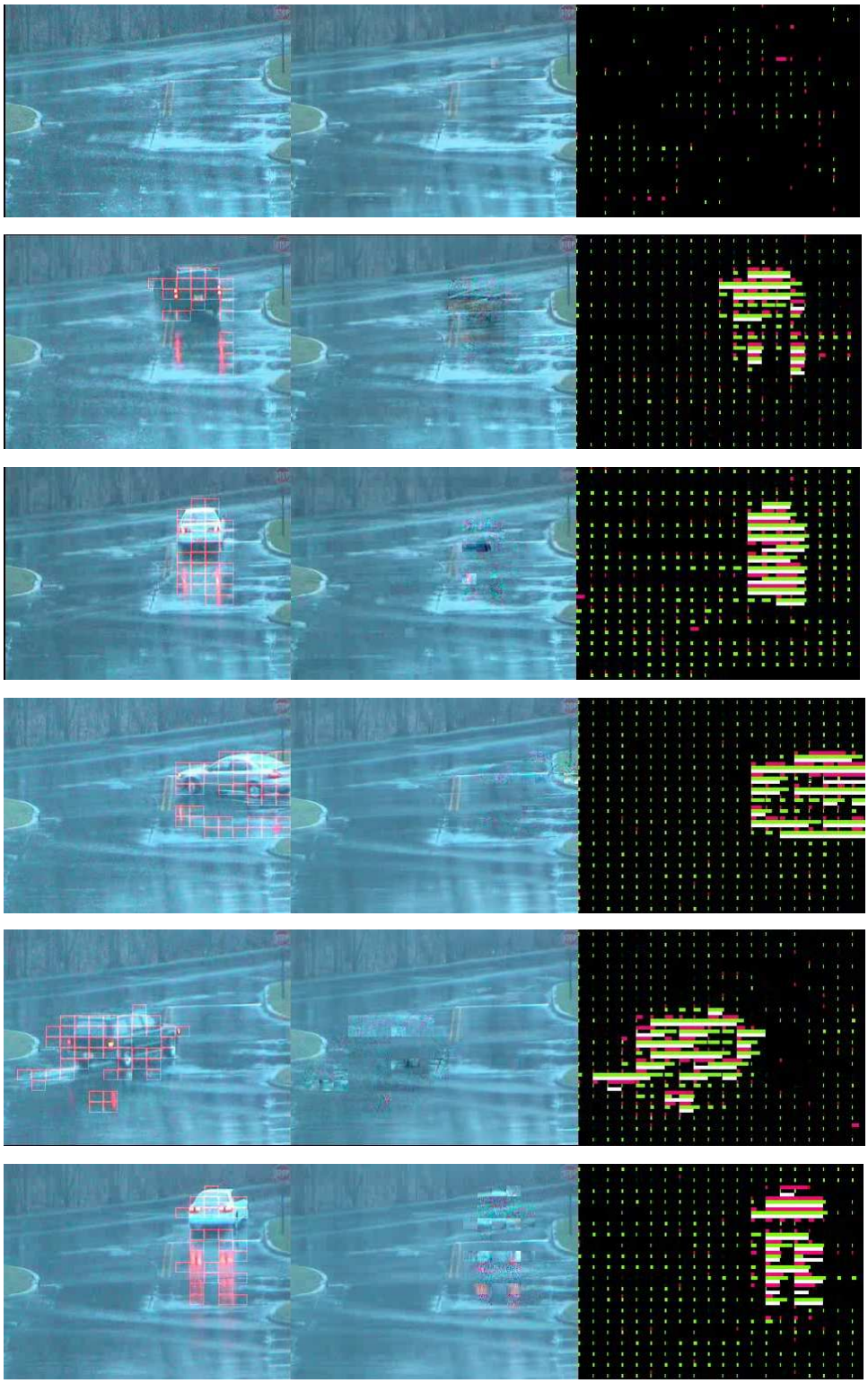
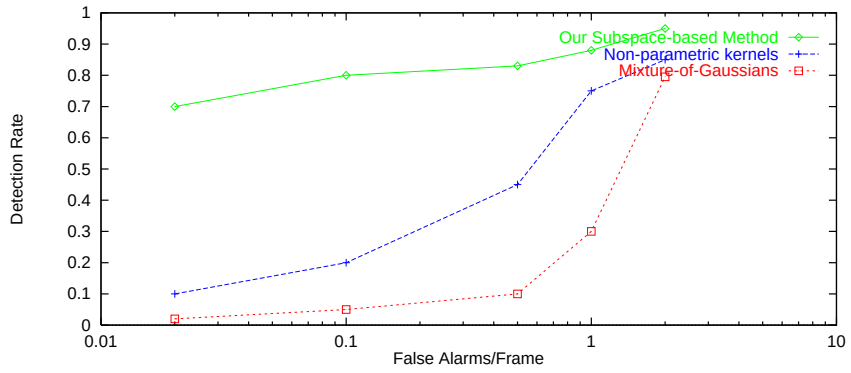
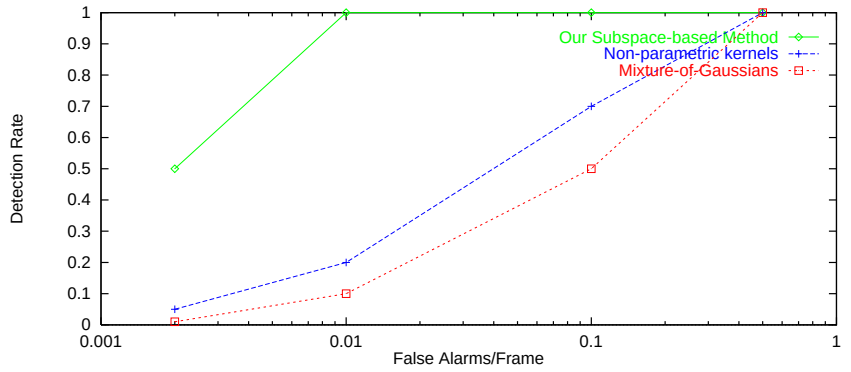


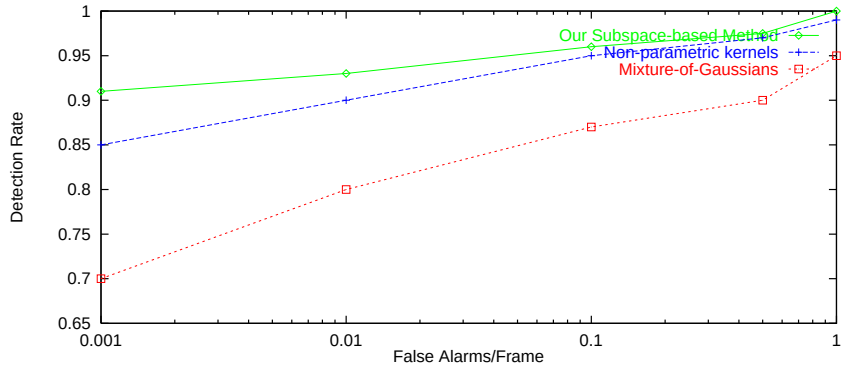
Fig. 9. Detection of traffic in rain and low light conditions. *Left*: Input signal, with detected blocks shown by a red squares around them, *Middle:* Predicted signal, *Right:* Block-wise response of the detection measures. As before, green represents r_1, pink represents r_2 and white represents detection in a block. A movie clip depicting results for this sequence has been uploaded to the submission website.



(a)



(b)



(c)

Fig. 10. *ROC* curves for (a) the "Ocean" sequence (Fig. 6) (b) the "Waving trees" sequence (Fig. 8), and (c) the "Rain" Sequence (Fig. 9) for: (i) Mixture-of-Gaussians model, (ii) Non-parametric Kernels, and (iii) Our method. Note that the improvement in performance of our method compared to prior work is more significant for ocean waves and waving trees than for rain since these scenes have persistent motion across space while it is possible to approximate the effect of rain as sensor noise due to the largely local effects.

For future work, one can explore the use of non-linear operators that can better capture the variation of the data leading to a better discriminability for the model. More elaborated prediction mechanisms can also be investigated. More complex detection mechanisms can also be considered by utilization of more complex rep-

resentation models for the multi-dimensional density. Last but not least, one can consider the modeling of scenes that exhibit some other more complex patterns of dynamical behavior. More sophisticated tools that take decisions at a higher level and are able to represent more sophisticated patterns of dynamical behavior is an interesting topic for further research.

Acknowledgments

The subspace method for scene prediction was inspired by the work of Soatto et. al ([51,10]). Moreover, they provided us with an implementation of their algorithm for which we are very grateful. We are also grateful to Silviu Minut for making available to us his implementation of Incremental PCA. The authors would also like to thank Visvanathan Ramesh and Radu Balan for some valuable discussions on the topic.

References

[1] K. Arun and S. Kung. Balanced approximations of stochastic systems. *SIAM Journal of Matrix Analysis and Applications*, 11(1):42–68, 1990.

[2] T.E. Boult, R.J. Micheals, X. Gao, and M. Eckmann. Into the woods: visual surveillance of non-cooperative and camouflaged targets in complex outdoor settings. *Proceedings of the IEEE*, pages 1382–1402, October 2001.

[3] M. Brand. Incremental singular value decomposition of uncertain data with missing values. In *European Conference on Computer Vision*, page I: 707 ff., Copenhagen, Denmark, May 2002.

[4] Peter J. Brockwell and ichard A. Davis. *Introduction to Time Series and Forecasting*. Springer-Verlag, NY, 2002.

[5] Jr. C. R. Johnson. *Lectures on Adaptive Parameter Estimation*. Prentice-Hall, 1988.

[6] I. Cohen and G. Medioni. Detecting and tracking moving objects in video surveillance. In *IEEE International Conference on Computer Vision and Pattern Recognition*, pages II: 319–325, Ft. Collins, CO, June 1999.

[7] R.T. Collins, A.J. Lipton, H. Fujiyoshi, and T. Kanade. Algorithms for cooperative multi-sensor surveillance. *Proceedings of the IEEE*, 89(10):1456–1477, October 2001.

[8] M. Cover and J.A. Thomas. *Elements of Information Theory*. John Wiley and Sons, New York, 1994.

[9] F. de la Torre and M.J. Black. Robust principal component analysis for computer vision. In *IEEE International Conference on Computer Vision*, pages I: 362–369, Vancouver, Canada, July 2001.

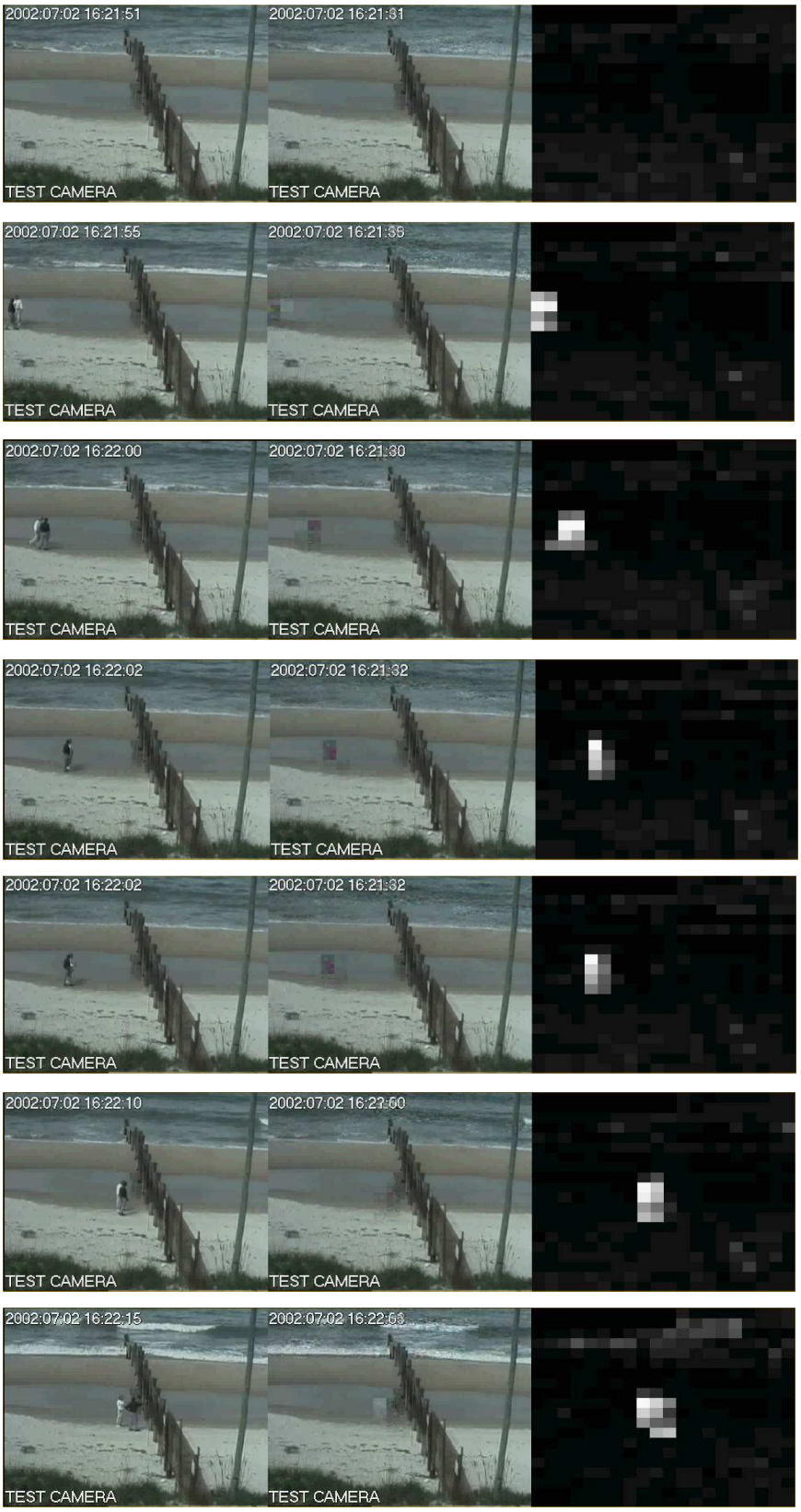


Fig. 11. Some results from a sequence. *Left*: Input signal, *Middle:* Predicted signal, *Right:* Block-wise response of the detection measures.

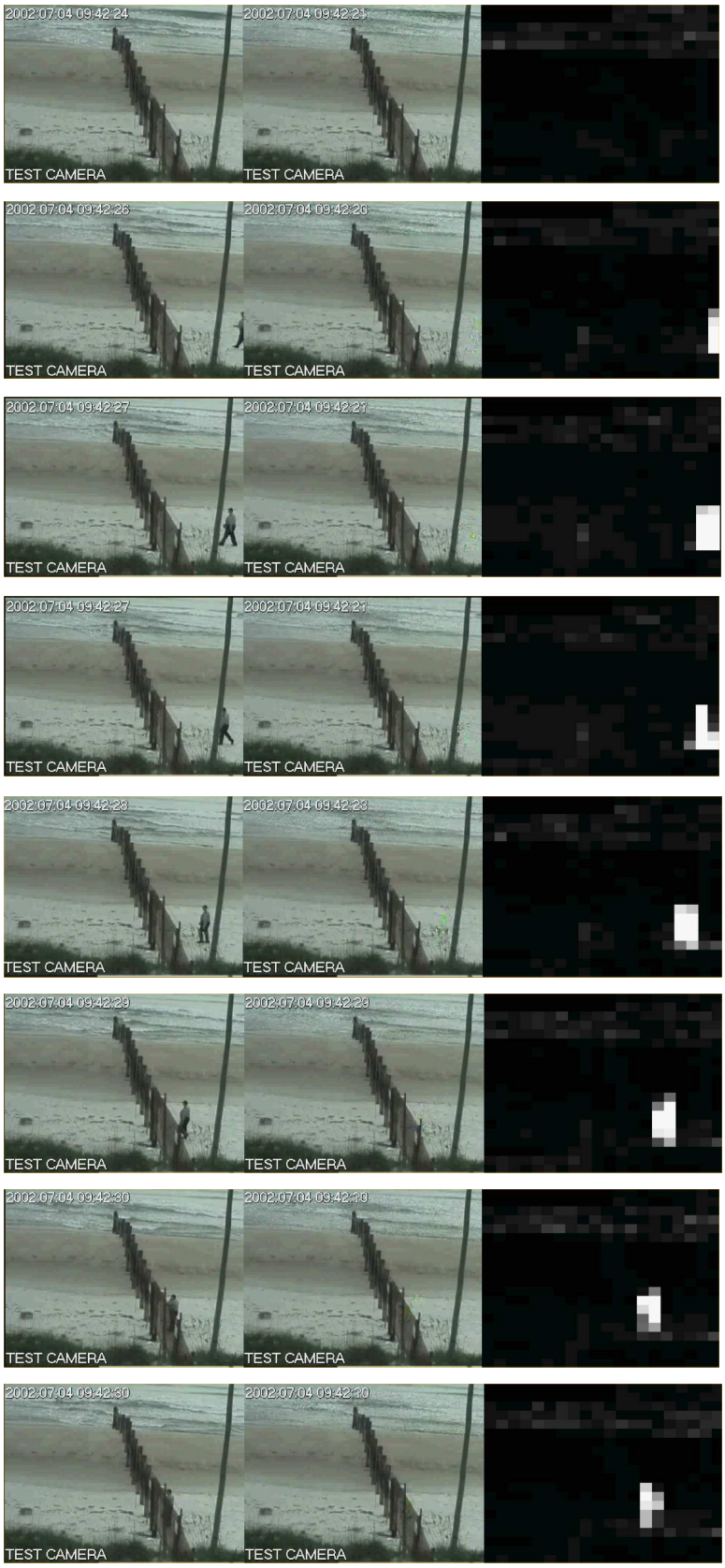


Fig. 12. Results from another sequence. Note that, in the later stages, the person is detected inspite of hiding behind the fence.

[10] G. Doretto, A. Chiuso, Y.N. Wu, and S. Soatto. Dynamic textures. *International Journal of Computer Vision*, 51(2):91–109, February 2003.

[11] A. Elgammal, R. Duraiswami, D. Harwood, and L.S. Davis. Background and foreground modeling using non-parametric kernel density estimation for visual surveillance. *Proceedings of the IEEE*, July 2002.

[12] A. Elgammal, D. Harwood, and L. Davis. Non-parametric model for background subtraction. In *European Conference on Computer Vision*, pages II:751–767, Dublin, Ireland, May 2000.

[13] C. Eveland, K. Konolige, and R.C. Bolles. Background modeling for segmentation of video-rate stereo sequences. In *IEEE International Conference on Computer Vision and Pattern Recognition*, pages 266–272, 1998.

[14] N. Friedman and S. Russell. Image segmentation in video sequences: A probabilistic approach. In *Thirteenth Conference on Uncertainty in Artificial Intelligence(UAI)*, August 1997.

[15] K. Fukunaga. *Introduction to Statistical Pattern Recognition*. Academic Press, Boston, second edition, 1990.

[16] X. Gao, T.E. Boult, F. Coetzee, and V. Ramesh. Error analysis of background adaption. In *IEEE International Conference on Computer Vision and Pattern Recognition*, pages I: 503–510, Hilton Head Island, SC, June 2000.

[17] G. H. Golub and C.F. Van Loan. *Matrix Computations*. John Hopkins University Press, 1996.

[18] M. Greiffenhagen, V. Ramesh, D. Comaniciu, and H. Niemann. Statistical modeling and performance characterization of a real-time dual camera surveillance system. In *IEEE International Conference on Computer Vision and Pattern Recognition*, pages II:335–342, Hilton Head, SC, 2000.

[19] M. Greiffenhagen, V. Ramesh, and H. Niemann. The systematic design and analysis cycle of a vision system: A case study in video surveillance. In *IEEE International Conference on Computer Vision and Pattern Recognition*, pages II:704–711, Hawaii, December 2001.

[20] W.E.L. Grimson, C. Stauffer, R. Romano, and L. Lee. Using adaptive tracking to classify and monitor activities in a site. In *IEEE International Conference on Computer Vision and Pattern Recognition*, Santa Barbara, CA, June 1998.

[21] I. Haritaoglu, D. Harwood, and L.S. Davis. W4: Real-time surveillance of people and their activities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):809–830, August 2000.

[22] M. Harville. A framework for high-level feedback to adaptive, per-pixel, mixture-of-gaussian background models. In *European Conference on Computer Vision*, page III: 543 ff., Copenhagen, Denmark, May 2002.

[23] T. Horprasert, D. Harwood, and L.S. Davis. A statistical approach for real-time robust background subtraction and shadow detection. In *Frame-Rate99*, 1999.

[24] Y.A. Ivanov and A.F. Bobick. Recognition of multi-agent interaction in video surveillance. In *IEEE International Conference on Computer Vision*, pages 169–176, Kerkyra, Greece, September 1999.

[25] S. Jabri, Z. Duric, H. Wechsler, and A. Rosenfeld. Detection and location of people in video images using adaptive fusion of color and edge information. In *International Conference on Pattern Recognition*, pages Vol IV: 627–630, 2000.

[26] O. Javed, K. Shafique, and M. Shah. A hierarchical approach to robust background subtraction using color and gradient information. In *MVC*, pages 22–27, Florida, December 2002.

[27] I.T. Jolliffe. *Principal Component Analysis*. Springler-Verlag, 1986.

[28] Klaus-Peter Karmann and Achim von Brandt. *V Cappellini (ed.), Time Varying Image Processing and Moving Object Recognition*, volume 2, chapter Moving Object Recognition Using an Adaptive Background Memory. Elsevier, Amsterdam, The Netherlands, 1990.

[29] Klaus-Peter Karmann, Achim von Brandt, and R. Gerl. *Signal Processing V: Theories and Application*, chapter Moving Object Segmentation based on adaptive reference images. Elsevier, Amsterdam, The Netherlands, 1990.

[30] Dieter Koller, Joseph Weber, and Jitendra Malik. Robust multiple car tracking with occlusion reasoning. In *European Conference on Computer Vision*, pages 189–196, Stockholm, Sweden, May 1994.

[31] L. Ljung. *System Identification - Theory for the User*. Prentice Hall: Englewood Cliffs, NJ, 1987.

[32] M.M. Loeve. *Probability Theory*. KVan Nostrand, Princeton, 1955.

[33] A. Mittal and L.S. Davis. M_2tracker: A multi-view approach to segmenting and tracking people in a cluttered scene. *International Journal of Computer Vision*, 51(3):189–203, February 2003.

[34] A. Mittal and D.P. Huttenlocher. Scene modeling for wide area surveillance and image synthesis. In *IEEE International Conference on Computer Vision and Pattern Recognition*, pages II: 160–167, Hilton Head, SC, 2000.

[35] A. Mittal and N. Paragios. Motion-based background subtraction using adaptive kernel density estimation. In *IEEE International Conference on Computer Vision and Pattern Recognition*, pages II: 302–309, 2004.

[36] B. Moghaddam and A.P. Pentland. Probabilistic visual learning for object detection. In *IEEE International Conference on Computer Vision*, pages 786–793, Boston, MA, 1995.

[37] B. Moghaddam and A.P. Pentland. Probabilistic visual learning for object representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):696–710, July 1997.

[38] N. Ohta. A statistical approach to background subtraction for surveillance systems. In *IEEE International Conference on Computer Vision*, pages II: 481–486, Vancouver, Canada, June 2001.

[39] E. Oja and J. Karhunen. On stochastic approximation of the eigenvectors and eigenvalues of the expectation of a random matrix. *Journal of Mathematical Analysis and Application*, 106:69–84, 1985.

[40] N.M. Oliver, B. Rosario, and A.P. Pentland. A bayesian computer vision system for modeling human interactions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):831–843, August 2000.

[41] P. Van Overshee and B. De Moor. N4sid: Subspace algorithms for the stochastic identification problem. *Automatica*, 30:75–93, 1994.

[42] N. Paragios and V. Ramesh. A mrf-based approach for real-time subway monitoring. In *IEEE International Conference on Computer Vision and Pattern Recognition*, pages I:1034–1040, Hawaii, December 2001.

[43] Robert Pless. Spatio-temporal background models for outdoor surveillance. *Journal on Applied Signal Processing*, 2005.

[44] Robert Pless, J. Larson, S. Siebers, and B. Westover. Evaluation of local models of dynamic backgrounds. In *IEEE International Conference on Computer Vision and Pattern Recognition*, Madison, WI, 2003.

[45] P. Remagnino, G.A. Jones, N. Paragios, and C.S. Regazzoni. *Video-Based Surveillance Systems: Computer Vision and Distributed Processing*. Kluwer Academic Publishers, 2001.

[46] C. Ridder, O. Munkelt, and H. Kirchner. Adaptive background estimation and foreground detection using kalman filtering. In *Proc. International Conference on recent Advances in Mechatronics, 193–199*, 1995.

[47] J. Rittscher, J. Kato, S. Joga, and A. Blake. A probabilistic background model for tracking. In *European Conference on Computer Vision*, pages II:336–350, 2000.

[48] D.W. Scott. *Multivariate Density Estimation*. Wiley-Interscience, 1992.

[49] M. Seki, H. Fujiwara, and K. Sumi. A robust background subtraction method for changing background. In *Workshop on Applications of Computer Vision*, pages 207–213, 2000.

[50] Makito Seki, Toshikazu Wada, H. Fujiwara, and K. Sumi. Background subtraction based on cooccurrence of image variations. In *IEEE International Conference on Computer Vision and Pattern Recognition*, Madison, WI, 2003.

[51] S. Soatto, G. Doretto, and Y.N. Wu. Dynamic textures. In *IEEE International Conference on Computer Vision*, pages II: 439–446, Vancouver, Canada, July 2001.

[52] C. Stauffer and W.E.L. Grimson. Adaptive background mixture models for real-time tracking. In *IEEE International Conference on Computer Vision and Pattern Recognition*, page II: 252, Ft. Collins, CO, June 1999.

[53] C. Stauffer and W.E.L. Grimson. Learning patterns of activity using real-time tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):747–757, August 2000.

[54] B. Stenger, V. Ramesh, N. Paragios, F. Coetzee, and J.M. Buhmann. Topology free hidden markov models: Application to background modeling. In *IEEE International Conference on Computer Vision*, pages I: 294–301, Vancouver, Canada, 2001.

[55] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers. Wallflower: Principles and practice of background maintenance. In *IEEE International Conference on Computer Vision*, pages 255–261, Kerkyra, Greece, September 1999.

[56] M.P. Wand and M.C. Jones. *Kernel Smoothing*. Chapman and Hall, 1995.

[57] D. Wang, T. Feng, H. Shum, and S. Ma. A novel probability model for background maintenance and subtraction. In *International Conference on Vision Interface*, page 109, 2002.

[58] J. Weng, Y. Zhang, and W.S. Hwang. Candid covariance-free incremental principal component analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(8), 2003.

[59] C.R. Wren, A. Azarbayejani, T.J. Darrell, and A.P. Pentland. Pfinder: Real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):780–785, July 1997.

[60] John Wright and Robert Pless. Analysis of persistent motion patterns using the 3d structure tensor. In *Proceedings of the IEEE Workshop on Motion and Video Computing*, pages II: 14–19, 2005.

[61] Y.H. Yang and M.D. Levine. The background primal sketch: An approach for tracking moving objects. *Machine Vision and Applications*, 5:17–34, 1992.

[62] Y. Zhang and J. Weng. Convergence analysis of complementary candid incremental principal component analysis. Technical report, Department of Computer Science and Engineering, Michigan State University, August 2001.

[63] J. Zhong and S. Sclaroff. Segmenting foreground objects from a dynamic, textured background via a robust kalman filter. In *IEEE International Conference on Computer Vision*, pages 44–50, Nice, France, October 2003.