Practical ReProCS for Separating Sparse and Low-dimensional Signal Sequences from their Sum

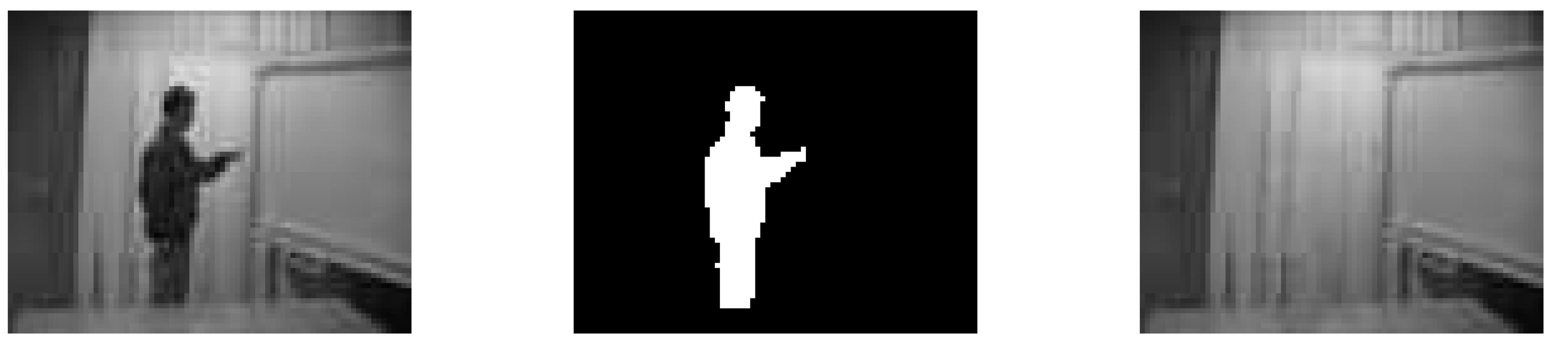
Han Guo, Chenlu Qiu and Namrata Vaswani
Iowa State University

IOWA STATE UNIVERSITY

Introduction

- **Goal:** recovering a time sequence of sparse vectors $\mathbf{S_t}$ and a time sequence of dense vectors $\mathbf{L_t}$ from their sum, $\mathbf{M_t := S_t + L_t}$, when any subsequence of the $\mathbf{L_t}$'s lies in a slowly changing low-dimensional subspace.
- **Key application: video layering.** Background scene can be assumed low-rank and the foreground is usually sparse.

Orig. **Foreground** **Background**



- **Contributions:**
  - We designs a practical online algorithm, **Recursive Projected Compressive Sensing(ReProCS)**, which requires fewer parameters and exploits practically valid assumptions;
  - We show via extensive simulation and real video experiments that ReProCS is more robust to correlated support change of $\mathbf{S_t}$ than many existing work.

Problem Definition

- The measurement vector at time $\mathbf{t}$, $\mathbf{M_t}$, can be decomposed as $\mathbf{M_t = S_t + L_t}$, where $\mathbf{S_t}$ is a sparse vector and $\mathbf{L_t}$ is a dense but low-dimensional vector.
- Given an initial sequence which does not contain the sparse components, we are able to get an initial subspace.
- Our goal is to recursively estimate $\mathbf{S_t}$ and $\mathbf{L_t}$ and the subspace in which the last several $\mathbf{L_t}$'s lie at each $\mathbf{t > t_{train}}$.

Basic Assumptions

- **Low-dimensionality and slow subspace change**
  We let $\mathbf{L_t = P_{(t)}a_t}$ where $\mathbf{P_{(t)}}$ is piecewise constant with time, i.e. $\mathbf{P_{(t)} = P_j}$ for all $\mathbf{t \in [t_j, t_{j+1})}$, and $\mathbf{r_j} = \text{rank}(\mathbf{P_j}) \ll \mathbf{(t_{j+1} - t_j)}$. A simple model for slow subspace change is to let $\mathbf{P_j}$ change as
  $$\mathbf{P_j = [(P_{j-1}R_j \setminus P_{j,old}), P_{j,new}]}$$
  where $\mathbf{R_j}$ is a rotation matrix. Moreover, the projection of $\mathbf{L_t}$ along $\mathbf{P_{j,new}}$ is small initially for the first α frames, i.e.
  $$\|\mathbf{(I - P_{j-1}P_{j-1}')L_t}\|_2 \ll \min(\|\mathbf{L_t}\|_2, \|\mathbf{S_t}\|_2) \text{ if } \mathbf{t \in [t_j, t_j + \alpha)}$$
  and can increase gradually after $\mathbf{t_j} + \alpha$.
  Why it is valid: Background images typically change only a little over time.

  - Verification method and description of videos can be found in [1]
  - As shown, after every subspace change time $\mathbf{(t_j = 725, 1450)}$, $\|\mathbf{(I - P_{j-1}P_{j-1}')L_t}\|_2 / \|\mathbf{L_t}\|_2$ is initially small.

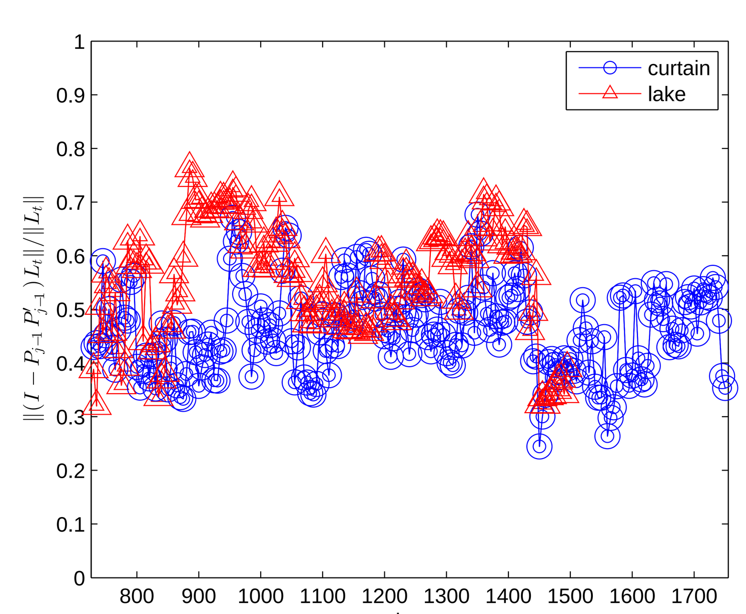


Figure 1: Verification of slow subspace assumption.

- **Denseness**
  We assume that the subspace spanned by the $\mathbf{L_t}$'s is dense, i.e.
  $$\kappa_{2s}(\mathbf{P_j}) = \kappa_{2s}([\mathbf{L_{t_j}}, \dots \mathbf{L_{t_{j+1}-1}}]) \leq \kappa_*$$
  for a κ_* significantly smaller than one. Here
  $$\kappa_s(\mathbf{B}) = \kappa_s(\text{range}(\mathbf{B})) := \max_{|\mathbf{T}| \leq s} \|\mathbf{I_T}'\text{basis}(\mathbf{B})\|_2$$
  is the denseness coefficient for any vector or matrix $\mathbf{B}$.
  Why it is valid: Very often, the background images primarily change due to lighting changes (indoor), moving waters or moving leaves (outdoor).
- **Support size, Support change of $\mathbf{S_t}$**
  - Either the support size is small or the support changes are slow or both.
  - There is *some* support change during any set of α frames.

  Why it is valid: foreground images typically consist of one or more moving objects/regions and hence are sparse. Also, typically the objects are not static.

Our Algorithm: ReProCS

Given the initial training sequence which does not contain the sparse components, $\mathcal{M}_{\text{train}} = [\mathbf{L_1, L_2, \dots L_{t_{train}}}]$, compute $\hat{\mathbf{P}}_0$ as an approximate basis for $\mathcal{M}_{\text{train}}$. After this, at each time $\mathbf{t}$, ReProCS involves 4 steps:

- **Perpendicular Projection.** Project the measurement vector, $\mathbf{M_t}$, into the space orthogonal to $\hat{\mathbf{P}}_{(t-1)}$ to get $\mathbf{y_t := \Phi_{(t)}M_t}$, where $\mathbf{\Phi_{(t)} := (I - \hat{P}_{(t-1)}\hat{P}_{(t-1)}')}$.
- **Sparse Recovery (Recover $\mathbf{T_t}$ and $\mathbf{S_t}$).** With the above projection,
  $$\mathbf{y_t := \Phi_{(t)}S_t} + \beta_t,$$
  where $\|\beta_t\|_2 = \|\mathbf{\Phi_{(t)}L_t}\|_2$ is small. To recover $\mathbf{S_t}$ from $\mathbf{y_t}$, solve
  $$\min_{\mathbf{x}} \|\mathbf{x}\|_1 \text{ s.t. } \|\mathbf{y_t - \Phi_{(t)}x}\|_2 \leq \xi.$$
  The support set $\hat{\mathbf{T}}_t$ is obtained by thresholding on the solution, $\hat{\mathbf{S}}_{t,cs}$. By computing a least squares (LS) estimate on $\hat{\mathbf{T}}_t$, we can get a more accurate estimate, $\hat{\mathbf{S}}_t$.
- **Recover $\mathbf{L_t}$.** The estimate $\hat{\mathbf{S}}_t$ is used to estimate $\mathbf{L_t}$ as $\hat{\mathbf{L}}_t = \mathbf{M_t} - \hat{\mathbf{S}}_t$.
- **Subspace Update (Update $\hat{\mathbf{P}}_{(t)}$).** We update the subspace every some frames. Usage of simple PCA, Proj-PCA are discussed in [1]

Results: Partly Simulated Data (real background, simulated foreground)

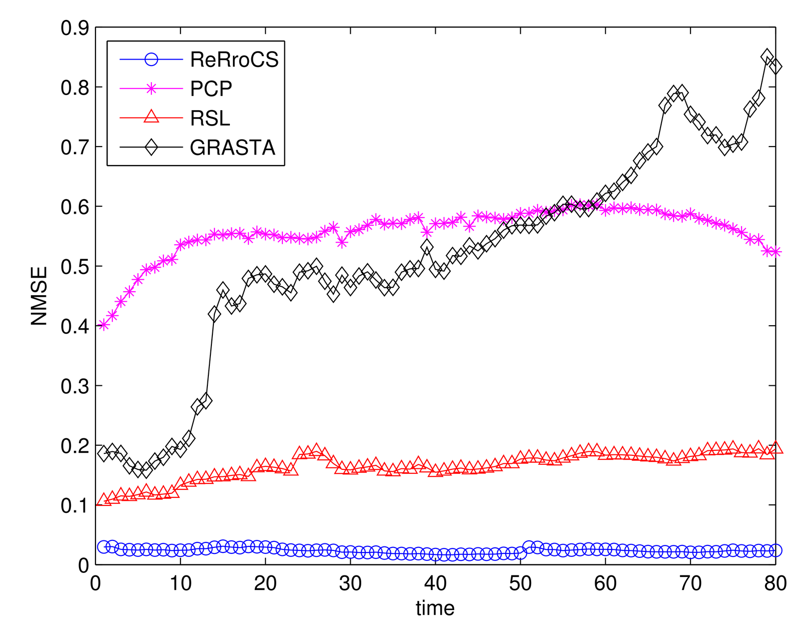


Figure 2: NMSE for recovering $\mathbf{S_t}$.

- We generated 50 realizations of this video sequence and compared all the algorithms to estimate $\mathbf{S_t}$.
- Normalized mean squared error (NMSE) in recovering $\mathbf{S_t}$ is shown in Figure 2, and visual comparisons for one realization is shown in Figure 3.

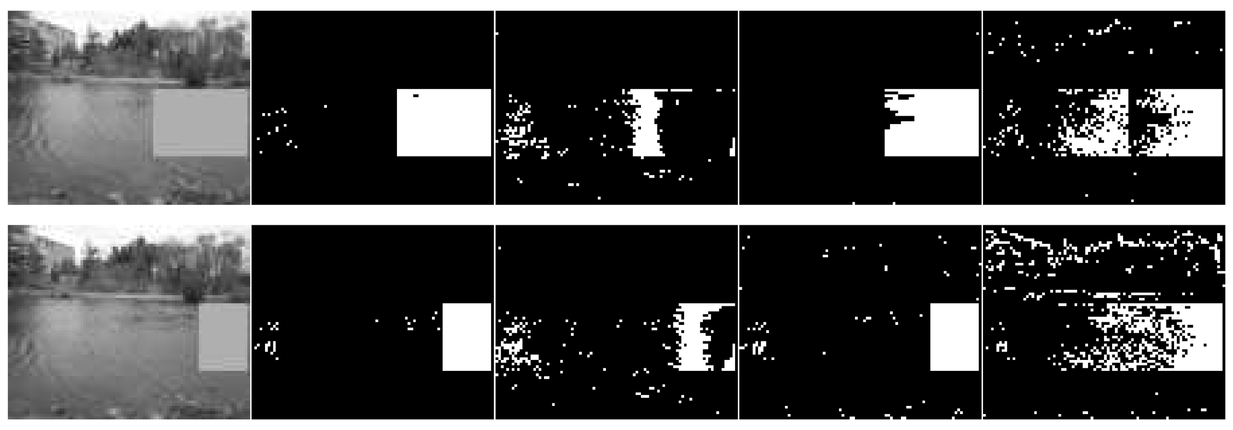


Figure 3: First column: original video frames at $\mathbf{N = t_{train} + t, t = 60, 70}$. Next columns: foreground layer estimated by ReProCS, PCP, RSL, and GRASTA.

Results: Real Video Data

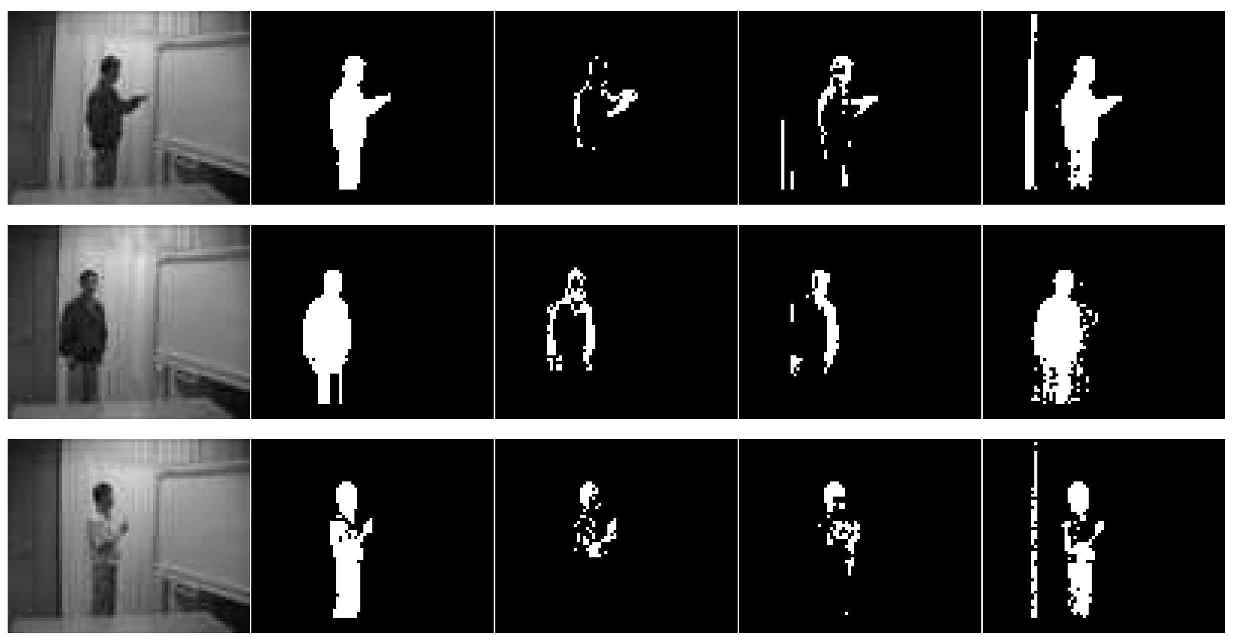


Figure 4: First column: original video frames at $\mathbf{N = t_{train} + t, t = 60, 120, 475}$. Next columns: foreground layer estimated by ReProCS, PCP, RSL, and GRASTA.

- This video is challenging because the background variations are quite large, and the white shirt color and the curtain's color is similar.
- As can be seen in Figure 4, ReProCS's performance is significantly better than that of the other algorithms.

Selected References

H. Guo, C. Qiu, and N. Vaswani,
"An online algorithm for separating sparse and low-dimensional signal sequences from their sum,"
arXiv:1310.4261v2, submitted to IEEE Trans. Signal Processing.

{hanguo, chenlu, namrata}@iastate.edu