Cramér–Rao Bound for Line Constrained Trajectory Tracking

Amr Elnakeeb and Urbashi Mitra
Ming Hsieh Department of Electrical Engineering, Viterbi School of Engineering
University of Southern California, Los Angeles, CA
{elnakeeb,ubli}@usc.edu

***Abstract*—In this paper, target tracking constrained to short-term linear trajectories is explored. The problem is viewed as an extension of the matrix decomposition problem into low-rank and sparse components by incorporating an additional line constraint. The Cramér–Rao Bound (CRB) for the trajectory estimation is derived; numerical results show that an alternating algorithm which estimates the various components of the trajectory image is near optimal due to proximity to the computed CRB. In addition to the theoretical contribution of incorporating an additional constraint in the estimation problem, the alternating algorithm is applied to real video data and shown to be effective in estimating the trajectory despite it not being exactly linear.**

***Index Terms*—Cramér–Rao bound, sparse methods, low-rank, augmented Lagrange Multiplier method, object tracking.**

I. Introduction

Low-rank and sparse structures have been ubiquitously considered in many applications, *e.g*, pattern analysis as well as image classification and reconstruction [1]–[5]. Background subtraction can be viewed as a decomposition of a low-rank matrix representing the background and a sparse matrix consisting of the foreground objects [3]–[6]. In our prior work [7], we introduced an additional linear constraint to these background subtraction methods which enables the finding of "lines" in videos. Exploiting application-unique, additional structure often yields strong improvements in performance to sparse/low-rank based methods (see *e.g.* [7]–[9]).

Target tracking is an important signal processing tool that has been considerably examined over the years [10], [11], with myriad of applications e.g., video-surveillance and navigation of vehicles. Sparse and other structured representations have been exploited to target tracking [3], [7], [12].

In [7], we extended the Robust Principal Component Analysis (RPCA) model considered in [4], [5] by adapting this low-rank and sparse framework to target tracking for objects on nearly linear trajectories— motivated by the fact that line segments arise in a lot of natural and synthetic objects. Furthermore, some complex objects can be represented by a combination of multiple linear features [13]; therefore the proposed approach could be exploited in larger, more sophisticated systems. The expected linear trajectory is translated into additional constraints for the optimization. Simualtion results showed superiority of the method proposed in [7] over other background subtracion methods proposed in [4], [5]. Herein, we derive a CRB to evaluate the potential optimality of the method proposed in [7]. Our derivation builds upon the work [14]–[18]. In particular, [14] presents the computation of the CRB for the traditional RPCA problem. In [19], the impact of multiple constraints is questioned for certain low rank and sparse problems with a particular structure. We examine the impact of selecting different subsets of the multiple constraints and confirm that all constraints are in fact needed to achieve the best performance. The main contributions of this paper are summarized as follows:

1) We adapt the derivations in [14]–[16] to accommodate the additional linear constraint. The computed CRB predicts that employing the additional constraint will reduce the mean-squared error as is evidenced by the numerical results.
2) Numerical results of performance of the new augmented Lagrange method of [7], which we called *Line Estimation via the Augmented Lagrange Multiplier* (**LE-ALM**), are shown to strongly improve over previous background subtraction methods [4], [5] with average improvement of 5 dB and 3.5 dB over [4], [5], respectively, and more importantly are very close to the computed CRB, average deviation of 1.2 dB, suggesting the near-optimality of our method.
3) Different from [7], we apply our target tracking algorithm to real video data, and observe that even for imperfectly linear trajectories, the algorithm works very well. We also show the efficacy of exploiting multiple structures rather than using just a single structure, by taking out one constraint and see how the performance degrades accordingly.

The rest of this paper is organized as follows: The signal model and optimization are introduced in Section II. The CRB for background subtraction with linear trajectories is derived in Section III. Simulation results are presented in Section IV. Finally, Section V concludes the paper.

II. Signal Model and Optimization Problem

We expand the model of RPCA given in [4], [5], where: $\boldsymbol{X} = \boldsymbol{L} + \boldsymbol{S}$, $\boldsymbol{X} \in \mathbb{R}^{m \times n}$, and $\boldsymbol{L}, \boldsymbol{S}$ have the same size of $\boldsymbol{X}$. The matrix $\boldsymbol{X}$ collects the observed frames which are vectorized column-wise. The matrix $\boldsymbol{L}$ is low rank and captures the background information. The matrix $\boldsymbol{S}$ is sparse and captures the motion of the object of interest. Fig. 1 provides a depiction of the signal model, wherein video frames are vectorized and collected in the observation matrix $\boldsymbol{X}$; the object is represented by black pixels. The optimization problem, that we proposed in [7], is as follows:

$$\begin{aligned} \underset{\boldsymbol{L},\boldsymbol{S},\boldsymbol{R}}{\text{minimize}} \quad & \|\boldsymbol{L}\|_* + \lambda_1 \|\boldsymbol{S}\|_{2,1} + \lambda_2 \|\boldsymbol{RS}\|_* \\ \text{subject to} \quad & \boldsymbol{X} = \boldsymbol{L} + \boldsymbol{S}, \qquad \boldsymbol{RR}^T = \boldsymbol{I}_m, \end{aligned} \tag{1}$$

where $\boldsymbol{R}$ is a rotation matrix; $\boldsymbol{R} \in \mathbb{R}^{m \times m}$, and $\boldsymbol{I}_m$ is $m \times m$ identity matrix. The nuclear norm of a general

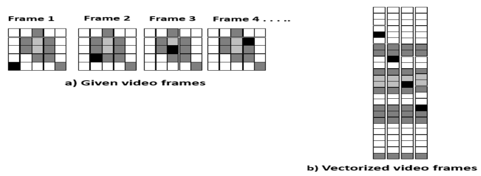

Figure 1. Signal model.

matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ is given as $\|\boldsymbol{A}\|_*$= Trace$\left(\sqrt{\boldsymbol{A}^T\boldsymbol{A}}\right) = \sum_{i=1}^{\min(m,n)} \sigma_i(\boldsymbol{A})$. The mixed $l_{2,1}$ norm of a general matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ is given as $\|\boldsymbol{A}\|_{2,1}$= $\sum_{i=1}^{m}\sqrt{\sum_{j=1}^{n} a_{ij}^2}$.

The first constraint is the usual decomposition constraint into low-rank and sparse components as in [4], [5]. The second constraint guarantees that $\boldsymbol{R}$ is a rotation matrix. We observe that if the line is appropriately "de-rotated," the resulting matrix $\boldsymbol{RS}$ will be rank one. Thus the line structure is captured by the third term in the objective function. To avoid confusion, it shall be mentioned that $\boldsymbol{R}$ is not a rotation of space; but rather it is just a transformation matrix that preserves energy before and after transformation; that is, $\|\boldsymbol{RS}\|_F^2 = \|\boldsymbol{S}\|_F^2$. With simple mathematical matrix algebra, we obtain the second constraint presented in (1).

For easier mathematical analysis, we define $\boldsymbol{S_r}$ as the matrix $\boldsymbol{S}$ after rotating via $\boldsymbol{R}$, *i.e*, $\boldsymbol{S_r} = \boldsymbol{RS}$, then the optimization model, as proposed in [7], can be rewritten as follows:

$$\begin{aligned} \underset{\boldsymbol{L},\boldsymbol{S},\boldsymbol{R},\boldsymbol{S_r}}{\text{minimize}} \quad & \|\boldsymbol{L}\|_* + \lambda_1\|\boldsymbol{S}\|_{2,1} + \lambda_2\|\boldsymbol{S_r}\|_* \\ \text{subject to} \quad & \boldsymbol{X} = \boldsymbol{L} + \boldsymbol{S}, \qquad \boldsymbol{R}\boldsymbol{R}^T = \boldsymbol{I}_m, \\ & \boldsymbol{S_r} = \boldsymbol{RS}. \end{aligned} \tag{2}$$

The above problem is solved in [7] via the method of alternating Augmented Lagrange Multipliers (ALM) [20]–[23] by decomposing the problem into four subproblems that are solved iteratively. Herein, we provide the final estimators of the four alternating terms. Further details are found in [7].

The final estimate of the matrix $\boldsymbol{L}$ is given by:

$$\hat{\boldsymbol{L}} = \boldsymbol{U}T_{\frac{1}{\beta}}(\boldsymbol{W})\boldsymbol{\Sigma}\boldsymbol{V}^T, \tag{3}$$

where $\boldsymbol{W} = \beta^{-1}\boldsymbol{M} + \boldsymbol{X} - \boldsymbol{S}$, $\boldsymbol{U\Sigma V}^T$ is the Singular Value Decomposition (SVD) of $\boldsymbol{W}$; with $\boldsymbol{M}$ being the Lagrangian multiplier corresponding to the first constraint, β is a positive scalar, and $T_\alpha(\cdot)$ is a soft thresholding operator defined as $T_\alpha(x) = \text{sign}(x)\max\{|x| - \alpha, 0\}$; for matrices, $T_\alpha(\cdot)$ is applied component-wise.

The final estimate of the matrix $\boldsymbol{S}$ is given by solving:

$$\frac{\lambda_1}{\beta}\boldsymbol{\Lambda}\,\text{sign}(\boldsymbol{Z}) + \boldsymbol{S} - \boldsymbol{Z} + \boldsymbol{R}^T(\boldsymbol{RS} - \boldsymbol{E}) = 0, \tag{4}$$

where $\boldsymbol{Z} = \beta^{-1}\boldsymbol{M} + \boldsymbol{X} - \boldsymbol{L}$, $\boldsymbol{E} = \beta^{-1}\boldsymbol{V} + \boldsymbol{S_r}$; with $\boldsymbol{V}$ being the Lagrangian multiplier corresponding to the second constraint, $\boldsymbol{\Lambda} = \text{diag}\left(\|\boldsymbol{S}^{j\rightarrow}\|_2^{-1}\right)|\boldsymbol{S}|$, and $\boldsymbol{S}^{j\rightarrow}$ is the j^{th} row of $\boldsymbol{S}$. Solving Equation (4) iteratively for $\boldsymbol{S}$ yields $\hat{\boldsymbol{S}}$.

The final estimate of the matrix $\boldsymbol{R}$ is given by solving:

$$2\boldsymbol{R}\boldsymbol{R}^T\boldsymbol{R} - (\boldsymbol{K}^T + \boldsymbol{K})\boldsymbol{R} + \boldsymbol{R}\boldsymbol{S}\boldsymbol{S}^T - \boldsymbol{E}\boldsymbol{S}^T = 0, \tag{5}$$

where $\boldsymbol{K} = \beta^{-1}\boldsymbol{N} + \boldsymbol{I}_m$; with $\boldsymbol{N}$ being the Lagrangian multiplier corresponding to the third constraint, and use $\boldsymbol{E}$ as defined before. Again, solving Equation (5) iteratively provides the estimate $\hat{\boldsymbol{R}}$.

Finally, the estimate of the matrix $\boldsymbol{S_r}$ is given by:

$$\hat{\boldsymbol{S}}_{\boldsymbol{r}} = \boldsymbol{U}T_{\frac{\lambda_2}{\beta}}(\boldsymbol{J})\boldsymbol{\Sigma}\boldsymbol{V}^T, \tag{6}$$

where $\boldsymbol{J} = \beta^{-1}\boldsymbol{V} + \boldsymbol{RS}$, $\boldsymbol{U\Sigma V}^T$ is the SVD of $\boldsymbol{J}$, and $T_\alpha(\cdot)$ is the soft thresholding operator we defined previously. Combining these solutions yields the overall **LE-ALM** algorithm.

III. Cramér–Rao bound

In this section, we generalize the derivations presented in [14]–[16] to obtain a new CRB for our problem. Herein, we answer the question of how the CRB would change as we go from the classical RPCA to more constrained RPCA that exploits the special structure in line estimation.

We define $\mathbf{y} = \mathcal{A}(\boldsymbol{L} + \boldsymbol{S}) + \mathbf{n}$; where and $\mathcal{A}$ is a linear operator; $\mathcal{A} : \mathbb{R}^{m\times n} \rightarrow \mathbb{R}^p$, and $\mathbf{n}$ is Gaussian-distributed noise vector $\mathcal{N}(0, \boldsymbol{\Sigma})$. Hence, using the matrix-vector notation, with $\mathbf{l}$ and $\mathbf{s}$ defined as vec($\boldsymbol{L}$) and vec($\boldsymbol{S}$), respectively, we have,

$$\mathbf{y} = \boldsymbol{A}(\mathbf{l} + \mathbf{s}) + \mathbf{n}, \tag{7}$$

where $\boldsymbol{A} \in \mathbb{R}^{p\times mn}$ is the matrix corresponding to the linear operator $\mathcal{A}$ and *vec* stands for vectorizing a given matrix column wise. We underscore that, one should not confuse the vector $\mathbf{l} = \text{vec}(\boldsymbol{L})$ with the all ones vector. We define the set

$$\mathcal{X}_{l,s} \stackrel{\text{def}}{=} \{(\boldsymbol{L}, \boldsymbol{S}) \in \mathbb{R}^{m\times n}\times\mathbb{R}^{m\times n} : \text{rank}(\boldsymbol{L}) \leq r, ||\boldsymbol{S}||_0 \leq s, \text{rank}(\boldsymbol{S}) = 1\}. \tag{8}$$

The first inequality in the definition of $\mathcal{X}_{l,s}$ represents the low-rank condition, and the second inequality represents the sparsity constraint. To enforce the line trajectory in the structure of $\boldsymbol{S}$, the third condition in $\mathcal{X}_{l,s}$ has been added. In other words, the third condition is what distinguishes our to-be-derived CRB from the CRB presented in [14]. The rank-one condition presumes that the trajectory is already in the obvious linear forms of a horizontal or vertical line. Thus the uncertainty of finding the appropriate rotation, $\boldsymbol{R}$, is removed from this computation. In this way, our computed CRB is a best-case scenario and should provide a further lower bound than the CRB incorporating the estimation of $\boldsymbol{R}$. We will see in the numerical results (Section IV), that in fact, the estimation of $\boldsymbol{R}$ results in limited uncertainty and in fact, the idealized CRB is quite close to the achieved performance where $\boldsymbol{R}$ must still be found. In our numerical results, random lines are generated with arbitrary lengths and angles and the auxiliary rotation is found via the **LE-ALM** algorithm. We then average the mean-squared error over not only noise, but also all of these possible realizations of the the trajectory. Thus the closeness of the simulated algorithm performance to the idealized CRB suggests near-optimality of our method.

Following the derivations in [14], there are no computational differences when handling the $\boldsymbol{L}$ matrix, which captures the low rank background information both in our work here and that in [14]. Differences in the derivation arise in the handling of the sparse component, $\boldsymbol{S}$. Herein, $\boldsymbol{S}$ has the additional constraint of being rank-one.

Let $\boldsymbol{L} = \boldsymbol{U}_0\boldsymbol{\Lambda}_0\boldsymbol{V}_0^T$ be the SVD of $\boldsymbol{L}$, where $\boldsymbol{U}_0 = [\mathbf{u_1}, \mathbf{u_2}, ...\mathbf{u_r}] \in \mathbb{R}^{m\times r}$, $\boldsymbol{\Lambda}_0 = \text{diag}([\lambda_1, \lambda_2, ...\lambda_r]) \in \mathbb{R}^{r\times r}$, and $\boldsymbol{V}_0 = [\mathbf{v_1}, \mathbf{v_2}, ...\mathbf{v_r}] \in \mathbb{R}^{n\times r}$. Furthermore, let $\boldsymbol{U}_1$ and $\boldsymbol{V}_1$ be the unit orthogonal bases for the spaces orthogonal to span$\{\boldsymbol{U}_0\}$ and span$\{\boldsymbol{V}_0\}$, respectively. Then, we define

$$\boldsymbol{Q}_l = [\boldsymbol{V}_1 \otimes \boldsymbol{U}_0, \boldsymbol{V}_0 \otimes \boldsymbol{U}_0, \boldsymbol{V}_0 \otimes \boldsymbol{U}_1] \in \mathbb{R}^{mn\times[(m+n)r-r^2]}.$$

Due to $\boldsymbol{Q}_l$ only relating to $\boldsymbol{L}$, this $\boldsymbol{Q}_l$ is the same as that in [14]. We next define $\boldsymbol{Q}_s \in \mathbb{R}^{mn\times \min(m+n-1,s)}$ to be the matrix whose columns are $\text{vec}\big(\mathbf{e}_i^m \mathbf{e}_j^{n^T}\big), (i,j) \in \mathcal{S}$; where $\mathcal{S} = \text{supp}(\boldsymbol{S}) = \{(i,j) \in [m]\times[n] : \boldsymbol{S}_{ij} \neq 0\}$. This $\boldsymbol{Q}_s$ is however different from the $\boldsymbol{Q}_s$ given in [14]. The change is developed in a way that limits $\boldsymbol{S}$ to be rank-one matrix per our signal model.

***Theorem* 1:** For $(\boldsymbol{L}, \boldsymbol{S}) \in \mathcal{X}_{l,s}$, the MSE for any locally unbiased estimator $(\hat{\boldsymbol{L}}, \hat{\boldsymbol{S}})$ satisfies

$$\text{MSE}_{\boldsymbol{L},\boldsymbol{S}} \geq \text{tr}\left(\begin{bmatrix}\boldsymbol{Q}_l^T\boldsymbol{A}^T\\ \boldsymbol{Q}_s^T\boldsymbol{A}^T\end{bmatrix}\boldsymbol{\Sigma}^{-1}\begin{bmatrix}\boldsymbol{A}\boldsymbol{Q}_l & \boldsymbol{A}\boldsymbol{Q}_s\end{bmatrix}\right)^{-1}, \qquad (9)$$

where $\boldsymbol{Q}_l$, $\boldsymbol{Q}_s$, and $\boldsymbol{A}$ are as defined before.

***Corollary* 1:** Let $\boldsymbol{\Sigma} = \sigma^2\boldsymbol{I}_p$. For any set Ω, let $\mathcal{P}_\Omega \in \mathbb{R}^{mn\times mn}$ denote a diagonal matrix whose diagonal entries are ones for indeces in Ω and zeros otherwise. Let $\mathcal{A}$ be the selection operator that observes the entries of $\boldsymbol{L}$ that are randomly and uniformly selected, and indexed by Ω, and $\mathcal{S}$ is a random and uniform subset of Ω. For RPCA, $i.e$, we have

$$\begin{aligned}&\bigg(\min(s, m+n-1) - N + \frac{1}{3}\frac{pN}{p-\min(s,m+n-1)}\\ &\quad + \frac{2}{3}\frac{mnN}{p-\min(s,m+n-1)}\bigg)\sigma^2 \leq \text{CRB} \leq\\ &\bigg(\min(s, m+n-1) - N + 3\frac{pN}{p-\min(s,m+n-1)}\\ &\quad + 2\frac{mnN}{p-\min(s,m+n-1)}\bigg)\sigma^2 \end{aligned} \qquad (10)$$

with probability greater than $1 - 10e^{\frac{-c}{\epsilon^2}}$. Furthermore, if $p = mn$, then

$$\begin{aligned}&\bigg(\min(s, m+n-1) - N + \frac{mnN}{mn-\min(s,m+n-1)}\bigg)\sigma^2\\ &\qquad \leq \text{CRB} \leq \bigg(\min(s, m+n-1) - N\\ &\qquad\qquad + 5\frac{mnN}{mn-\min(s,m+n-1)}\bigg)\sigma^2 \end{aligned} \qquad (11)$$

with probability greater than $1 - 10e^{\frac{-c}{\epsilon^2}}$; where $N = (m+n)r - r^2$, c and ϵ are some constants.

Due to space limits, proofs of Theorem 1 and Corollary 1 are omitted. However, the key steps of the corollary are as follows: First, we apply block matrix inversion formula to the right hand side of (9). Second, we take advantage of the assumptions provided in the corollary statement along with the fact $\mathcal{P}_\mathcal{S} = \boldsymbol{I}_{mn} - \mathcal{P}_{\mathcal{S}^c}$. Finally, we apply a concentration result, that is given in [15], to one of the resulting terms to ultimately achieve the above CRB bounds. The whole proof is provided in extended version [24].

Examining our obtained CRB relative to the old CRB (the one without considering the special structure of the line, in [14], Equation (26) therein), the difference in the final result is the term: $\min(m+n-1, s)$; instead of just s.; which is a result of using a different $\boldsymbol{Q}_s$ matrix. This means we select $(m+n)r - r^2 + \min(m+n-1, s)$ linearly independent rows of $\begin{bmatrix}\boldsymbol{Q}_l & \boldsymbol{Q}_s\end{bmatrix}$ through the operation $\begin{bmatrix}\boldsymbol{A}\boldsymbol{Q}_l & \boldsymbol{A}\boldsymbol{Q}_s\end{bmatrix}$ (see right hand side of (9)); and this requires $\boldsymbol{A} \in \mathbb{R}^{p\times mn}$ with p being sufficiently large to select $(m+n)r - r^2 + \min(m+n-1, s)$ linearly independent rows of $\begin{bmatrix}\boldsymbol{Q}_l & \boldsymbol{Q}_s\end{bmatrix}$. This makes sense since we now limit the choices of $\boldsymbol{S}$ to being not only sparse but also rank-one, due to the line constraint, which renders the choices to be smaller than those of considering sparsity only. This effect of $\boldsymbol{S}$, and accordingly $\boldsymbol{Q}_s$, propagates on the derivations until we reach the bounds provided in (10) and (11). For low sparsity level s, both CRBs approach each other since our new obtained CRB $\min(m+n-1, s)$ is dominated by s, and in turn reduces to the old CRB. On the other hand, as s increases, the term $m+n-1$ dominates, and therefore gives lower values than the old CRB.

IV. Simulations

In this section, ground truth is denoted by $\boldsymbol{X}_0$, and the estimated matrices as $\hat{\boldsymbol{X}}$; normalized recovery errors for $\boldsymbol{L}$ and $\boldsymbol{S}$, respectively, are defined as follows:

$$e_L = \frac{||\boldsymbol{L}_0 - \hat{\boldsymbol{L}}||_F^2}{||\boldsymbol{L}_0||_F^2}, \qquad \text{and} \quad e_S = \frac{||\boldsymbol{S}_0 - \hat{\boldsymbol{S}}||_F^2}{||\boldsymbol{S}_0||_F^2}.$$

We do not consider errors in either $\boldsymbol{R}$ or $\boldsymbol{S_r}$ as they are auxiliary matrices generated to regularize the optimization.

A. CRB Comparisons

In this subsection, we investigate the possible optimality of our method by comparing to the CRB derived in Section III. In this subsection, we are using synthetic data. The noise instances are taken entry-wise from the matrix $\boldsymbol{Z}_0 \sim \mathcal{N}(0, \sigma^2)$. Matrices $\boldsymbol{X}, \boldsymbol{L}_0$, $\boldsymbol{S}_0$, and $\boldsymbol{Z}_0$ are $m \times n = 120 \times 300$. The matrix $\boldsymbol{L}_0$ is low-rank (with rank $r = 5$), and is generated as $\boldsymbol{L}_0 = \boldsymbol{U}\boldsymbol{V}^T$, where $\boldsymbol{U}$ is of size $m \times r$, and $\boldsymbol{V}$ is of size $n \times r$. Both $\boldsymbol{U}$ and $\boldsymbol{V}$ are matrices with i.i.d $\mathcal{N}(0, \epsilon\sigma^2)$, where ϵ is chosen arbitrarily to make the singular values of $\boldsymbol{L}_0$ are much larger than those of $\boldsymbol{Z}_0$ (we set ϵ to 10). The entries of $\boldsymbol{S}_0$ are independently distributed, each taking on value 0 with probability $p = 0.8$, and object values are generated with probability $1-p = 0.2$, with an arbitrarily fixed preselected value that represents the intensity of the object. Each run of a simulation (total of 10^5 runs) has randomly generated velocities drawn from a discrete uniform distribution [-10,10]. To generate a linear trajectory, the equation that describes the motion from position (i,j) to position (k,l) with velocity $v_{kl,ij}$ is as follows: $(k,l) = (i,j) + v_{kl,ij}$. This equation is translated into two equations for x and y axes, respectively, as follows: $k = i + v_{kl,ij}|\cos(\gamma_{kl,ij})|$, and $l = j + v_{kl,ij}|\sin(\gamma_{kl,ij})|$; where $\gamma_{kl,ij} = \tan^{-1}\Big((l-j)/(k-i)\Big)$. The direction of the motion (positive or negative) is considered in the sign of $v_{kl,ij}$. The matrix $\boldsymbol{X}$ is normalized: $\bar{\boldsymbol{X}} = \boldsymbol{X}/||\boldsymbol{X}||_F^2$. The SNR is defined as $||\bar{\boldsymbol{X}}||_F^2/\sigma^2$; as a result, the noise variance is then $1/\text{SNR}$.

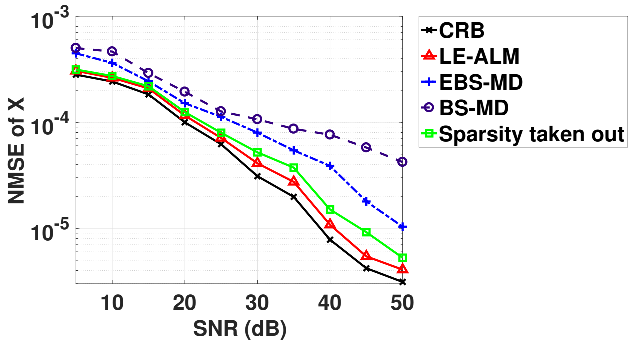

Figure 2. Comparison of ALL techniques with CRB for different SNR values.

Figure 3. Frame 136.

Figure 4. Frame 136 with noise (SNR=30 dB).

Figure 5. Ideal Scenario.

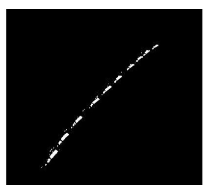

Figure 6. SNR = 30 dB.

Fig. 2 shows the NMSE error for the matrix $\boldsymbol{X}$ for our method in comparison with both background subtraction methods as well as the derived CRB. From now on, we denote the method of [5] as Efficient Background Subtraction based on Matrix Decomposition (**EBS-MD**) and [4] as Background Subtraction based on Matrix Decomposition (**BS-MD**), where the abbreviation **BS** is to emphasize the fact that both methods are based on *background subtraction*; however, as they do not exploit the side information of object tracking on a line as we do, we expect performance improvement relative to these schemes, as seen in Fig. 2. Moreover, **LE-ALM** is the closest to the derived CRB, with only a deviation of 1.2 dB, on average. Furthermore, motivated by the work in [19] that using multiobjective optimization with multiple structures can do no better than exploiting only one of the structures, we take out the sparsity objective $\left(i.e.\ \lambda_1\|\boldsymbol{S}\|_{2,1}\right)$ from the multiobjective optimization given in (1), and see how the performance is accordingly affected. As can be seen from Fig. 2, the performance; however, degrades, verifying that considering multiobjective optimization, in regard to our line estimation problem, does better than exploiting only a single structure.

B. Real Data

Herein, we apply our method to real video data from [25], [26]. In [25], [26], there is a video of two moving boats following nearly linear trajectories. The following pre-processing was necessary in order to apply our method. Specifically, we extracted the segment between seconds 51 and 72, which provides an approximate straight line of movement; see [27]. This part of the video is converted into a sequence of 641 frames (images). Each frame is of size 741×1920. Since we are detecting a single object; that is represented by the large boat, the contribution due to the other small boat has been removed from the frames.

Hence, we have now frames of size 741×900. We used 10 frames. So, we have matrices of size $(741 \times 900) \times 10 = 6669000$. These matrices are the matrices used in simulations to estimate the matrices $\boldsymbol{L}$ and $\boldsymbol{S}$. Since these matrices are too much large to display as images, what we do display as images are the ten images added together in one matrix of size 741×900. To help the reader visualize the process, frame 136 is shown in Fig. 3.

We tested our technique for different values of synthetic noise represented through the parameter SNR (as defined in previous subsection). For convenience, Fig. 4 shows the noisy version of frame 136 for SNR=30 dB. Fig. 5 represents the ideal scenario; whereas Fig. 6, as an example, is the estimated sequences for SNR=30 dB. The white objects represent the moving object for the ten frames stacked on top of each other. Table I summarizes the NMSE of the recovered background matrix and the recovered sparse object matrix for various SNR values. As can be seen from the table, the proposed **LE-ALM** method provides strong estimation results and it has very small error even at low SNR values.

Table I
NMSE for $\boldsymbol{L}$ and $\boldsymbol{S}$ for various SNR values

SNR (in dB)	e_L	e_S
10	5.97×10^{-5}	4.83×10^{-4}
20	3.92×10^{-5}	2.21×10^{-4}
30	1.69×10^{-5}	1.24×10^{-4}
40	0.85×10^{-5}	0.62×10^{-4}
50	0.013×10^{-5}	0.18×10^{-4}

V. Conclusions

In this paper, we considered our previous extension of structured estimation beyond low rank and sparsity to finding "lines" in matrices. This generalization has been applied to the tracking of a single object in a real video that has been sampled and viewed as a sequence of multiple images. Our method of estimation has showed its superiority over state-of-the-art background subtraction methods that do not exploit the additional structure. Moreover, a new CRB, that matches the new presented problem, has been derived. We investigated that our estimation error is very close to the computed CRB with, on average, only deviation of 1.2 dB, suggesting the near-optimality of our proposed method.

Acknowledgments

This research has been funded in part by one or more of the following grants: ONR N00014-15-1-2550, NSF CNS-1213128, NSF CCF-1718560, NSF CCF-1410009, NSF CPS-1446901, and AFOSR FA9550-12-1-0215. We also would like to thank the USC Robotic Embedded Systems Lab: Hordur Heidarsson and Stephanie Kemna for providing us with the video data used herein.

References

[1] Y. Zhang, Z. Jiang, and L. S. Davis, "Learning structured low-rank representations for image classification," in *Computer Vision and Pattern Recognition, 2013 IEEE Conference on*, pp. 676–683, June 2013.

[2] H. K. Aggarwal and A. Majumdar, "Multi-spectral demosaicing: A joint-sparse elastic-net formulation," in *Advances in Pattern Recognition (ICAPR), 2015 Eighth International Conference on*, pp. 1–5, Jan 2015.

[3] O. Oreifej, X. Li, and M. Shah, "Simultaneous video stabilization and moving object detection in turbulence," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, pp. 450–462, Feb. 2013.

[4] X. Liu, G. Zhao, J. Yao, and C. Qi, "Background subtraction based on low-rank and structured sparse decomposition," *IEEE Transactions on Image Processing*, vol. 24, pp. 2502–2514, Aug 2015.

[5] S. E. Ebadi, V. G. Ones, and E. Izquierdo, "Efficient background subtraction with low-rank and sparse matrix decomposition," in *2015 IEEE International Conference on Image Processing (ICIP)*, pp. 4863–4867, Sept 2015.

[6] B. Lois and N. Vaswani, "Online matrix completion and online robust pca," in *2015 IEEE International Symposium on Information Theory (ISIT)*, pp. 1826–1830, June 2015.

[7] A. Elnakeeb and U. Mitra, "Low-rank, sparse and line constrained estimation: Applications to target tracking and convergence," in *2017 IEEE International Symposium on Information Theory (ISIT)*, pp. 2283–2287, June 2017.

[8] S. Beygi and U. Mitra, "Multi-scale multi-lag channel estimation using low rank approximation for ofdm," *IEEE Transactions on Signal Processing*, vol. 63, pp. 4744–4755, Sept 2015.

[9] S. Beygi, U. Mitra, and E. G. Ström, "Nested sparse approximation: Structured estimation of v2v channels using geometry-based stochastic channel model," *IEEE Transactions on Signal Processing*, vol. 63, pp. 4940–4955, Sept 2015.

[10] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Comput. Surv.*, vol. 38, Dec. 2006.

[11] X. R. Li and V. P. Jilkov, "Survey of maneuvering target tracking. part i. dynamic models," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, pp. 1333–1364, Oct 2003.

[12] F. Chen, Q. Wang, S. Wang, W. Zhang, and W. Xu, "Object tracking via appearance modeling and sparse representation," *Image Vision Comput.*, vol. 29, pp. 787–796, Oct. 2011.

[13] S. El Mejdani, R. Egli, and F. Dubeau, "Old and new straight-line detectors: Description and comparison," *Pattern Recogn.*, vol. 41, pp. 1845–1866, June 2008.

[14] G. Tang and A. Nehorai, "Constrained Cramer-Rao bound on robust principal component analysis," *IEEE Transactions on Signal Processing*, vol. 59, pp. 5070–5076, Oct 2011.

[15] G. Tang and A. Nehorai, "Lower bounds on the mean-squared error of low-rank matrix reconstruction," *IEEE Transactions on Signal Processing*, vol. 59, pp. 4559–4571, Oct 2011.

[16] P. Stoica and B. C. Ng, "On the Cramer-Rao bound under parametric constraints," *IEEE Signal Processing Letters*, vol. 5, pp. 177–179, July 1998.

[17] X. Ding, L. He, and L. Carin, "Bayesian robust principal component analysis," *IEEE Transactions on Image Processing*, vol. 20, pp. 3419–3430, Dec 2011.

[18] J. D. Gorman and A. O. Hero, "Lower bounds for parametric estimation with constraints," *IEEE Transactions on Information Theory*, vol. 36, pp. 1285–1301, Nov 1990.

[19] S. Oymak, A. Jalali, M. Fazel, Y. C. Eldar, and B. Hassibi, "Simultaneously structured models with application to sparse and low-rank matrices," *IEEE Transactions on Information Theory*, vol. 61, pp. 2886–2908, May 2015.

[20] M. Tao and X. Yuan, "Recovering low-rank and sparse components of matrices from incomplete and noisy observations," *SIAM J. on Optimization*, vol. 21, pp. 57–81, Jan. 2011.

[21] Z. Lin, M. Chen, and Y. Ma, "The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices," *in UIUC Technical Report*, 2009.

[22] Z. Lin, R. Liu, and Z. Su, "Linearized alternating direction method with adaptive penalty for low-rank representation," in *Advances in Neural Information Processing Systems 24* (J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger, eds.), pp. 612–620, Curran Associates, Inc., 2011.

[23] Z. Lin, R. Liu, and H. Li, "Linearized alternating direction method with parallel splitting and adaptive penalty for separable convex programs in machine learning," *Machine Learning*, vol. 99, pp. 287–325, May 2015.

[24] "Full version paper, link: https://www.dropbox.com/s/7489qy2trdya7yk/papercrb.pdf?dl=0."

[25] S. Bhattacharya, S. Kim, H. Heidarsson, G. S. Sukhatme, and V. Kumar, "A topological approach to using cables to separate and manipulate sets of objects," *The International Journal of Robotics Research*, vol. 34, no. 6, pp. 799–815, 2015.

[26] "Full video, link: https://www.dropbox.com/s/adj31afj7ald5w3/GP040018_.MP4?dl=0."

[27] "Cropped video, link: https://www.dropbox.com/s/9hplw3y8fn48iuh/video3.mp4?dl=0."