SEPARATING BACKGROUND AND FOREGROUND IN VIDEO BASED ON A NONPARAMETRIC BAYESIAN MODEL

Xinghao Ding and Lawrence Carin

Electrical and Computer Engineering Department
Duke University, Durham, NC, USA

ABSTRACT
Separating background and foreground in video is a fundamental problem in computer vision. We present a Bayesian hierarchical model to address this challenge, and apply it to video with dynamic scenes. The model uses a nonparametric prior, a beta-bernoulli process, for both the background and foreground representation. Additionally, the model uses neighborhood information of each pixel to encourage group clustering of the foreground. A collapsed Gibbs sampler is used for efficient posterior inference. Experimental results show competitive performance of the proposed model.

Index Terms—Background subtraction, dynamic scenes, nonparametric Bayesian hierarchical model, beta-bernoulli process, group sparsity

1. INTRODUCTION
In computer vision applications, such as video surveillance, motion analysis, human-machine interaction, detection of moving objects from an image sequence is important for target tracking, activity recognition, and behavior understanding. Background subtraction is a popular approach for object extraction, and there exist many methods for this problem. To accommodate multi-modal characteristics of the background, a mixture of Gaussians (MoG) modeling technique was proposed in [1, 2]. In this approach the color evolution of background pixels are modeled by a MoG. As a parametric method, the number of Gaussian needs to be specified in the MoG. To deal with this limitation, Gammal et al. [3] proposed a nonparametric method via kernel density estimation (KDE). This approach can deal with multimodality characteristics in background pixel distributions, without specifying the number of modes in the background. In [4] Tavakkoli proposed an adaptive kernel density estimation (AKDE) method to improve the performance of the KDE. Toyama applied a Wiener filter [5] to model each pixel. This filter-based methods can adapt to changes in illumination, but perform not good in complex dynamic scenes. In [6] Sheikh and Shah proposed a Bayesian framework for object detection in dynamic scenes, using spatial and temporal constraints. In [7] the background values at each pixel are quantized into codebooks, which represent a compressed form of background model for a long image sequence. In [8] Gallego combined a simple pixel-wise model for the background with a general-purpose region-based model for the foreground, performing better in situations where the foreground objects have similar colors to those of the background. Assuming the moving objects and the stationary background in an image are independent, Tsai [9] proposed a background-subtraction method based on independent component analysis. In [10] Jing modeled the dynamic texture by an Autoregressive Moving Average Model (ARMA), and proposed an algorithm to segment foreground objects from a dynamic textured background via a robust Kalman filter.

Unlike the above models that use the variability of pixel intensity at a particular location in the image, incorporating spatial and temporal constraints to identify the background and foreground, Huang considered background-foreground separation as a compressive-sensing inverse problem; he proposed a new greedy sparse recovery algorithm using both the prior of the background and foreground (AdaDGS) [11]. The success of the AdaDGS algorithm is based on two assumptions. One is the sparse representation-constancy assumption, which means that a new video background should be sparsely represented by the preceding backgrounds. The second is that the foreground is group-clustering sparse. The AdaDGS yields state-of-the-art results. But as a parametric method, there are user-defined tuning parameters that need be set, for example the range of the sparsity number and the termination condition. Similar to the AdaDGS, Ding et al. proposed a background-foreground separation method based on Bayesian robust principal component analysis (BRPCA) [12], and achieved competitive performance. BRPCA assumes the background resides in a low-rank subspace, and does not need to assume that background subtraction has already been performed on the first $t$ frames (as required by AdaDGS). But if the low-rank constraint for the background is inappropriate, BRPCA may not perform well.

In this paper we extend the BRPCA algorithm to efficiently handle video with a dynamic background. We propose a nonparametric Bayesian hierarchical model to separate the foreground and background of the video, using similar assumptions to that of AdaDGS. A beta-Bernoulli prior
is employed to infer the appropriate previous backgrounds to represent the current background. The model uses the beta-Bernoulli prior as well as neighborhood information of pixels to infer the foreground. We present experimental results on dynamic scenes video.

The remainder of the paper is organized as follows. We present and discuss the model in Section 2, with inference discussed in Section 3. We present experimental results in Section 4 and provide conclusions in Section 5.

2. MODEL CONSTRUCTION

Let \( \{I_t\}_{t=1}^T \) be a collection of \( T \) frames of size \( m \times n \times 3 \) (color) from a video sequence, reshaped into \( P = 3mn \) dimensional vectors. Let \( \{B_t\}_{t=1}^T \) and \( \{F_t\}_{t=1}^T \) be the collection of backgrounds and foregrounds respectively (which will be inferred). We model every frame, \( I_t \), as a superposition of the background \( B_t \) and foreground \( F_t \), with additive noise/residual \( \epsilon_t \).

We assume that inference of the background has already been performed on the first \( \tau \) frames (of course, with this implemented sequentially as frames arrive). Each background \( B_t \) is modeled as a sparse, weighted combination of a dictionary matrix, \( D_t \in \mathbb{R}^{P \times \tau} \), composed of backgrounds from previous frames. Let binary vector \( b_t \in \{0,1\}^P \) denote which of the columns of \( D_t \) are used for representation of \( B_t \). The \( k \)th component of \( b_t \) is drawn \( b_{kt} \sim \text{Bernoulli}(\pi_{kt}) \); \( \pi_{kt} \sim \text{Beta}(a_0, b_0) \) represents the probability that \( b_{kt} \) is used to represent \( B_t \). The sparseness of the \( b_t \) is explicitly imposed through hyperparameters \( a_0 \) and \( b_0 \). However, \( b_t \) imposes that the coefficients of the dictionary expansion must be binary. To address this, we draw weights \( s_t \sim N(0, \gamma_s^{-1} I_T) \), and the dictionary weights are \( b_t \circ s_t \), where \( \circ \) represents the Hadamard (element-wise) multiplication of two vectors. The background is now approximated as \( B_t = D_t(b_t \circ s_t) \). Note that this imposition of sparseness is distinct from the widely used Laplace shrinkage prior [13], which imposes that many coefficients are small, but not necessarily exactly zero.

The initial foreground is modeled as \( \tilde{F} = z_t \circ w_t \), where binary vector \( z_t \in \{0,1\}^P \) denotes which pixels belong to the foreground. As described above the foreground is sparse, with non-zero values anticipated to be manifested in the form of contiguous groups. This implies that if a pixel belongs to the foreground, pixels in its neighborhood are also in the foreground with high probability. To utilize this property, the probability \( \tilde{\omega}_{pt} \) that \( z_t \) is foreground is defined by its neighborhood average as \( \tilde{\omega}_{pt} = (\sum_{i \in N_p} \omega_{it})/|N_p| \), where \( N_p \) defines the set of neighborhood pixels for pixel \( p \) and \( |N_p| \) is the number of pixels in \( N_p \). The \( \tilde{\omega}_{pt} \) is drawn as \( \tilde{\omega}_{pt} \sim \text{Beta}(c_0, d_0) \), with the sparseness of \( z_t \) is imposed through hyperparameters \( c_0 \) and \( d_0 \). The weight of \( z_{pt} \) is drawn from \( w_{pt} \sim N(0, \gamma_w^{-1}) \). The \( s_t, \epsilon_t \) and \( w_t \) are drawn from Gaussian distributions that have respective precisions \( \gamma_s, \gamma_{\epsilon_t} \) and \( \gamma_w \), and conjugate noninformative gamma distributions are imposed on them (discussed further below).

The hierarchical form of the model described above is now summarized as

\[
I_t = B_t + F_t + \epsilon_t, \quad t = 1, \ldots, T
\]

\[
B_t = D_t(b_t \circ s_t)
\]

\[
D_t = [B_{t-\tau}, \ldots, B_{t-1}]
\]

\[
s_{kt} \sim N(0, \gamma_s^{-1}), \gamma_s \sim \text{Gamma}(c_0, f_0), \quad k = 1, \ldots, \tau
\]

\[
b_{kt} \sim \text{Bernoulli}(\pi_{kt}), \pi_{kt} \sim \text{Beta}(a_0, b_0)
\]

\[
F_t = \tilde{F} \circ \text{sign}([\tilde{F}_t - Th], Th = \lambda|I_t|
\]

\[
\tilde{F} = z_t \circ w_t
\]

\[
w_{pt} \sim N(0, \gamma_w^{-1}), \gamma_w \sim \text{Gamma}(g_0, h_0), \quad p = 1, \ldots, P
\]

\[
z_{pt} \sim \text{Bernoulli}(\tilde{\omega}_{pt}), \tilde{\omega}_{pt} = (\sum_{i \in N_p} \omega_{it})/|N_p|
\]

\[
\omega_{pt} \sim \text{Beta}(c_0, d_0)
\]

\[
\epsilon_t \sim N(0, \gamma_{\epsilon_t}^{-1} I_T), \gamma_{\epsilon_t} \sim \text{Gamma}(j_0, k_0)
\]

3. MODEL INFERENCE

Approximate inference of the model posterior is computed via Markov Chain Monte Carlo (MCMC) computations, employing collapsed Gibbs sampling. At each MCMC iteration, the samples are drawn from the following conditional distributions:

1) Sampling \( b_t \) and \( s_t \):

\[
p(b_{kt}, s_{kt}|-) \propto \text{Bernoulli}(b_{kt}; \pi_{kt})N(s_{kt}; 0, \gamma_s^{-1})N(\tilde{I}_t; b_{kt}s_{kt}d_{sk}, \gamma_{s_t}^{-1} I_T)
\]

with \( \tilde{I}_t = I_t - D_t(b_t \circ s_t) - w_t \circ z_t + b_{kt}s_{kt}d_{sk} \). Here \( d_{sk} \) is the \( k \)th column of \( D_t \). Thus we can sample \( b_t \) and \( s_t \) from

\[
p(b_{kt}|-) = \int p(b_{kt}, s_{kt}|-)d_{sk} = \text{Bernoulli}(b_{kt}; \tilde{\pi}_{kt})
\]

\[
p(s_{kt}|-) = (1 - b_{kt})N(s_{kt}; 0, \gamma_s^{-1}) + b_{kt}N(s_{kt}; \mu_{sk_t}, \Sigma_{sk_t})
\]

where,

\[
\ln \frac{\tilde{\pi}_{kt}}{1 - \tilde{\pi}_{kt}} = \ln \frac{\pi_{kt}}{1 - \pi_{kt}} + \frac{1}{2} ln \Sigma_{sk_t} + \frac{\mu_{sk_t}^2}{2 \Sigma_{sk_t}} + \frac{1}{2} |n| \gamma_s
\]

\[
\Sigma_{sk_t} = (\gamma_{s_t} + \gamma_{s_t}d_{sk}d_{sk}^{-1})^{-1}, \mu_{sk_t} = \gamma_{s_t} \Sigma_{sk_t} d_{sk}^{-1} \tilde{I}_t
\]

2) Sampling \( \pi_{kt} \):

\[
p(\pi_{kt}|-) = \text{Beta}(a_0 + 1(s_{kt} \neq 0), b_0 + 1(s_{kt} = 0))
\]
where \(1(x)\) denotes an indicator function such that \(1(x) = 1\) if \(x\) is true and 0 otherwise.

3) Sampling \(\gamma_{st}\):

\[
p(\gamma_{st}|-) = \text{Gamma}(c_0 + \tau/2, f_0 + \|s_t\|^2/2) \tag{6}
\]

4) Sampling \(z_t\) and \(w_t\):

\[
p(z_{pt}, w_{pt}|-) \propto \text{Bernoulli}(z_{pt}; \omega_{pt}) N(w_{pt}; 0, \gamma_{w_1}^{-1}) N(\hat{I}_{pt}; z_{pt} w_{pt}, \gamma_{e_1}^{-1})
\]

where, 

\[
l_n \omega_{pt} = l_n \omega_{pt} + \frac{1}{2} ln(\Sigma_{w_{pt}+1}) + \frac{1}{2} ln(\gamma_{w_1})
\]

\[
\Sigma_{w_{pt}} = (\gamma_{w_1} + \gamma_{e_1})^{-1}, \mu_{w_{pt}} = \gamma_{e_1}\Sigma_{w_{pt}} \hat{I}_{pt}
\]

5) Sampling \(\gamma_{w_1}\):

\[
p(\gamma_{w_1}|-) = \text{Gamma}(g_0 + P/2, h_0 + \|w_t\|^2/2) \tag{9}
\]

6) Sampling \(\omega_t\):

Denote \(\omega_{pt}^{-1} = \sum_{i \in N_p, i \neq j} \omega_{it} / |N_p| = \omega_{pt} - \omega_{jt} / |N_p|\), the posterior of \(\omega_{jt}\) can be expressed as,

\[
p(\omega_{jt}|-) \propto \text{Beta}(\omega_{jt}; c_0, d_0) \prod_{p; \{j \in N_p\}} \text{Bernoulli}(z_{pt}; \omega_{jt} / |N_p| + \omega_{pt}^{-1}) \tag{11}
\]

which cannot be directly sampled. We can utilize Metropolis-Hastings algorithm to sample it, but this may be computationally expensive. We derived an approximately method:

\[
p(\omega_{jt}|-) \approx \text{Beta}(c_0 + \sum_{p; \{j \in N_p\}} z_{pt}, d_0 + |N_p| - \sum_{p; \{j \in N_p\}} z_{pt}) \tag{12}
\]

with this found to work well in practice.

7) Sampling \(\gamma_{e_1}\):

\[
p(\gamma_{e_1}|-) = \text{Gamma}(j_0 + P/2, k_0 + \|I_t - B_t - F_t\|^2/2) \tag{13}
\]

4. EXPERIMENTS

In order to examine the effectiveness of the proposed method for dynamic scenes, we conduct experiments using two video sequences. The first is from [10], which involves waves on water. The second is from [14], where part of the scene involves dynamic waves. The hyperparameters within the gamma distributions were set as \(c_0 = f_0 = g_0 = h_0 = j_0 = k_0 = 10^{-6}\), as is typically done in models of this type [15]. Similar to our previous work [12], the hyperparameters of the beta distribution are set as \(a_0 = 0.1, b_0 = 0.9, c_0 = 0.1P\) and \(d_0 = 0.9P\), which is proportional to the number of the coefficients. These hyperparameters are the same for all the following experiments. The only parameter that needs to be set differently for each experiment is the proportionality factor \(\lambda\) of threshold \(Th\). The proportionality factor is set as \(\lambda = 0.1\) for the first experiment, and \(\lambda = 0.25\) for the second experiment.

We only show the performance of the proposed method, in Figs. 1 and 2, due to space limitations. For the AdaDGS, MoG and KDE results the reader is referred to [11] or http://paul.rutgers.edu/~jzhuang/R_DGS_BG.htm. Note that all results with our proposed method are not post-processed with image-processing operations. Compared to the traditional pixel based MoG and KDE methods, the proposed method produces clean estimates of the background frames. Compared to AdaDGS, one observes that the proposed model has competitive performance, without many parameters to be set (as indicated above, the hyperparameter settings above are “standard” in hierarchical Bayesian analysis, and no tuning has been performed). The experimental results show that the proposed method can handle highly dynamic scenes well.

![Fig. 1. Results on the [10] dataset. Left column are the original frames; middle column are the estimated foregrounds; right column are the estimated backgrounds](image-url)
Fig. 2. Results on the [14] dataset. Left column are the original frames; middle column are the foreground; right column are the background.

5. CONCLUSION

We have presented a Bayesian nonparametric model for background and foreground estimation in dynamic scenes (video). The model assumes that every background can be sparsely represented by preceding backgrounds, and the foreground is assumed sparse, with non-zero components manifested in contiguous groups. The model uses the beta-Bernoulli prior for the sparse coefficient learning. Besides the beta-Bernoulli prior, the neighborhood information is also used to infer the foreground, which encourages contiguous sparseness properties. Experiments on several videos show an advantage of our model compared with other algorithms.

6. REFERENCES


