

SPATIOTEMPORAL ALGORITHM FOR BACKGROUND SUBTRACTION

S. Derin Babacan, Thrasyvoulos N. Pappas

EECS Department, Northwestern University
2145 Sheridan Rd, Evanston, IL 60208, USA
sdb@northwestern.edu, pappas@ece.northwestern.edu

ABSTRACT

Background modeling and subtraction is a fundamental task in many computer vision and video processing applications. We present a novel probabilistic background modeling and subtraction method that exploits spatial and temporal dependencies between pixels. By using an initial clustering of the background scene, we model each pixel by a mixture of spatiotemporal Gaussian distributions, where each distribution represents locally a region in the neighborhood of the pixel. By extracting the local properties around each pixel, the proposed method obtains accurate models of dynamic backgrounds that are highly effective in detecting foreground objects. Experimental results for indoor and outdoor surveillance videos in comparison with other multimodal methods demonstrate the performance advantages of the proposed method.

Index Terms— background subtraction, object detection, probabilistic model, Bayesian formulation, video processing.

1. INTRODUCTION

Identification of foreground objects and background regions in a video is one of the fundamental tasks in computer vision and video processing, especially in applications like video surveillance, traffic analysis and monitoring, video coding, and tracking systems. The most widely used approach for this task is background subtraction, where a model for the semantically uninteresting stationary background regions is built and maintained throughout the process. Once a background model is created, the moving objects (the foreground) in the image sequence are extracted.

The most challenging part in background subtraction is building an adaptive model for the background, as in most scenes the background shows a spatiotemporally varying behavior that results from variations in illumination due to cloud cover or blocking of the light source in indoor videos, as well as from moving background objects such as tree leaves, rain, and snow. An efficient background model must incorporate the necessary invariance and adaptation to overcome these problems.

The most popular approaches for background subtraction are based on probabilistic models. In these methods, the prob-

ability distribution of the pixel values is estimated by a number of different techniques. A single Gaussian is used to model the statistics of a pixel in [1], where the mean and variance of the Gaussian is recursively updated over time. To model multimodal distributions, [2] uses a mixture of K Gaussians, where each Gaussian is classified as foreground or background distribution depending on the frequency of occurrence. This has achieved great success because of its high capability of handling multimodal backgrounds and adapting to varying scene properties. In [3], the authors proposed a nonparametric kernel density estimation (KDE) method. The background pdf is estimated using a smoothed histogram of N (typically 50-100) recent pixel intensities. Adaptation is done simply by updating the histogram values with new pixel intensities. KDE is more suitable for modeling a wide range of pdfs and has been quite successful despite of its relatively high computational load.

All of the techniques described so far model the background independently for each pixel. However, there is a substantial correlation between neighboring pixels. Many of the above-mentioned techniques have been augmented to take advantage of this correlation, by either modifying the background model or by introducing a postprocessing step. However, there is a need for a comprehensive approach that takes full advantage of the spatial information. Such an approach was initially proposed in [4]. Here we extend that approach, which used spatial MRF constraints and spatiotemporal adaptation of the region intensity functions, and investigate the necessity of spatial and temporal constraints, as well as their effect on computational complexity. What we have proposed so far, is the spatiotemporal approach proposed by Hinds and Pappas in [5]. Here we use a new computationally efficient way to calculate the region intensity functions, and also investigate the necessity of spatial and temporal constraints. As we will see below, we also introduce an additional region type to model the foreground objects.

This paper proposes a new method for background modeling that relies on segmentation to incorporate a better understanding of the background. Starting from a Bayesian formulation, we model the pixel processes in the context of their neighbors. The proposed algorithm provides an efficient background subtraction by preserving multimodality both spa-

tially and temporally.

This paper is organized as follows: Section 2 describes the modeling of the video based on a Bayesian formulation. The algorithm based on this model is also presented in this section. In Section 3 we show the performance of our algorithm in comparison with the existing methods. We provide our conclusions in Section 4.

2. MODEL

A video can be considered as a three dimensional volume \mathbf{y} of arbitrary shaped objects in space-time. Each video frame \mathbf{y}_t is a two dimensional slice in this volume taken at time t . Spatial coordinates are indexed with s , so a pixel intensity on an video frame \mathbf{y}_t is denoted as $y_{s,t}$. A clustering of the video sequence into spatiotemporal regions is denoted by \mathbf{x} , where \mathbf{x}_t is a segmentation of the frame \mathbf{y}_t , and $x_{s,t}$ denotes the labeling of the pixel at (s, t) ($x_{s,t} \in \{1, 2, \dots, K\}$). The parameter K is the number of clusters in the video. In this work, we use $K = 4$ clusters. Generally, values $3 \leq K \leq 5$ give similar results.

The probability of a clustering \mathbf{x} can be calculated using the *maximum a posteriori* (MAP) estimation approach. Given the observed video sequence \mathbf{y} , the *a posteriori* probability density function $p(\mathbf{x}|\mathbf{y})$. can be expressed by Bayes' theorem as follows:

$$p(\mathbf{x}|\mathbf{y}) \propto p(\mathbf{y}|\mathbf{x})p(\mathbf{x}), \quad (1)$$

where $p(\mathbf{x})$ is the *a priori* density of the region distribution and $p(\mathbf{y}|\mathbf{x})$ is the density of the observed video sequence given the distribution of the regions. We model the clustering process \mathbf{x} by a three dimensional Gibbs random field (GRF), which satisfies the Markovian property [6], i.e., the probability density of the a pixel at (s, t) can be completely characterized by its neighborhood $N_{s,t}$, that is,

$$p(x_{s,t}|x_{q,r}, \text{all } (q, r) \neq (s, t)) = p(x_{s,t}|x_{q,r}, (q, r) \in N_{s,t}). \quad (2)$$

By the Hammersley-Clifford theorem [6], $p(\mathbf{x})$ has the form of the Gibbs density:

$$p(\mathbf{x}) = \frac{1}{Z} \exp \left(\sum_C V_C(\mathbf{x}) \right), \quad (3)$$

where Z is a normalizing constant, $V_C(\mathbf{x})$ are the clique potentials, and the summation is over all cliques C . The clique potentials V_C depend only on the pixels that belong to clique C . A *clique* C is a subset of the neighborhood system defined on the Cartesian grid, where every pair of distinct pixels in C are neighbors.

We assume that the only nonzero potentials the ones that correspond to two-point cliques. The cliques are either spatial or temporal, so that the potential functions can be expressed as

$$V_S(\mathbf{x}) = \begin{cases} -\beta_S, & \text{if } x_{s,t} = x_{q,t} \text{ and } (s, t), (q, t) \in S \\ +\beta_S & \text{if } x_{s,t} \neq x_{q,t} \text{ and } (s, t), (q, t) \in S \end{cases} \quad (4)$$

$$V_T(\mathbf{x}) = \begin{cases} -\beta_T, & \text{if } x_{s,t} = x_{s,r} \text{ and } (s, t), (s, r) \in T \\ +\beta_T & \text{if } x_{s,t} \neq x_{s,r} \text{ and } (s, t), (s, r) \in T \end{cases} \quad (5)$$

where Z is a normalizing constant, $V_C(\mathbf{x})$ are the clique potentials, and the summation is over all cliques C . The clique potentials V_C depend only on the pixels that belong to clique C . The parameters β_S and β_T are positive so two neighboring pixels are more likely to belong to the same cluster than different clusters. Note that using two different parameters for temporal and spatial clique potentials, one can control the interaction between pixels within a single frame as well as across frames.

We model the objects as having uniform or slowly varying intensity in a small neighborhood so that the only discontinuities occur at the volume boundaries, that is, a cluster can be accurately modeled locally by a unimodal distribution, e.g. by a Gaussian. Thus, given a clustering \mathbf{x} , the conditional density of a pixel intensity in cluster i in a neighborhood of (s, t) can be modeled as

$$\hat{p}(y_{(s,t)}|x = i) = \eta(y_{s,t}; \mu_{i,s,t}, \Sigma_{i,s,t}) \quad (6)$$

Summing over all possible clusters i in the neighborhood $\mathcal{N}_{(s,t)}$ of (s, t) we achieve a spatiotemporal mixture of Gaussians model:

$$P(y_{s,t}|x_{p,q} \in \mathcal{N}_{(s,t)}) = \frac{1}{K} \sum_{i=1}^K \eta(y_{s,t}; \mu_{i,s,t}, \Sigma_{i,s,t}) \quad (7)$$

It is interesting to note that the model presented in [2] is a special case of this formulation where the neighborhood is taken as a single pixel, or the clique potentials β_T and β_S are set to zero (no spatial or temporal constraints), which is consistent with their approximation that all pixels are independent. However, this approximation is obviously very weak, and our Bayesian formulation generalizes to the concept of dependent pixel processes following the Markovian property.

The combined conditional probability in Eq. 1 can now be written as:

$$p(x_{s,t}|y_{s,t}) \propto \left[\frac{1}{K} \sum_{i=1}^K \eta(y_{s,t}; \mu_{i,s,t}, \Sigma_{i,s,t}) \right] \exp \left[\sum_C V_C(x_{s,t}) \right] \quad (8)$$

This formulation can be used to detect the foreground regions as follows: The first frame of the video is clustered using ACA [7] with K clusters, which follows the above formulation and takes approximately 1-2 seconds, depending on frame size. The models for pixel processes are generated using these K background clusters, and no foreground model

is generated. After this initialization phase, each consecutive frame can be clustered into $K + 1$ regions where the pixels in the cluster $K + 1$ belong to the foreground regions. Conceptually, a foreground cluster is not different than the K background clusters, and it is modeled locally as a single Gaussian like the other clusters. It is important to note that the single Gaussian model is very accurate in a very small neighborhood (we use a square neighborhood of 5 by 5 pixels). Using this local model even foreground objects consisting of very different textures can be detected.

As performance is a common important issue in background subtraction algorithms, the solution of Eq. (8) using common methods (simulated annealing, iterated conditional modes, etc.) (see, for example, [5]) is impractical. We propose two efficient approximate solutions: In our first solution, spatial and temporal constraints are eliminated ($\beta_T = \beta_S = 0$). In the consecutive frames, the foreground pixels are determined as the pixels having a low probability of belonging to the local models, specified solely by the spatiotemporal mixture of Gaussians, so that the probability density in Eq. (8) reduces to Eq. (7).

Alternatively, if the continuity is desired, the dependency constraints can be included and the probability can be calculated using both terms in Eq. (8). A causal ICM [8] approach can be utilized in the case of temporal continuity only, while for spatially continuous cases the algorithm needs a number of iterations. Considering the importance of computational constraints in background subtraction, we limit the number of iterations to 2 for spatial smoothing, so a real-time performance is achievable. Generally the algorithm accuracy is affected with the spatial smoothness term, and the resulting clustering is much smoother. However, for applications where the performance is critical, this regularization term can be dropped, which results in a one-pass algorithm.

To track the changes in dynamic scenes, this temporal model has to be adaptive. In our model, adaptation in each image plane \mathbf{y}_t can be done by reestimating the distribution parameters using the segmentation data \mathbf{x}_t . However, errors in the segmentation can lead to estimation problems and therefore to propagation of segmentation errors. Prefiltered data could be beneficial to decrease the effect of outliers. Since more weight has to be given to the present data than past for fast adaptation, we use an autoregressive model to update the distribution parameters $(\mu_{i,s,t}, \sigma_{i,s,t})$, that is,

$$\mu_{s,t}^i = (1 - \alpha)\mu_{i,s,t-1} + \alpha\hat{\mu}_{i,s,t} \quad (9)$$

$$\Sigma_{i,s,t} = (1 - \alpha)\Sigma_{i,s,t-1} + \alpha(\hat{\mu}_{i,s,t} - \mu_{i,s,t})^T(\hat{\mu}_{i,s,t} - \mu_{i,s,t}), \quad (10)$$

where α is the learning rate with $0 \leq \alpha \leq 1$, and $\hat{\mu}_{i,s,t}$ is the local intensity mean estimate calculated for region i using \mathbf{x}_t within the neighborhood defined by a square window of a

Control variables: $K, \alpha, W, \beta_S, \beta_T$

Initialization: Obtain initial segmentation x using ACA

Calculate local statistics $\mu_{i,s,t}, \Sigma_{i,s,t}$

Initialize K Gaussian pdf's G_i using $\mu_{i,s,t}, \Sigma_{i,s,t}$

while new data \mathbf{y}_t **do**

//Obtain new pixel label i

$i = \operatorname{argmax}_i \frac{|y_{s,t} - \mu_{i,s,t}|}{\sigma_{i,s,t}} \exp[\sum_C V_C(x_{s,t})]$

//Update

Calculate local statistics $\hat{\mu}_{i,s,t}$ using x_t

Apply Eqs. (9)-(10) to G_i

//(Optional) Spatial Smoothing

Apply ICM to \mathbf{x}_t until convergence or by a fixed number of iterations

end while

Fig. 1. Proposed Algorithm

side length W . This updating can be interpreted as an autoregressive filter with the current mean estimates as the input. The learning parameter α determines the rate of adaptation to the new pixel intensities. As opposed to [2], the sensitivity of the parameter α is very low for two reasons: The new data is already clustered among the Gaussians, and the smoothed $\hat{\mu}_{i,s,t}$ is used instead of the raw pixel intensities $y_{s,t}$. Thus, this parameter requires very little, if not, adjusting to specific videos where [2] requires careful tuning.

The overall model and the algorithm are summarized in Fig. 1.

3. EXPERIMENTAL RESULTS

In this section we show the performance of the proposed background subtraction algorithm and compare it with KDE [3] and MoG [2] methods.

A frame of an outdoor surveillance sequence is shown in Fig. 2(a), where the static camera is placed at a long distance so the objects have small appearances (as small as 5 pixels tall). This is a significant challenge for the proposed approach since the segmentation has to achieve high accuracy. The motion of trees and changing shadows present similar challenges. The pixel labels in frame number 260 are shown in Fig. 2(b) and the detected foreground objects are shown in Fig. 2(c). The advantage of the proposed algorithm is that it achieves a high accuracy at foreground detection while maintaining a low noise level, which is generally hard to achieve for conventional background subtraction algorithms that utilize independent pixel processes.

Second, a frame of a typical low-quality, compressed surveillance video is shown in Fig 3(a), where the resolution is low, blocking and artificial lighting artifacts are present. The result of MoG [2] is shown in Fig. 3(c), where maximum number of Gaussians per pixel set to 4. The result of KDE [3] is shown in Fig. 3(b). The KDE algorithm is run by setting the number of

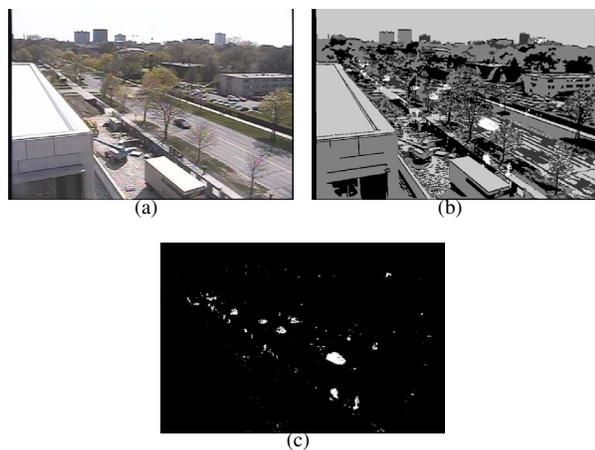


Fig. 2. Detection results on an outdoor surveillance video. (a) Original Image. (b) Pixel labels (c) Detected foreground.

history pixels to 100. The high noise level in this result is due to the short initialization time, which is a major drawback of KDE. In contrast, the proposed algorithm achieves much better results with a very short initialization time; as shown in Fig. 3(d), it clearly outperforms both of the methods. The most significant property to note is the very low noise level which is mainly due to temporal constraints.

Our method also compares favorably in terms of computational complexity and memory requirements. KDE method requires saving 50-100 frames in the memory and MoG method holds 3-5 distributions/pixel in the memory and both methods require a substantial amount of computation. Our approach requires storing $K + 1$ distributions per pixel and the clustering information in the memory, and the processing speed achieves real-time performance 10-30 fps (using temporal constraints only), depending on the window size used to calculate the local statistics). When spatial constraints are included for smoother results, a performance of 5-10 fps can be achieved. The initialization phase requires only 1-2 seconds, which is acceptable in most applications.

4. CONCLUSIONS

In this paper, we presented a novel background subtraction algorithm based on a Bayesian formulation that generalizes conventional algorithms. The pixel processes are modeled as a Gibbs-Markov random field which helps to exploit spatiotemporal dependencies between pixels in the video to help reduce the detection noise while maintaining the desired accuracy. The labeling of pixels capture the general appearance of the scene and greatly enhance the final result. The model is updated by a simple yet efficient manner so that the algorithm captures the changes in the observed scene. Although the model is more complicated than the previous parametric mixture of Gaussians methods, the algorithm reaches real time

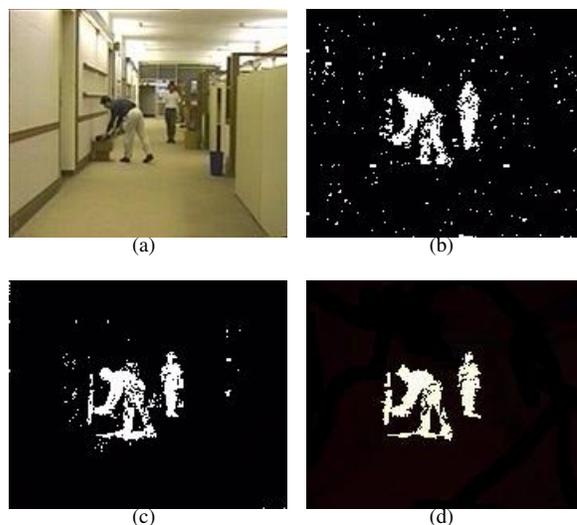


Fig. 3. Detection results on an indoor surveillance video. (a) Original Image. Detection results using (b) Nonparametric (KDE) model (c) Mixture of Gaussians model (d) Proposed approach.

performance. Experimental results show that our algorithm gives superior performance compared to existing methods.

5. REFERENCES

- [1] C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland, "Pfinder: Real-time tracking of the human body," *IEEE Tr: PAMI.*, vol. 19, no. 7, pp. 780–785, 1997.
- [2] C. Stauffer and W. E. L. Grimson, "Learning patterns of activity using real-time tracking," *IEEE Tr: PAMI.*, vol. 22, no. 8, pp. 747–757, 2000.
- [3] A. M. Elgammal, D. Harwood, and L. S. Davis, "Non-parametric model for background subtraction," in *ECCV '00*, London, UK, 2000, pp. 751–767.
- [4] S. D. Babacan and T. N. Pappas, "Spatiotemporal algorithm for joint video segmentation and foreground detection," *Proc. EUSIPCO*, Florence, Italy, September 2006.
- [5] R.O. Hinds and T.N. Pappas, "An adaptive clustering algorithm for segmentation of video sequences," *ICASSP*, May 1995, vol. 4, pp. 2427–2430.
- [6] J. M. Hammersley and P. Clifford, "Markov field on finite graphs and lattices," 1971.
- [7] T. N. Pappas, "An adaptive algorithm for image segmentation," *IEEE Trans. Signal Process.*, vol. 40(4), pp. 901–913, april 1992.
- [8] J. Besag, "Spatial interaction and the statistical analysis of lattice systems," *J. Royal Statist. Soc. B*, vol. 26, no. 2, pp. 192–236, 1974.