Background Subtraction via Generalized Fused Lasso Foreground Modeling

Bo Xin Yuan Tian Yizhou Wang Wen Gao
Nat'l Engineering Laboratory for Video Technology
Cooperative Medianet Innovation Center
Key Laboratory of Machine Perception (MoE)
Sch'l of EECS, Peking University, Beijing, 100871, China

Abstract

Background Subtraction (BS) is one of the key steps in video analysis. Many background models have been proposed and achieved promising performance on public data sets. However, due to challenges such as illumination change, dynamic background etc. the resulted foreground segmentation often consists of holes as well as background noise. In this regard, we consider generalized fused lasso regularization to quest for intact structured foregrounds. Together with certain assumptions about the background, such as the low-rank assumption or the sparse-composition assumption (depending on whether pure background frames are provided), we formulate BS as a matrix decomposition problem using regularization terms for both the foreground and background matrices. Moreover, under the proposed formulation, the two generally distinctive background assumptions can be solved in a unified manner. The optimization was carried out via applying the augmented Lagrange multiplier (ALM) method in such a way that a fast parametric-flow algorithm is used for updating the foreground matrix. Experimental results on several popular BS data sets demonstrate the advantage of the proposed model compared to state-of-the-arts.

1. Introduction

Background Subtraction (BS) is often regarded as a key step in video analysis. In general, it is challenging to devise a good background model and some well-known challenges include: illumination changes, dynamic background, bootstrapping, camouflage etc. To meet these challenges, many works on BS have been proposed. In the following, we discuss some related topics.

Models of BS. From the representation perspective, most existing works could be categorized into two classes: pixel-wise modeling and frame-wise modeling. In the first category, representative methods model pixel-wise statistics of the background using mixture of Gaussian models (MoG) [21, 27, 12] and neural network models [9, 17] etc. Non-parametric models are also proposed for improved efficiency [1]. The pixel-wise models are prone to resulting in fragmentary foregrounds, i.e. there are both "holes" in the foregrounds and false positive pixels from the background. Whereas, the models of the second category, i.e. frame-wise models, usually achieve better performance by exploring structure information of the background. These works can be generally viewed as follow-ups of the celebrated eigen-background model proposed in [20], of which the key assumption is that when camera motion is small, the matrix consists of background frames is approximately low-rank [20, 7]. Hence, these models project video frames onto the subspace spanned by the eigen-vectors associated with the largest eigen-values of the matrix composed of all the frames of a video sequence. The recovered signal in the subspace is regarded as background and the residual is assumed to be foreground. Although structure information acquired by such holistic models helps to improve the integrity of the recovered background, such improvement can be limited in many situations due to the neglect of foreground structural prior at the same time. Thus deliberate post-processing steps are often needed e.g. in [5, 12, 19]. *Therefore, is there a way to quest for intact structured foregrounds, which in turn can benefit background estimation?*

Learning in BS. In some scenarios, when pure background frames are available, learning background models can be achieved in a supervised manner. We call this situation the *supervised model learning (SML)* case. Many pixel-wise models belong to this category and they learn/update the models from given background pixels. Whereas frame-wise models, due to their blind decomposition origin of the eigen-background, tend to neglect this piece of information.

In many other situations, foreground background coexist in each frame. We call this situation the *unsupervised model learning (UML)* case and it is more challenging than the SML case. In practice, however, pixel-wise models are learnt based on the frames ahead of the test frame, regardless of such an existence of foregrounds. Consequently, in the case of UML, the pixel-wise models are less robust compared to the frame-wise counterpart, since the latter can exploit the holistic structure prior of background frames without knowing explicit background labels.

Considering the two learning cases, *is there a framework that unifies both SML and UML situations of BS?*

The Proposed Model. To address the above two questions, we devise a BS model that explicitly models the cohesion structure of the foregrounds in addition to the background structural prior, and propose a unified framework that solves both UML and SML cases of the BS problem.

Notice that the foregrounds in a video sequence often correspond to meaningful objects such as people, cars, etc., therefore, the foreground pixels are usually both spatially connected as well as sparse if their sizes are relatively small w.r.t. the background scene. We realize such generic foreground structural priors by adopting an adaptive version of the generalized fused lasso (GFL) regularization in [25]. GFL can be viewed as a combination of the l_1 norm of both the variable values and their pairwise differences, i.e. the total variation (TV) penalty [13]. By further modeling the connection/fusion strength between pixel pairs according to their similarity, (which is a strong prior in semantic segmentation [4]), our foreground model can be considered as a flexible structural prior model without any pre-defined organization of the pixels. Specifically, we denote each frame as a vector and the sequence of frames as a matrix concatenating all the frame vectors. We assume that the observed matrix is a summation of a background matrix and a foreground matrix. Thus, by inducing a low-rank term of the background matrix and the GFL term for all foreground vectors, we formulate BS as a matrix decomposition problem. In this way, the proposed model exploits structure information from both the background and the foreground.

To harness the availability of pure background frames in the SML situation, we derive a special case of the proposed formulation. This is done by explicitly adding constraints such that part of the observed matrix equals to the given background matrix. We further assume that the unknown background vectors of the testing frames lie in the span of the given background matrix, which is itself a low-rank matrix. In this way, we show that the resulted optimization is equivalent to a sparse estimation problem.

From the perspective of optimization, the derived objective and constraints form a new problem. We propose an iterative algorithm by applying the augmented Lagrange multiplier (ALM) method, which alternatively updates either the background matrix or the foreground matrix. When updating the background, singular value thresholding (SVT) [6] is applied for UML and fast iterative softhreshing (FISTA) [2] is applied for SML. While updating the foreground, we solve the fused optimization with a fast parametric-flow algorithm [11]. The idea behind this alternation is that, simultaneous estimation of the foreground and the background can reinforce each other. Indeed, experiments show that the proposed model achieves better than state-of-the-art performance on several popular data sets including both natural and synthetic videos.

Related Works. In [7], the robust principle component analysis (RPCA) model was applied to solve the BS problem. From the standpoint of BS per se, RPCA can be viewed as an extension of the eigen-background model where explicit sparse assumption of the foregrounds are taken into account, but not the connectedness. Here we introduce a stronger foreground model. In [8] and [26], the group lasso (with overlap) regularization was applied to model the foregrounds, where the structure of foreground is assumed to be group sparse with predefined atomic group structures. These works reported improved performance over RPCA. However, in practice, experiments show that our model outperforms that of [26][1] on all of the tested sequences. This indicates that the adaptive GFL could be a more flexible foreground structural prior compared to group lasso. In particular, Figure 6 shows such a comparison.

Contributions. In summary, the contributions of this work are three folds. (1) We introduce an adaptive generalized fused lasso as a flexible structural prior to modeling foreground objects in the background subtraction problem. We show that the performance of BS can be much improved by exploiting the structure information of both the foreground and the background. (2) We propose an effective algorithm to optimize the new objective function, i.e. constrained rank minimization with GFL, by extending the method of augmented Lagrange multiplier. (3) The proposed solution to BS is a unified method which is able to solve both supervised and unsupervised learning cases depending on whether pure background frames are available, though they lead to different objectives.

2. Proposed Background Subtraction Method

2.1. Unsupervised Model Learning

We start by introducing our model for the unsupervised model learning problem, i.e. UML. Given a sequence of n video frames, each frame is denoted as $\mathbf{d}^{(i)} \in \mathbb{R}^p$, $i = 1, ..., n$. All data are concatenated into one matrix $\mathbf{D} \in \mathbb{R}^{p \times n}$, which is called the observation matrix. We assume that the observation matrix is the summation of a background matrix and a foreground matrix, i.e. $\mathbf{D} = \mathbf{B} + \mathbf{F}$, where $\mathbf{B}, \mathbf{F} \in \mathbb{R}^{p \times n}$ are the background matrix and the foreground matrix, respectively. Therefore, by assuming low-rank of $\mathbf{B}$ and structured sparsity of $\mathbf{F}$, we propose the

[1] The model in [8] applied group sparsity to a trajectory representation of videos, instead of pixels we considered here. Therefore in the experiments, we focus on comparing with [26], which applied various of group sparsity to pixel representation and it is more recent than [8].

following matrix decomposition objective,

$$\begin{aligned}&\min_{\mathbf{B},\mathbf{F}}\ \operatorname{rank}(\mathbf{B}) + \lambda\|\mathbf{F}\|_{gfl}\\ &\text{s.t.}\ \ \mathbf{D} = \mathbf{B} + \mathbf{F},\end{aligned} \tag{1}$$

where $\lambda \geq 0$ is a tuning parameter (controlling the relative contribution) and $\|\cdot\|_{gfl}$ is the generalized fused lasso regularization defined as

$$\|\mathbf{F}\|_{gfl} = \sum_{k=1}^{n}\{\|\mathbf{f}^{(k)}\|_1 + \rho \sum_{(i,j)\in\mathcal{N}} w_{ij}^{(k)}|f_i^{(k)} - f_j^{(k)}|\}, \tag{2}$$

where $\mathbf{f}^{(k)}$ is the kth foreground vector and $\mathcal{N}$ is the spatial neighborhood set, i.e. $(i,j) \in \mathcal{N}$ when pixel i and j are spatially connected. Due to the l_1 penalties on each pixel as well as each adjacent pair of pixels, solutions of $\mathbf{f}$s tend to be both sparse and spatially connected. Here w_{ij} are introduced to enhance the conventional GFL model [25] such that w_{ij} encode the strength of the fusion between neighboring pixels. In our model w_{ij} is defined as

$$w_{ij}^{(k)} = \exp\frac{-\|d_i^{(k)} - d_j^{(k)}\|_2^2}{2\sigma^2}, \tag{3}$$

where d is the pixel intensity. This definition of w_{ij} makes it an adaptive weight encouraging spatial cohesion according to the associated pixels' intensity in the observed images. To be specific, when we observe a large difference between two neighbouring pixels, there is a high probability that this pair of pixels belongs to different segments, therefore we decrease the fusion of this pair. $\sigma \geq 0$ is a tuning parameter empirically set. When $\sigma \to \infty$, all $w_{ij} = 1$, the model reduces to the conventional GFL [25], where the fused term encourages pure spatial cohesion regardless of the pixel differences. When $\sigma \to 0$, all $w_{ij} = 0$, the model reduces to the RPCA model [7], where the foreground pixels are only assumed to be sparse.

For ease of optimization, the convex nuclear/trace norm is often applied to relax the matrix rank. Thus in practice, the following surrogate is considered.

$$\begin{aligned}&\min_{\mathbf{B},\mathbf{F}} \|\mathbf{B}\|_* + \lambda\|\mathbf{F}\|_{gfl}\\ &\text{s.t.}\ \ \mathbf{D} = \mathbf{B} + \mathbf{F},\end{aligned} \tag{4}$$

where $\|\mathbf{B}\|_*$ is the nuclear norm of matrix $\mathbf{B}$, i.e. the sum of the singular values of $\mathbf{B}$.

Algorithm 1 ALM algorithm for Eq. (4).

```
Input: $\mathbf{D} \in \mathbb{R}^{p\times n}$, $\lambda > 0$.
Output: $\mathbf{B}, \mathbf{F} \in \mathbb{R}^{p\times n}$.
Initialization: Set $\mathbf{Y}_0 = \mathbf{0}$, $\mathbf{B}_0 = \mathbf{0}$, $\mathbf{F}_0 = \mathbf{0}$, $\mu_0 > 0$, $\beta > 1$ and $\mu_{\max}$.
while not converged do
    $\mathbf{B}_{k+1} = \operatorname{argmin}_{\mathbf{B}} L(\mathbf{B}, \mathbf{F}_k, \mathbf{Y}_k, \mu_k)$
    $\mathbf{F}_{k+1} = \operatorname{argmin}_{\mathbf{F}} L(\mathbf{B}_{k+1}, \mathbf{F}, \mathbf{Y}_k, \mu_k)$
    $\mathbf{Y}_{k+1} = \mathbf{Y}_k + \mu_k(\mathbf{D} - \mathbf{B}_{k+1} - \mathbf{F}_{k+1})$
    $\mu_{k+1} = \min\{\beta\mu_k, \mu_{\max}\}$
return $\mathbf{B}_k, \mathbf{F}_k$
```

2.2. Optimization via ALM

Eq (4) is a convex optimization problem. Off-the-shelf-solvers can be applied to solve it. However, when the dimension of $\mathbf{D}$ is large (which is often the case in BS), more efficient algorithms have to be devised. Here we employ the augmented Lagrange multiplier method [3, 16] to solve such an equality constrained optimization. We first formulate the following augmented Lagrangian function

$$\begin{aligned}L(\mathbf{B},\mathbf{F},\mathbf{Y},\mu) &= \|\mathbf{B}\|_* + \lambda\|\mathbf{F}\|_{gfl} +\\ \langle\mathbf{Y}, \mathbf{D} - \mathbf{B} - \mathbf{F}\rangle &+ \frac{\mu}{2}\|\mathbf{D} - \mathbf{B} - \mathbf{F}\|_F^2,\end{aligned} \tag{5}$$

where $\|\cdot\|_F$ is the Frobenius norm, $\mathbf{Y}$ is the Lagrangian multiplier and μ is an auxiliary positive scalar. According to [16], the optimization problem in Eq. (4) can be solved by iteratively searching for the optimal $\mathbf{B}$, $\mathbf{F}$ and $\mathbf{Y}$ to minimize Eq. (5). Under some rather general conditions, e.g. when $\{\mu_k\}$ is an increasing sequence and bounded, the searching process will converge Q-linearly to the optimal solution. We summarize the pseudo code in Algorithm 1[2] and discuss how to update $\mathbf{B}$ and $\mathbf{F}$ in each iteration.

Updating $\mathbf{B}$. We consider the following problem

$$\begin{aligned}\mathbf{B}_{k+1} &= \operatorname*{argmin}_{\mathbf{B}} L(\mathbf{B}, \mathbf{F}_k, \mathbf{Y}_k, \mu_k)\\ &= \operatorname*{argmin}_{\mathbf{B}} \|\mathbf{B}\|_* + \langle\mathbf{Y}_k, \mathbf{D} - \mathbf{B} - \mathbf{F}_k\rangle + \frac{\mu_k}{2}\|\mathbf{D} - \mathbf{B} - \mathbf{F}_k\|_F^2\\ &= \operatorname*{argmin}_{\mathbf{B}} \frac{1}{\mu_k}\|\mathbf{B}\|_* + \frac{1}{2}\|\mathbf{B} - \mathbf{M_1}\|_F^2,\end{aligned} \tag{6}$$

where $\mathbf{M_1} = \mathbf{D} - \mathbf{F}_k + \frac{1}{\mu_k}\mathbf{Y}_k$. Eq. (6) is a standard nuclear norm minimization problem, which is known to be fast solvable via Singular Value Thresholding (SVT) [6]. According to [6], the solution to Eq. (6) is

$$\mathbf{B}_{k+1} = \mathbf{U}\mathrm{T}_{\frac{1}{\mu_k}}(\mathbf{\Sigma})\mathbf{V}^T,\ \text{where}\ (\mathbf{U}, \mathbf{\Sigma}, \mathbf{V}^T) = \operatorname{svd}(\mathbf{M_1}). \tag{7}$$

$\mathrm{T}_\tau(\cdot)$ is an element-wise soft-thresholding operator, i.e. $\operatorname{diag}(\mathrm{T}_\tau(\mathbf{\Sigma})) = [\mathrm{t}_\tau(\sigma_1), \mathrm{t}_\tau(\sigma_2), ..., \mathrm{t}_\tau(\sigma_r)]$ where $\mathrm{t}_\tau(\sigma)$ is defined as

$$\mathrm{t}_\tau(\sigma) = \operatorname{sign}(\sigma)\max(|\sigma| - \tau, 0). \tag{8}$$

[2] Notice that Algorithm 1 is an approximate version of the original ALM. This approximation generally gives satisfactory results but converges much faster in practice [16].

Updating $\mathbf{F}$. Now we consider the updating of $\mathbf{F}$

$$\begin{aligned}
&\mathbf{F}_{k+1} = \underset{\mathbf{F}}{\operatorname{argmin}}\, L(\mathbf{B}_{k+1}, \mathbf{F}, \mathbf{Y}_k, \mu_k) \\
&= \underset{\mathbf{F}}{\operatorname{argmin}}\, \lambda \|\mathbf{F}\|_{gfl} + \langle \mathbf{Y}_k, \mathbf{D} - \mathbf{B}_{k+1} - \mathbf{F} \rangle \\
&\qquad + \frac{\mu_k}{2} \|\mathbf{D} - \mathbf{B}_{k+1} - \mathbf{F}\|_F^2 \\
&= \underset{\mathbf{F}}{\operatorname{argmin}}\, \frac{\lambda}{\mu_k} \|\mathbf{F}\|_{gfl} + \frac{1}{2} \|\mathbf{F} - \mathbf{M_2}\|_F^2 \\
&= \underset{\mathbf{f}^{(l)}}{\operatorname{argmin}} \sum_{l=1}^{n} \Big\{ \frac{\lambda}{\mu_k} \|\mathbf{f}^{(l)}\|_1 + \frac{\lambda\rho}{\mu_k} \sum_{(i,j)\in\mathcal{N}} w_{ij}^{(l)} |f_i^{(l)} - f_j^{(l)}| \\
&\qquad + \frac{1}{2} \|\mathbf{f}^{(l)} - \mathbf{m}^{(l)}\|_2^2 \Big\},
\end{aligned} \tag{9}$$

where $\mathbf{M_2} = \mathbf{D} - \mathbf{B}_{k+1} + \frac{1}{\mu_k}\mathbf{Y}_k$ and $\mathbf{m}^{(l)}$ is the l-th column of $\mathbf{M_2}$. Notice that in Eq. (9), the optimizations of each column are independent of each other. Therefore, solving Eq. (9) equals to n times of solving the following problem

$$\mathbf{f}^* = \underset{\mathbf{f}}{\operatorname{argmin}}\, \lambda_1 \|\mathbf{f}\|_1 + \lambda_2 \sum_{(i,j)\in\mathcal{N}} w_{ij} |f_i - f_j| + \frac{1}{2} \|\mathbf{f} - \mathbf{m}\|_2^2, \tag{10}$$

where $\lambda_1 = \frac{\lambda}{\mu_k}$ and $\lambda_2 = \frac{\lambda\rho}{\mu_k}$. In order to solve Eq. (10), according to [10], we introduce the following Lemma.

Lemma 2.1. *Suppose we have*

$$\hat{\mathbf{f}} = \underset{\mathbf{f}}{\operatorname{argmin}}\, \lambda_2 \sum_{(i,j)\in\mathcal{N}} w_{ij} |f_i - f_j| + \frac{1}{2} \|\mathbf{f} - \mathbf{m}\|_2^2, \tag{11}$$

the solution to Eq. (10)*, i.e.* $\mathbf{f}^*$*, can be achieved by element-wise soft-thresholding such that* $f_i^* = t_{\lambda_1}(\hat{f}_i)$ *for* $i = 1, ..., p$*.*

The proof can be shown by exploring the optimality condition of Eq. (10) and (11). We provide the sketch of the proof as follows. A rigorous proof can be found in [10].

Proof. We define the objectives in Eq. (10) and (11) as $g_1(\mathbf{f})$ and $g_2(\mathbf{f})$ respectively. Since $\hat{\mathbf{f}}$ is the optimizer of $g_2(\mathbf{f})$, it satisfies $\partial g_2(\mathbf{f})/\partial \mathbf{f} = 0$ (sub-gradient is applied where necessary). Because the additional $\|\mathbf{f}\|_1$ term in Eq. (10) is separable with respect to f_i, after applying the element-wise soft-thresholding e.g. $f_i^* = \mathrm{t}_{\lambda_1}(\hat{f}_i)$, the resulted $\mathbf{f}^*$ satisfies $\partial g_1(\mathbf{f})/\partial \mathbf{f} = 0$. □

Due to Lemma 2.1, we can first solve Eq .(11) and then use such an element-wise soft-thresholding technique to finally solve Eq (10) and therefore update $\mathbf{F}$. Notice that Eq. (11) is a continuous total variation formulation. In [25, 24], it is shown that Eq (11) is equivalent to a parametric graph-cut problem which can be efficiently solved via fast flow algorithms such as the parametric-flow proposed in [11].

2.3. Supervised Model Learning

In the situation where pure background frames are given (i.e. SML), we can of course still apply the same method above for background subtraction. However, by doing so, we do not fully exploit the provided information about the background. To utilized such extra information, we derive a variant of the model introduced above.

We separate the observation matrix $\mathbf{D}$ as $\mathbf{D} = [\mathbf{D}_1, \mathbf{D}_2]$, where $\mathbf{D}_1$ is the matrix of all pure background frames (the training data) and $\mathbf{D}_2$ is the matrix containing the rest frames with mixed content. The unknown $\mathbf{B}$ and $\mathbf{F}$ are separated correspondingly. We assume $\mathbf{D}_1 = \mathbf{B}_1$ and thus $\mathbf{F}_1 = \mathbf{0}$. By applying them to Eq. (1), we have

$$\begin{aligned}
\min_{\mathbf{B},\mathbf{F}}\ & \operatorname{rank}([\mathbf{B}_1, \mathbf{B}_2]) + \lambda \|\mathbf{F}_2\|_{gfl} \\
\text{s.t. }\ & \mathbf{D}_1 = \mathbf{B}_1,\ \ \mathbf{D}_2 = \mathbf{B}_2 + \mathbf{F}_2,
\end{aligned} \tag{12}$$

We now make another assumption that $\operatorname{rank}([\mathbf{B_1}, \mathbf{B_2}]) = \operatorname{rank}(\mathbf{B}_1)$. The idea behind this assumption is that if we have enough pure background frames, the corresponding background vectors fully span the background subspace. By taking this assumption, the columns of the unknown $\mathbf{B}_2$ can be represented using linear combinations of the columns of $\mathbf{B}_1$. Specifically, we have $\mathbf{B}_2 = \mathbf{B}_1\mathbf{S} = \mathbf{D}_1\mathbf{S}$, where $\mathbf{S}$ is the coefficient matrix. Thus, Eq. (12) becomes

$$\begin{aligned}
\min_{\mathbf{D}_1,\mathbf{S},\mathbf{F}_2}\ & \operatorname{rank}(\mathbf{D}_1[\mathbf{I}, \mathbf{S}]) + \lambda \|\mathbf{F}_2\|_{gfl} \\
\text{s.t. }\ & \mathbf{D}_2 = \mathbf{D}_1\mathbf{S} + \mathbf{F}_2.
\end{aligned} \tag{13}$$

Interestingly, since $\mathbf{D}_1$ is observed/given and its rank is irrelevant to the optimization. As before, we assume $\mathbf{D}_1$ to be low-rank, therefore there must exists a sparse $\mathbf{S}$. This is because each column of $\mathbf{B_2}$ can be represented as a linear combination of a small number of the columns of $\mathbf{D}_1$ (given that $\mathbf{D}_1$ is low-rank). So we can instead propose to solve

$$\begin{aligned}
\min_{\mathbf{S},\mathbf{F}_2}\ & \|\mathbf{S}\|_1 + \lambda \|\mathbf{F}_2\|_{gfl} \\
\text{s.t. }\ & \mathbf{D}_2 = \mathbf{D}_1\mathbf{S} + \mathbf{F}_2,
\end{aligned} \tag{14}$$

where $\|\cdot\|_1$ is a convex surrogate for $\|\cdot\|_0$, which counts the number of non-zero entries.

Eq. (14) is our SML BS model. Since it is again an equality constrained optimization, the ALM introduced above can still be applied. This time, when updating the foregrounds, the optimization is the same as before, except that we are now dealing with $\mathbf{F}_2$ instead of $\mathbf{F}$. While updating the background, we solve the following problem

$$\mathbf{S}_{k+1} = \underset{\mathbf{S}}{\operatorname{argmin}}\, \frac{1}{\mu_k} \|\mathbf{S}\|_1 + \frac{1}{2} \|\mathbf{D}_1\mathbf{S} - \mathbf{M}\|_F^2. \tag{15}$$

Eq. (15) can be further decomposed into column-wise optimization and each of which is a standard Lasso [22]

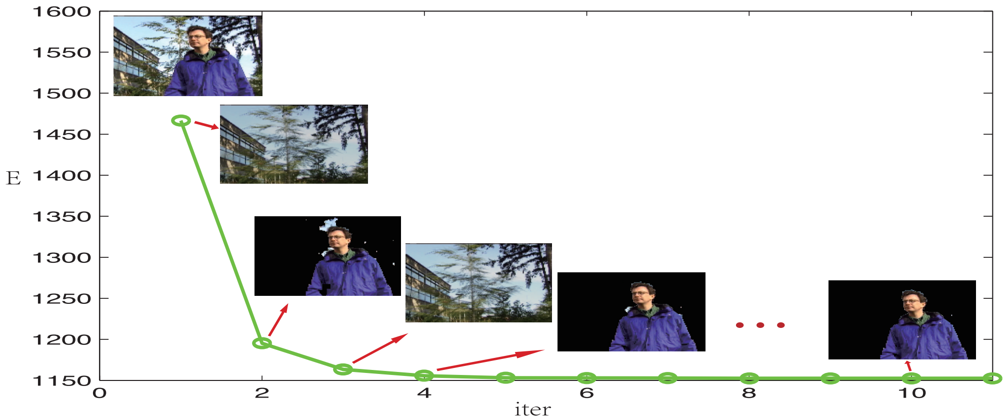


Figure 1. Alternated updating of the background and the foreground. In each iteration (iter) either the background model or the foreground is updated and the objective value (the green plots) keeps decreasing until convergence.

problem. Many fast algorithms can be applied e.g. the FISTA algorithm proposed in [2].

In summary, we can effectively solve both the UML and SML BS models by applying the ALM algorithm described in Algorithm 1. Detailed updating rules for both the background $\mathbf{B}$ and the foreground $\mathbf{F}$ are given above. Interestingly, although ALM is a general optimization method, its application to BS helps us to understand how our model alternately pursues and refines the background and the foregrounds. In Figure 1, we visualize the estimation in each iteration of ALM. We observe that the foreground estimation becomes better as the iteration goes on. This is mainly due to the synergy of the background estimation and foreground estimation.

3. Experiments

3.1. Data sets

We test our model on three popular BS data sets, namely, the Wallflower[3] data set [23] , the Li[4] data set [15] and the SABS[5] data set [5]. All together, there are 17 sequences of both natural and synthetic videos. Most well-known BS challenges are presented in these sequences, e.g. gradual/sudden illumination changes, moving background, bootstrapping, camouflage, and occlusion etc. We give a brief introduction of these data sets respectively as follows.

- "Wallflower": The Wallflower data set consists of 7 natural video sequences representing different BS challenges. The resolution of the frames is about 160×120. Manually labeled ground truth are provided. Most of the sequences have pure background frames, which can be used for SML.
- "Li": The Li data set consists of 9 natural video sequences. The resolution of the frames is about 176×144. Manually labeled ground truth are provided. Part of them have pure background frames.
- SABS: The SABS data set is a synthetic data set and therefore provides high quality ground truth. The resolution of the frames is 800×600. Several BS challenges are synthesized to the same scene. Following [12], only the basic sequence is used. Pure background frames are available.

All three data sets are popular public data sets. Results of many existing models have been reported based on these sets. In order to evaluate the proposed model, we directly compare with the results reported in the respective papers.

3.2. Comparison with RPCA

Recall that the proposed model in the UML case can be reduced to RPCA when the model parameter $w_{ij} = 0$ in Eq.2 (Section.2.1). Therefore, we first show how the proposed model improves performance over the RPCA model due to GFL foreground modeling.

Quantitative comparisons on all the sequences of the three data sets are shown in Table 2∼4. From the comparisons we can see that the proposed model consistently outperforms RPCA. Qualitatively, we use the same 200 frames of the airport sequence in "Li" data set as reported in [7] to construct a head-to-head comparison, where we apply both RPCA and our model for background subtraction. In Figure 2, we illustrate the BS results of the test frame used in [7]. In practice, even after fine-grid search for the best parameters, the detected foregrounds of RPCA have more "holes" and more false positives from background than those of the proposed model. (Some obvious examples are marked by

[3] http://research.microsoft.com/en-us/um/people/jckrumm/Wallflower/testimages.htm

[4] http://perception.i2r.a-star.edu.sg/bk_model/bk_index.html

[5] http://www.vis.uni-stuttgart.de/forschung/informationsvisualisierung-und-visual-analytics/visuelle-analyse-videostroeme/sabs.html

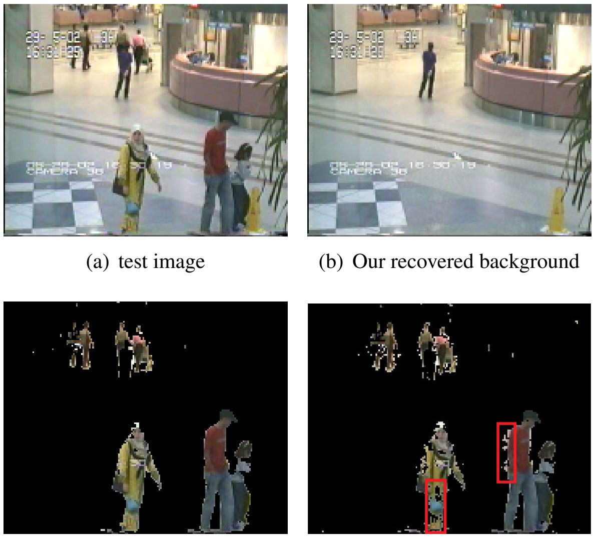
Table 1. Brief summaries of the models compared. (Part of the descriptions are from [12, 5])

methods	notation	description
[1]	KDE	Kernel density estimation (KDE) with a spherical kernel. Uses a stochastic history.
[9]	G-KDE	Neural network variant of Gaussian KDE.
[14]	C-KDE	Codebook based; almost KDE with a cuboid kernel.
[15]	Hist	Histogram based, includes co-occurrence statistics. Lots of heuristics.
[17]	Map	Uses a self organising map, passes information between pixels.
[21]	MoG	Classic MoG approach. Assigns mixture components to bg/fg.
[27]	R-MoG	Refinement of [21]. Has an adaptive learning rate.
[20]	Eigen	Eigenbackground.
[18]	Gauss	unimodal (Gaussian)
[12]	D-MoG	Dirichlet process Gaussian mixture Model.
[7]	RPCA	Robust PCA model.
[26]	G-Lasso	Online subspace learning with group lasso with overlap regularization.

Table 2. Results for the SABS data set, given as the F-score.

	Gauss	C-KDE	Eigen	Map	KDE	R-MoG	MoG	Hist	RPCA	G-Lasso	**Ours**
F-score	.3806	.5601	.5891	.6672	.7177	.7232	.7284	.7457	.6483	0.7326	**.7775***

the red boxes in Figure 2 (d). Note that these results are not post-processed.) Qualitative results of the whole sequence are provided on our webpage[6].


(a) test image

(b) Our recovered background

(c) Our detected foregrounds

(d) RPCA detected foreground

Figure 2. Results comparison of RCPA and our model for the airport data from the Li data set.

3.3. Comparison with State-of-the-Art

A brief summary of all the models we compared can be found in Table 1. We compare our model to these models on all three data sets[7]. Following the literature, for the "Wallflower" data set, mis-classified number of pixels is used as the evaluation criteria; for both "Li" and "SABS," F-score (F) is used as the evaluation criterion. We put a "$*$" on the upper-right corner of the scores to indicate that the sequence is of the SML case.

On Wallflower data set. We tested our model on all the seven sequences of this data set. In Table 3, we provide quantitative comparisons, where our model achieved the least mis-classified number of pixels on five sequences and the second least on one sequence. Note, however, our model performed poorly on the sequence "CF". The reason is that the foreground in "CF" occupies a large portion of the tested frame, which violates the prior assumption on foreground sparsity. The same failure happened to both the RPCA and G-Lasso models, since both of them also assume sparse foreground prior. In Figure 3, we show the qualitative results of our model on the seven sequences.

On Li data set. We applied our model to all the nine sequences of the data set. In Table 4, we show quantitative comparisons, where our model achieved the highest F-score on all these sequences. Notably, in some sequences such as "lb", "ap" etc., the improvements over the second best are more than 10%. On average, our model achieved an 8% F-score gain ahead of the second best model. The qualitative results of all nine sequences are shown in Figure 4.

On SABS data set. Following [12], we apply our model to the "Basic" sequence and compared with the other models on this representative sequence. The results of different

[6] http://idm.pku.edu.cn/staff/wangyizhou/

[7] Note that, since we are using the results reported by respective papers, not all the models have results on every sequence.

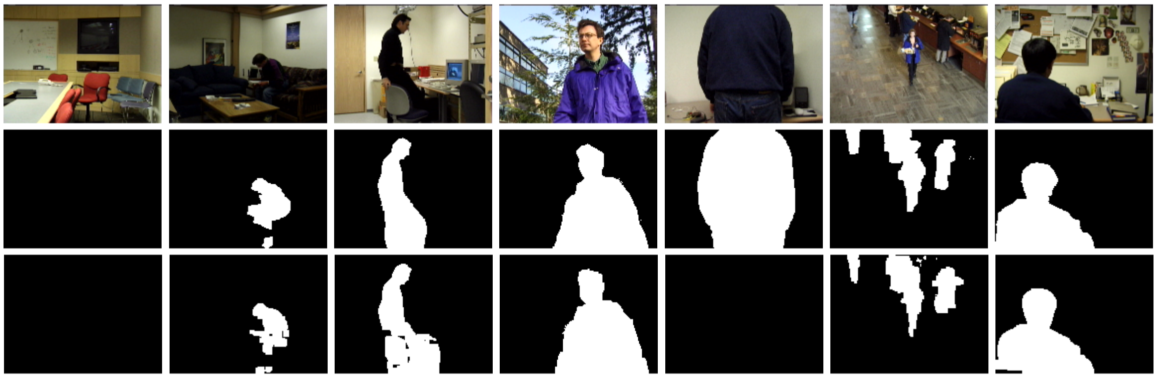

(a) MO (b) TD (c) LS (d) WT (e) CF (f) BS (g) FA

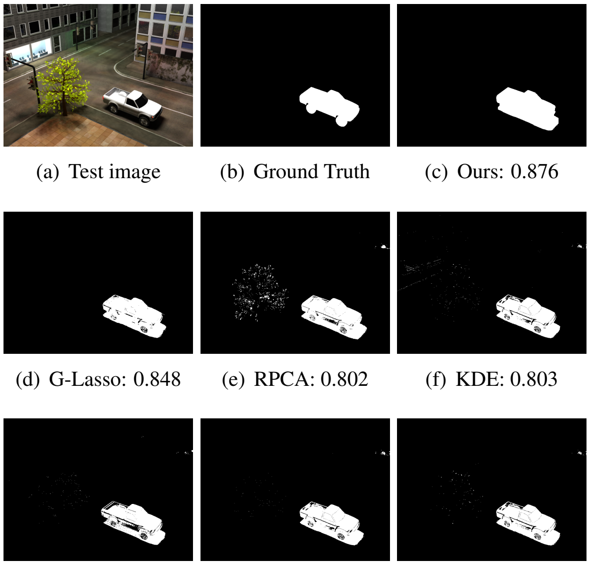
Figure 3. Results on Wallflower data set. From top to bottom: test images, the ground truth and the estimations of our model.

Table 3. Results for Wallflower, given as the number of pixels that have been mix-classified.

methods	MO	TD	LS	WT	CF	BS	FA
Frame Difference	0	1358	2565	6789	10070	2175	4354
Mean+threshold	0	2593	16232	3285	1832	3236	2818
Block correlation	1200	1165	3802	3771	6670	2673	2402
MoG	0	1028	15802	1664	3496	2091	2972
Eigen	1065	895	1324	3084	1898	6433	2978
D-MoG	0	**330**	3945	184	**384**	1236	1569
RPCA	0	628	2016	1014		1465	2875
G-Lasso	0	912	1067	629		1779	1139
Ours	**0***	418*	**686***	**166***		**795***	**192***

models on an example frame (No. 448) are illustrated in Figure 5. (The qualitative results on the whole sequence can be found on our webpage.) As is shown, our model almost cuts a perfect foreground (including its shadow). In the ground truth, the shadow is not included, which makes the value of the F-score relatively low. However, this definition of foreground may be controversial depending on the actually situations. Nevertheless, our model outperforms all the rest models on the test image. The average F-scores of all the models on the whole sequence are summarized in Table 2, where our model is shown to have achieved the highest performance.

Compare with group lasso. As mentioned in the related works of Section 1, the group lasso regularization was applied to modeling foregrounds of BS in [26] . The authors used both "3 × 3 blocks group" and "coarse-to-fine superpixel group" structures to pursue connected sparse foregrounds. However, as can be seen from the above comparisons e.g. Table 4, 3 & 2 and Figure 5, their performance are not as good as those of the proposed model. In Figure 6, we provide a close-up comparison with the deliberately pre-defined grouping of pixels for foreground modeling. It shows that the group lasso model generates artifacts of detected foreground objects due to inappropriate pre-defined group structure. This arguably indicates that, compared to (adaptive) GFL, the group sparse models may not be flexible enough for recovering arbitrary foreground shapes.


(a) Test image (b) Ground Truth (c) Ours: 0.876

(d) G-Lasso: 0.848 (e) RPCA: 0.802 (f) KDE: 0.803

(g) Hist 0.782 (h) MoG: 0.819 (i) R-MoG: 0.807

Figure 5. Results on the SABS data set. F-scores are shown.

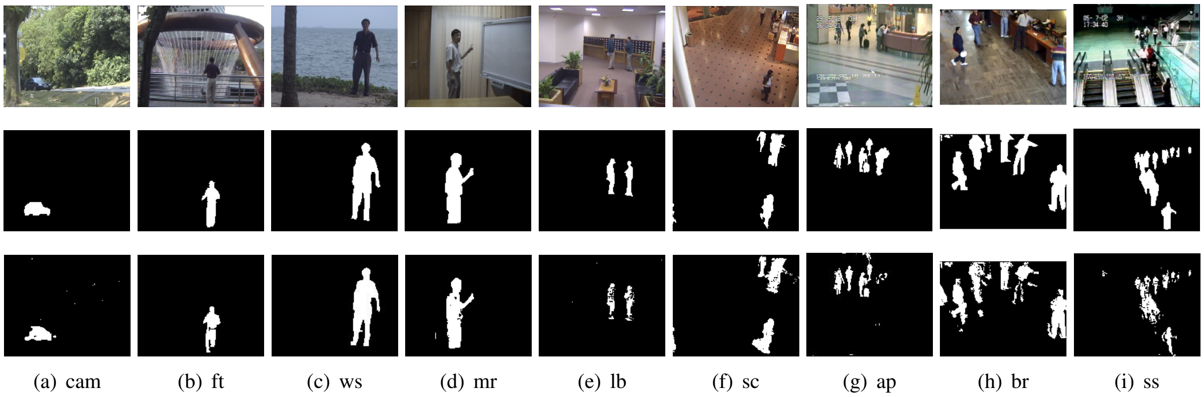


Figure 4. Results on Li data set. From top to bottom: test images, the ground truth and the estimations of our model.

Table 4. Results for Li, given as F-score.

methods	cam	ft	ws	mr	lb	sc	ap	br	ss	mean
Hist	.1596	.0999	.0667	.1841	.1554	.5209	.1135	.3079	.1294	.1930
MoG	.0757	.6854	.7948	.7580	.6519	.5363	.3335	.3838	.1388	.4842
Map	.6960	.6554	.8247	.8178	.6489	.6677	.5943	.6019	.5770	.6760
D-MoG	.7624	.7265	.9134	.3871	.6665	.6721	.5663	.6273	.5269	.6498
RPCA	.5226	.8650	.6082	.9014	.7245	.7785	.5879	.8322	.7374	.7286
G-Lasso	.8347	.8789	.9236	.8995	.6996	.8019	.5616	.7475	.6432	.7767
Ours	**.8386**	**.9011**	**.9424***	**.9592**	**.8208**	**.8500**	**.7422**	**.8476**	**.7613**	**.8515**

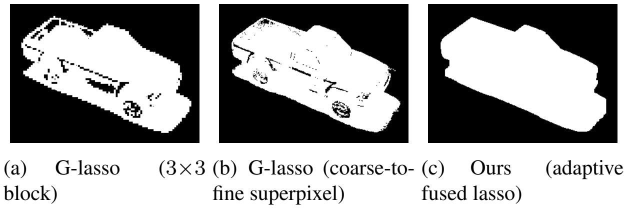

(a) G-lasso (3×3 block)
(b) G-lasso (coarse-to-fine superpixel)
(c) Ours (adaptive fused lasso)

Figure 6. Different foreground regularization comparison.

3.4. Discussions

Computation. The algorithm does not take many iterations to converge, see e.g. Figure 1, and in practice the average number of iterations is about 10-20. Therefore, the major computational cost to pursue structured background and foregrounds in the mid-steps can be eased up by this few iterations. Moreover, since the updating of the foreground are column-wise, the implementation can be highly paralleled in practice. The code can be downloaded on our webpage.

SML vs. UML. Note that, in general when pure background frames are available, like most of the sequences in the Wallflower dataset, we have reported the results of the SML model. Such a choice outperforms its unsupervised counterpart, e.g. with an improvement of 24 (for WT) to 179 (for TD) pixels on the Wallflower dataset. However, this is not always the case. For example, in the "cam" sequence of the Li dataset, although there are pure background frames, they seem to be less representative possibly due to some background changes. Then, the supervised model did not achieve obviously better results but still competitive, in this case: 0.8382 vs 0.8386.

Comparison with Explicit Post-processing. Arguably, explicit post-processing in BS e.g. [19] can be viewed as a special case of foreground modeling since these methods are fundamentally using foreground structural priors to guide post-processing. Therefore, we carried out some experiments with the data used in [19], where MoG models are post-processed by a hole-filling method. In summery, our model achieved competitive or even better results, detailed comparisons can be found on our webpage. .

4. Conclusion

In this paper, we propose a method of background subtraction by exploiting structure information of the foregrounds to help background modeling. Our model works for both supervised and unsupervised learning paradigms and automatically pursue meaningful background and foregrounds. To optimize the new objective function, we proposed an effective algorithm by extending the ALM, which alternatively updates the background and the foreground matrices. Experimental results show that the proposed model achieves better than state-of-the-art performance on several popular public data sets.

Acknowledgements. This work was supported in part by 973 grant 2015CB351800 and NSFC grants 61272027, 61231010, 61421062, 61210005.

References

[1] O. Barnich and M. Van Droogenbroeck. Vibe: A universal background subtraction algorithm for video sequences. *Image Processing, IEEE Transactions on*, 20(6):1709–1724, 2011.

[2] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[3] D. P. Bertsekas. Constrained optimization and lagrange multiplier methods. *Computer Science and Applied Mathematics, Boston: Academic Press, 1982*, 1, 1982.

[4] Y. Boykov and G. Funka-Lea. Graph cuts and efficient nd image segmentation. *International Journal of Computer Vision*, 70(2):109–131, 2006.

[5] S. Brutzer, B. Hoferlin, and G. Heidemann. Evaluation of background subtraction techniques for video surveillance. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1937–1944. IEEE, 2011.

[6] J.-F. Cai, E. J. Candès, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010.

[7] E. J. Candès, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *Journal of the ACM (JACM)*, 58(3):11, 2011.

[8] X. Cui, J. Huang, S. Zhang, and D. N. Metaxas. Background subtraction using low rank and group sparsity constraints. In *Computer Vision–ECCV 2012*, pages 612–625. Springer, 2012.

[9] D. Culibrk, O. Marques, D. Socek, H. Kalva, and B. Furht. Neural network approach to background modeling for video object segmentation. *Neural Networks, IEEE Transactions on*, 18(6):1614–1627, 2007.

[10] J. Friedman, T. Hastie, H. Höfling, R. Tibshirani, et al. Pathwise coordinate optimization. *The Annals of Applied Statistics*, 1(2):302–332, 2007.

[11] G. Gallo, M. D. Grigoriadis, and R. E. Tarjan. A fast parametric maximum flow algorithm and applications. *SIAM Journal on Computing*, 18(1):30–55, 1989.

[12] T. S. Haines and T. Xiang. Background subtraction with dirichlet processes. In *Computer Vision–ECCV 2012*, pages 99–113. Springer, 2012.

[13] C. Jordan. Sur la série de fourier. *CR Acad. Sci. Paris*, 92(5):228–230, 1881.

[14] K. Kim, T. H. Chalidabhongse, D. Harwood, and L. Davis. Background modeling and subtraction by codebook construction. In *Image Processing, 2004. ICIP'04. 2004 International Conference on*, volume 5, pages 3061–3064. IEEE, 2004.

[15] L. Li, W. Huang, I.-H. Gu, and Q. Tian. Statistical modeling of complex backgrounds for foreground object detection. *Image Processing, IEEE Transactions on*, 13(11):1459–1472, 2004.

[16] Z. Lin, M. Chen, and Y. Ma. The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. *arXiv preprint arXiv:1009.5055*, 2010.

[17] L. Maddalena and A. Petrosino. A self-organizing approach to background subtraction for visual surveillance applications. *Image Processing, IEEE Transactions on*, 17(7):1168–1177, 2008.

[18] S. J. McKenna, S. Jabri, Z. Duric, A. Rosenfeld, and H. Wechsler. Tracking groups of people. *Computer Vision and Image Understanding*, 80(1):42–56, 2000.

[19] A. Nurhadiyatna, W. Jatmiko, B. Hardjono, A. Wibisono, I. Sina, and P. Mursanto. Background subtraction using gaussian mixture model enhanced by hole filling algorithm (gmmhf). In *Systems, Man, and Cybernetics (SMC), 2013 IEEE International Conference on*, pages 4006–4011. IEEE, 2013.

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

[21] C. Stauffer and W. E. L. Grimson. Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, volume 2. IEEE, 1999.

[22] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[23] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers. Wallflower: Principles and practice of background maintenance. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 1, pages 255–261. IEEE, 1999.

[24] B. Xin, L. Hu, Y. Wang, and W. Gao. Stable feature selection from brain smri. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2014.

[25] B. Xin, Y. Kawahara, Y. Wang, and W. Gao. Efficient generalized fused lasso and its application to the diagnosis of alzheimers disease. In *Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.

[26] J. Xu, V. K. Ithapu, L. Mukherjee, J. M. Rehg, and V. Singh. Gosus: Grassmannian online subspace updates with structured-sparsity. In *The IEEE International Conference on Computer Vision (ICCV)*, 2013.

[27] Z. Zivkovic and F. van der Heijden. Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern recognition letters*, 27(7):773–780, 2006.