Fast and Provable Nonconvex Tensor Robust PCA

Yao Wang

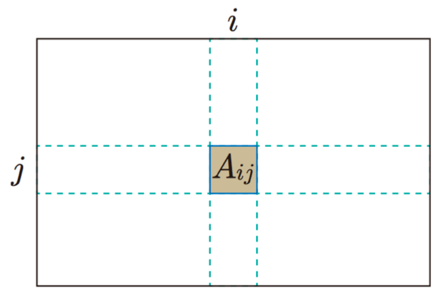
Center for Intelligent Decision-making and Machine Learning

Xi’an Jiaotong University

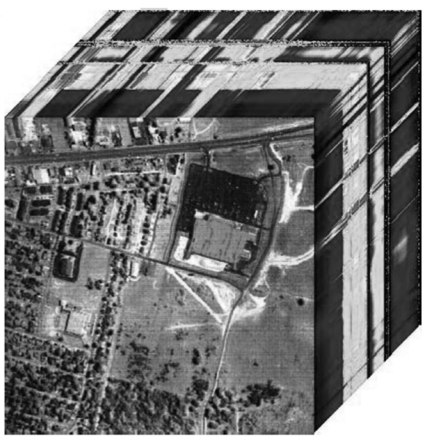
August 2022

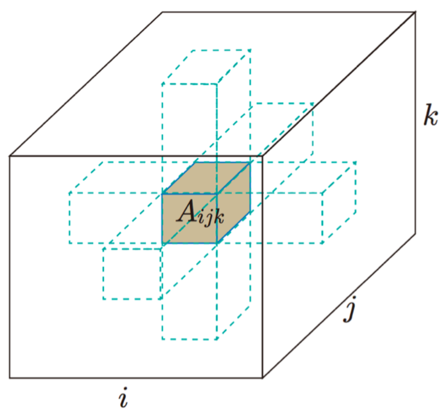
Matrix vs Tensor

- Image: $X = (X_{ij}) \in \mathbb{R}^{n_1 \times n_2}$
- Hyperspectral: $\mathcal{X} = (\mathcal{X}_{ijk}) \in \mathbb{R}^{n_1 \times n_2 \times d}$



2nd order relation

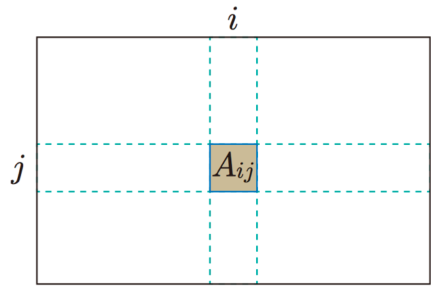




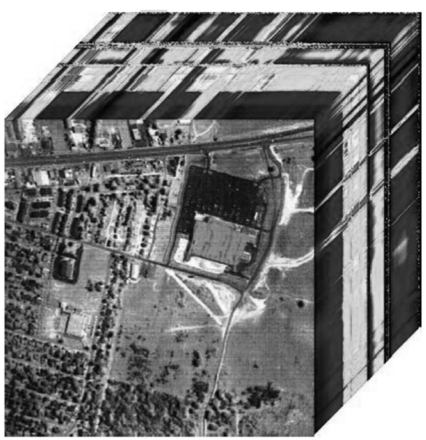
3rd order relation

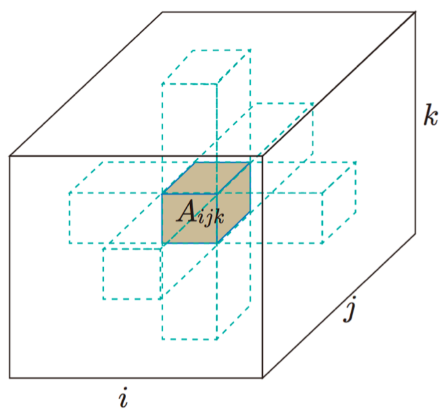
Matrix vs Tensor

- Image: $X = (X_{ij}) \in \mathbb{R}^{n_1 \times n_2}$
- Hyperspectral: $\mathcal{X} = (\mathcal{X}_{ijk}) \in \mathbb{R}^{n_1 \times n_2 \times d}$

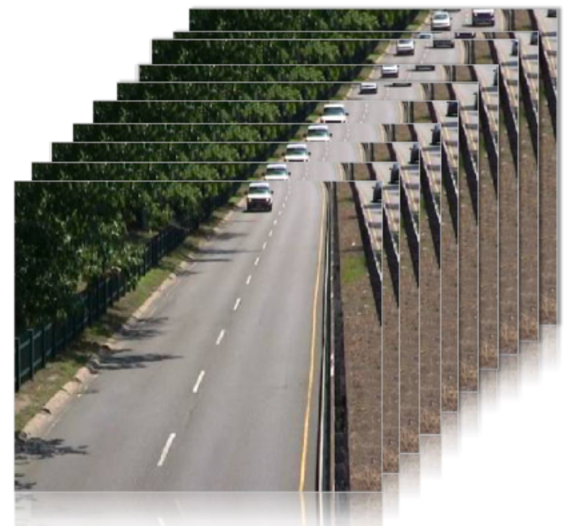


2nd order relation



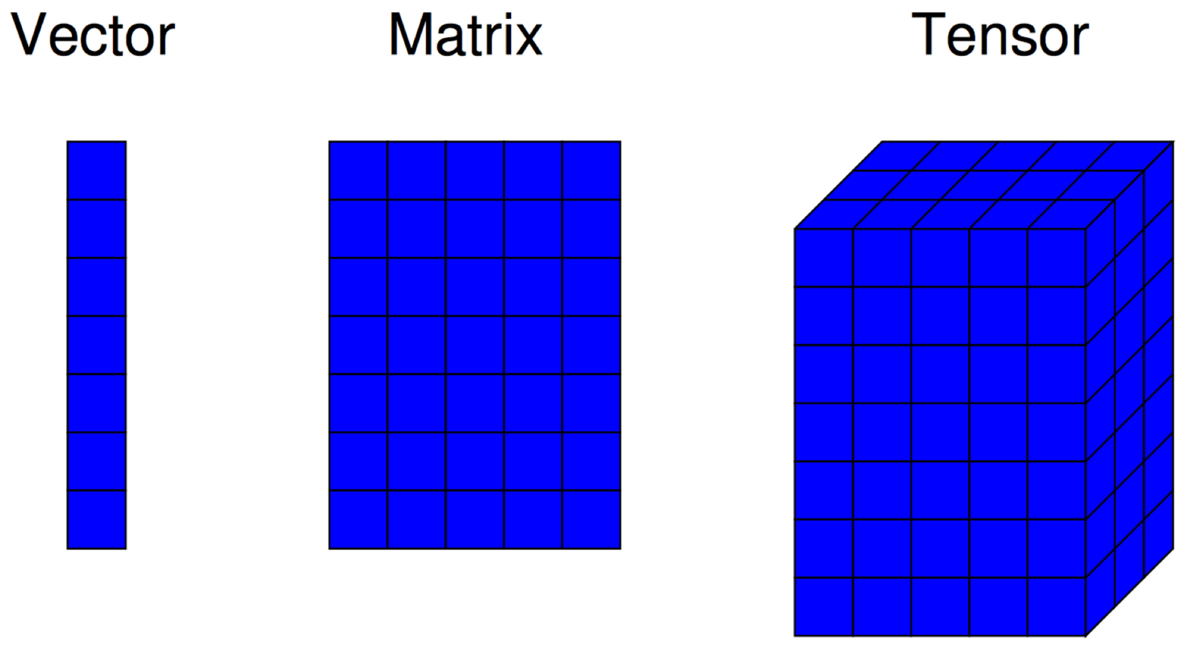


3rd order relation



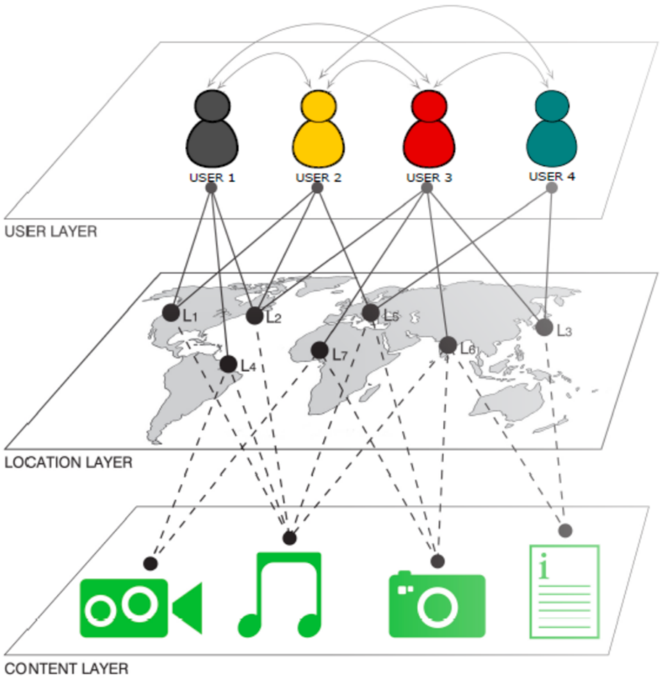
$\mathcal{X} = (\mathcal{X}_{ijkl}) \in \mathbb{R}^{n_1 \times n_2 \times 3 \times t}$

Unified Perspective



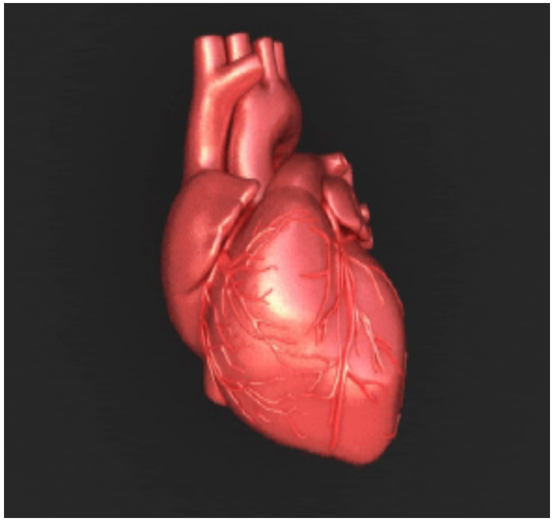
- scalar = tensor of order 0
- (column) vector = tensor of order 1
- matrix = tensor of order 2
- tensor of order 3
  $= n_1 n_2 n_3$ numbers arranged in $n_1 \times n_2 \times n_3$ array

Tensor Data Everywhere



Recommendation

Surveillance



Fast Imaging

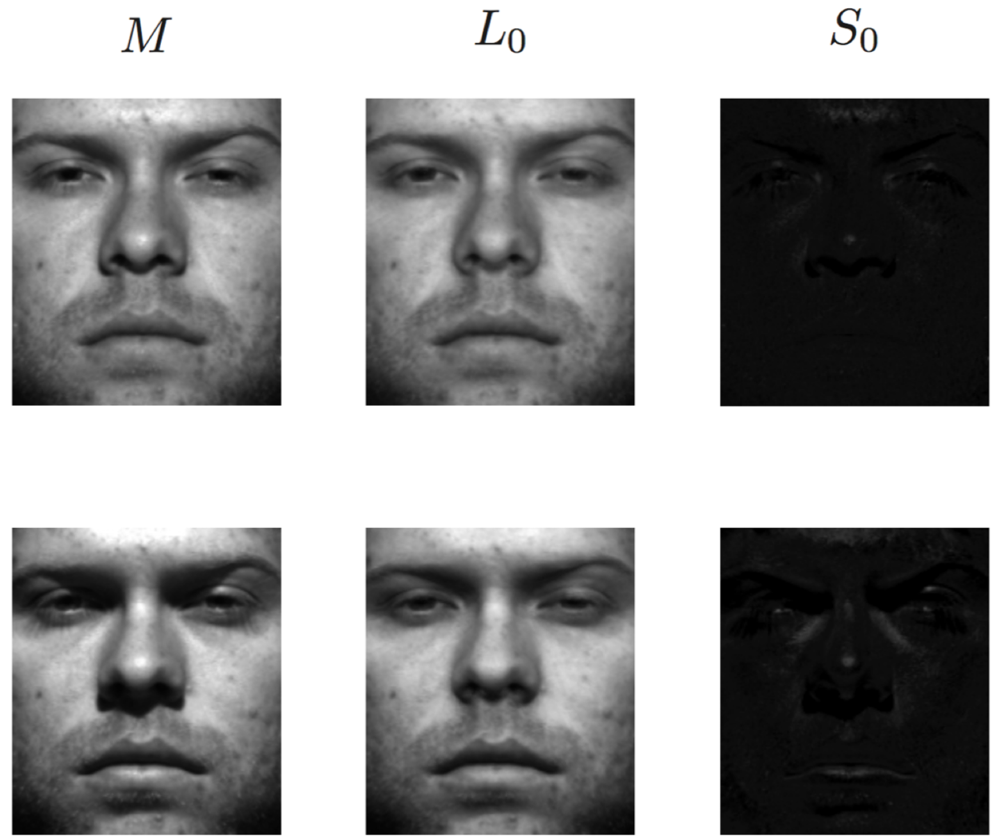
How to extract ***compact knowledge*** *from such datasets in complex scenes?*

Motivating Examples: Video Surveillance

(a) Original frames (b) Low-rank $\hat{L}$ (c) Sparse $\hat{S}$

- sequence of 200 frames each with resolution 176×144

Motivating Examples: Removing Face Illumination



- Yale B faces database, 192×168 images of a subject under 58 different illuminations.

L+S Matrices Decomposition

Robust Principle Component Analysis (RPCA)

- recover a low-rank matrix L^* from highly corrupted observations

$M = L^* + S^*$ where S^* is a sparse matrix

- optimization objective

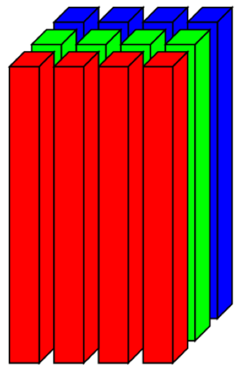
$$\min_{L,S} \left|\left|M - L - S\right|\right|_F^2$$
$$s.t. \quad rank(L) \leq r,$$
$$\left|\left|S\right|\right|_0 \leq M.$$

  - convex surrogate (*Candès+Li+Ma+Wright '12*)

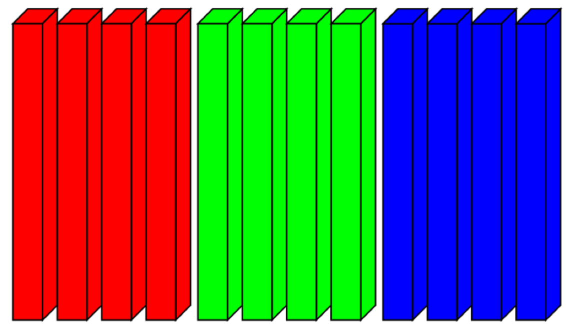
$$\min_{L,S} \left|\left|L\right|\right|_* + \lambda \left|\left|S\right|\right|_1$$
$$s.t. \quad M = L + S$$

Matricization

Stack vectors into an $n_1 \times (n_2 \cdots n_d)$ matrix:



$\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_d}$



$X^{(1)} \in \mathbb{R}^{n_1 \times (n_2 n_3 \cdots n_d)}$

For $\mu = 1, \ldots, d$, the μ-mode matricization of $\mathcal{X}$ is a matrix

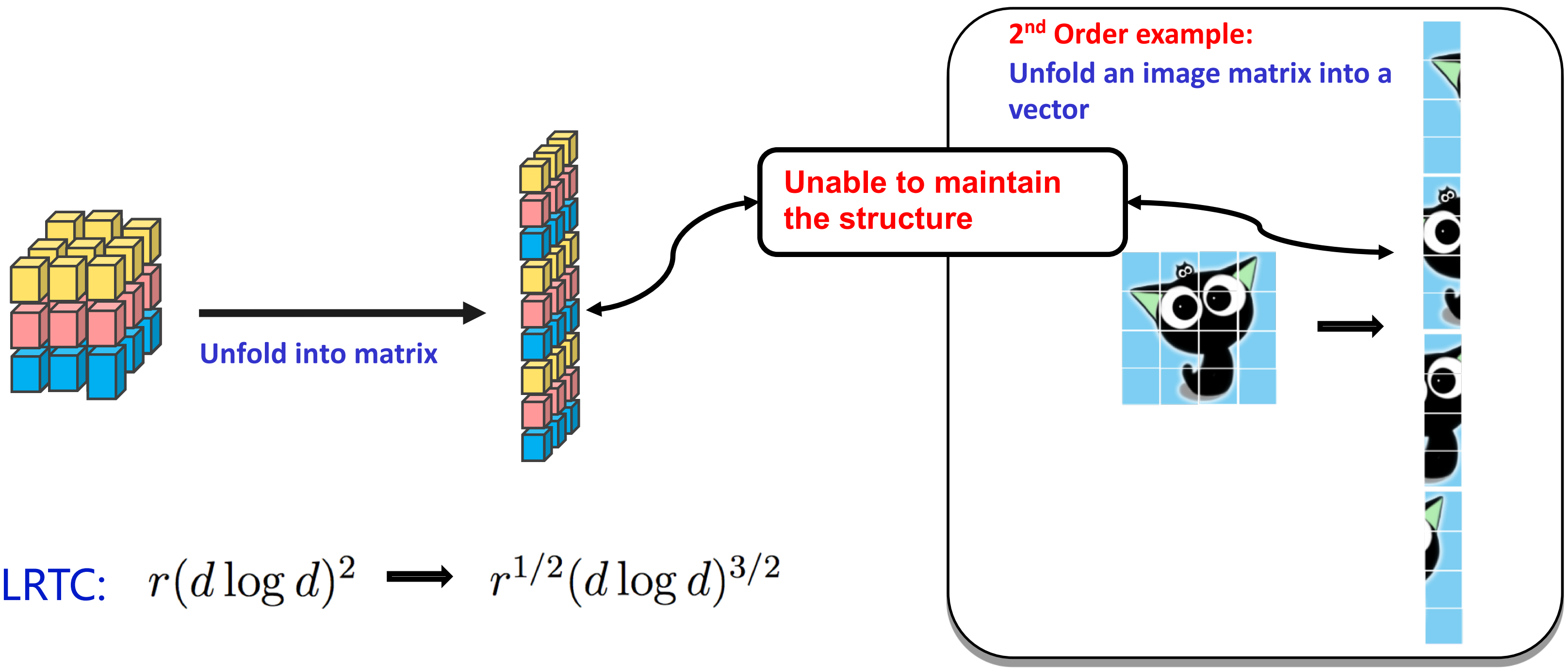
$$X^{(\mu)} \in \mathbb{R}^{n_\mu \times (n_1 \cdots n_{\mu-1} n_{\mu+1} \cdots n_d)}$$

with entries

$$\left(X^{(\mu)}\right)_{i_{\mu_1},(i_1,\ldots,i_{\mu-1},i_{\mu+1}\ldots i_d)} = \mathcal{X}_i \quad \forall i \in \mathfrak{I}.$$

Lower-order Methodology is Inappropriate

- Any high-order tensor problems can be treated by lower-order methodologies.
- The unfolding sometimes may lead to the loss of useful structures of tensors [Yuan and Zhang, 2016, 2017; Mu et al., 2015]



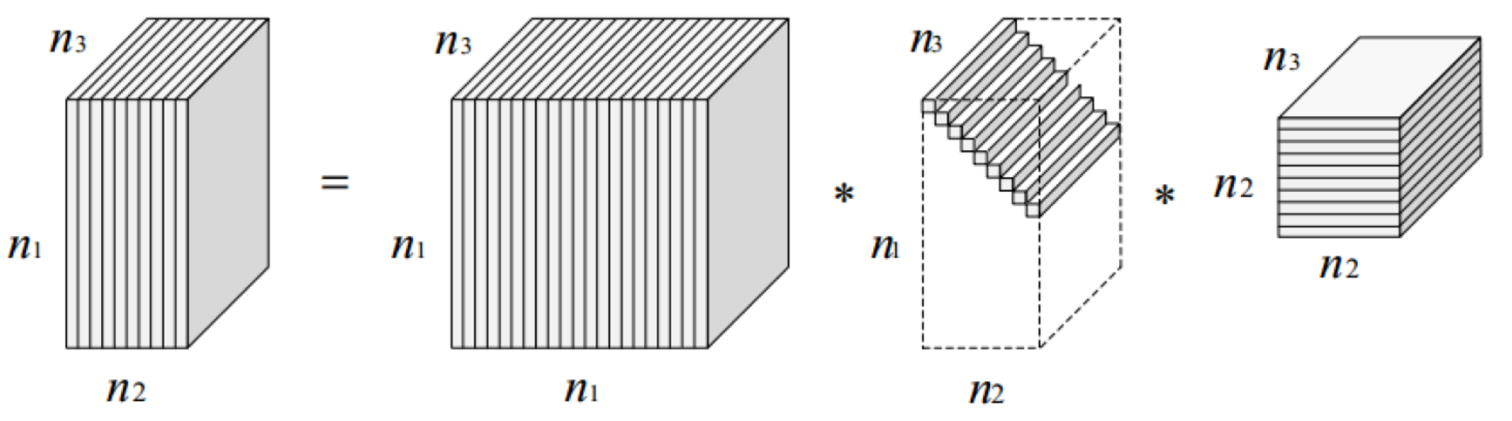
LRTC: $r(d \log d)^2 \longrightarrow r^{1/2}(d \log d)^{3/2}$

t-Product and t-SVD

$$\mathcal{A} * \mathcal{B} = \texttt{fold}(\texttt{bcirc}(\mathcal{A}) \cdot \texttt{unfold}(\mathcal{B}))$$

$$\texttt{bcirc}(\mathcal{A}) = \begin{bmatrix} A^{(1)} & A^{(n_3)} & \cdots & A^{(2)} \\ A^{(2)} & A^{(1)} & \cdots & A^{(3)} \\ \vdots & \vdots & \ddots & \vdots \\ A^{(n_3)} & A^{(n_3-1)} & \cdots & A^{(1)} \end{bmatrix} \quad \texttt{unfold}(\mathcal{A}) = \begin{bmatrix} A^{(1)} \\ A^{(2)} \\ \vdots \\ A^{(n_3)} \end{bmatrix} \quad A^{(i)} := \mathcal{A}(:,:,i)$$

$$\mathcal{A} = \mathcal{U} * \mathcal{S} * \mathcal{V}^*$$



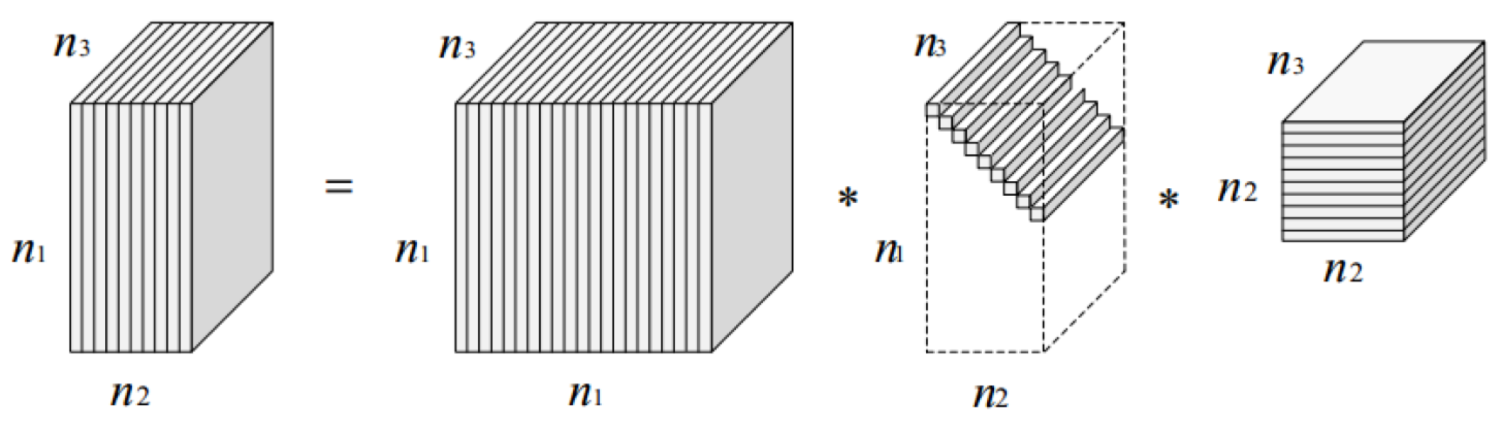
Kilmer and Martin, 2011'; 2015'; 2021'

t-Product and t-SVD

$$\mathcal{A} * \mathcal{B} = \texttt{fold}(\texttt{bcirc}(\mathcal{A}) \cdot \texttt{unfold}(\mathcal{B}))$$

$$\texttt{bcirc}(\mathcal{A}) = \begin{bmatrix} A^{(1)} & A^{(n_3)} & \cdots & A^{(2)} \\ A^{(2)} & A^{(1)} & \cdots & A^{(3)} \\ \vdots & \vdots & \ddots & \vdots \\ A^{(n_3)} & A^{(n_3-1)} & \cdots & A^{(1)} \end{bmatrix} \quad \texttt{unfold}(\mathcal{A}) = \begin{bmatrix} A^{(1)} \\ A^{(2)} \\ \vdots \\ A^{(n_3)} \end{bmatrix} \quad A^{(i)} := \mathcal{A}(:,:,i)$$

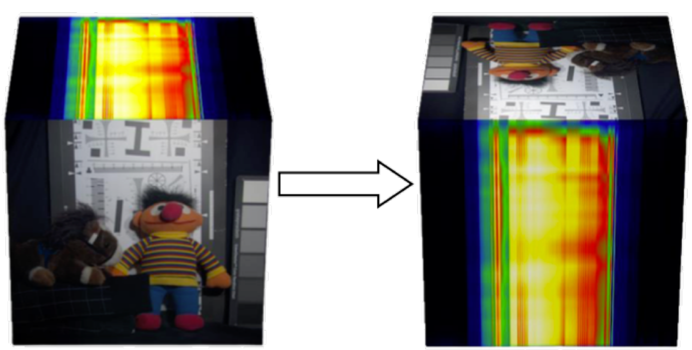
$$\mathcal{A} = \mathcal{U} * \mathcal{S} * \mathcal{V}^*$$



One can compute T-SVD efficiently in the Fourier domain!

Motivation

- Tensor decomposition
  - Transformed t-SVD is used for our method, because it can model tensor in both the time domain and frequency domain
    - Time domain
      - equivalent to the low rank of tensor's mode-1 unfolding matrix
    - Frequency domain
      - noises are concentrated in high frequency slices
  - Other decompositions such as CP / Tucker cannot model tensor in both time and frequency domains and do not have trackable best low rank approximation



- right tensor has lower rank than the left tensor
- noise will be transformed into high frequency term in frequency domain
- applying smaller rank to high frequency term will further improve performance

Tensor RPCA

- Recover a low-rank tensor $\mathcal{L}^* \in \mathbb{R}^{n\times n\times q}$ from highly corrupted measurements $\mathcal{D} = \mathcal{L}^* + \mathcal{S}^*$ where $\mathcal{S}^*$ is a sparse tensor

$$\min_{\mathcal{L}\in\mathbb{L},\mathcal{S}\in\mathbb{S}} \|\mathcal{D}-\mathcal{L}-\mathcal{S}\|_F^2,$$

$$\text{subject to } \begin{cases} \mathbb{L} \equiv \{\mathcal{X} \mid \text{rank}_m(\mathcal{X}) \le \boldsymbol{r}\}, \\ \mathbb{S} \equiv \{\mathcal{X} \mid \|\mathcal{X}\|_0 \le K\} \end{cases}$$

- Convex method based on Transformed t-SVD with recovery guarantee (Lu, et al. 2019';2021')

$$\min_{\mathcal{L},\mathcal{S}} \|\mathcal{L}\|_{\Phi^*} + \lambda\|\mathcal{S}\|_1, \text{subject to } \mathcal{D} = \mathcal{L}+\mathcal{S}.$$

Alternating **P**rojection for **TRPCA**

Optimization objective

$$\min_{\mathcal{L}\in\mathbb{L},\mathcal{S}\in\mathbb{S}} \|\mathcal{D}-\mathcal{L}-\mathcal{S}\|_F^2,$$

$$\text{subject to } \begin{cases} \mathbb{L} \equiv \{\mathcal{X} \mid \text{rank}_m(\mathcal{X}) \le \boldsymbol{r}\}, \\ \mathbb{S} \equiv \{\mathcal{X} \mid \|\mathcal{X}\|_0 \le K\} \end{cases}$$

- APT alternatively projects between low-rank space $\mathbb{L}$ and sparse space $\mathbb{S}$

Algorithm 2 APT: Alternating Projection Algorithm for Tensor RPCA

1: Run Algorithm 1 for initialization
2: **for** $k = 0$ to $T-1$ **do**
3: $\mathcal{L}_{k+1} = P_{\mathbb{L}}(\mathcal{D}-\mathcal{S}_k)$
4: $\mathcal{S}_{k+1} = T_{\zeta_{k+1}}(\mathcal{D}-\mathcal{L}_{k+1})$ where $\zeta_{k+1} = \beta\gamma^k\bar{\sigma}_1(\mathcal{D}-\mathcal{S}_k)$
5: **end for**
6: **Return:** $\mathcal{L}_T$ and $\mathcal{S}_T$

Low-rank space projection

- truncated t-SVD (**high computational complexity**)

$$\mathcal{L}_{k+1} = P_{\mathbb{L}}(\mathcal{D}-\mathcal{S}_k)$$

Theorem B.1. *If $\mathcal{X} = \mathcal{U} \diamond_\Phi \mathcal{S} \diamond_\Phi \mathcal{V}^H \in \mathbb{C}^{m\times n\times q}$ is the transformed t-SVD of $\mathcal{X}$. Define $H_{\boldsymbol{r}}(\mathcal{X})$ to be the approximation having multi-rank $\boldsymbol{r}$: that is,*

$$\hat{\mathcal{X}}_\Phi(:,:,i) = \hat{\mathcal{U}}_\Phi(:,1:r_i)\hat{\mathcal{S}}_\Phi(1:r_i,1:r_i,i)\hat{\mathcal{V}}_\Phi^H(:,1:r_i,i).$$

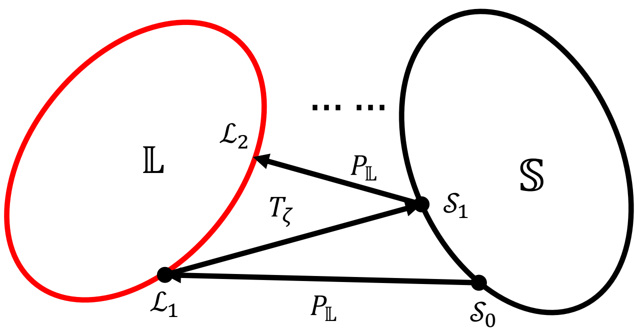
Then, $H_{\boldsymbol{r}}(\mathcal{X})$ is the best multi-rank $\boldsymbol{r}$ approximation to $\mathcal{X}$ in the Frobenius norm and

$$H_{\boldsymbol{r}} = \arg\min_{rank_m(\tilde{\mathcal{X}})\le \boldsymbol{r}} \|\mathcal{X}-\tilde{\mathcal{X}}\|_F$$

Sparse space projection

- hard thresholding

$$\mathcal{S}_{k+1} = T_{\zeta_{k+1}}(\mathcal{D}-\mathcal{L}_{k+1})$$



Efficient Alternating Projection for TRPCA

- less computational complexity on low-rank space projection than APT
- apply a tangent space projection before projecting the residue onto the low-rank space
- Low-rank space projection

$$\mathcal{L}_{k+1} = P_{\mathbb{L}} \circ P_{\mathbb{T}_k}(\mathcal{D} - \mathcal{S}_k)$$

  - truncated t-SVD on a smaller tensor of size $2r \times 2r \times q$ (original size $n \times n \times q$, r is tensor tubal rank)

Algorithm 4 EAPT: Efficient Alternating Projection Algorithm for Tensor RPCA

Run Algorithm 1 for initialization
for $k = 0$ to $T-1$ **do**
  $\tilde{\mathcal{L}}_k = \mathrm{Trim}(\mathcal{L}_k, \mu)$
  $\mathcal{L}_{k+1} = P_{\mathbb{L}} \circ P_{\mathbb{T}_k}(\mathcal{D} - \mathcal{S}_k)$
  $\mathcal{S}_{k+1} = T_{\zeta_{k+1}}(\mathcal{D} - \mathcal{L}_{k+1})$ where ζ_{k+1} is in (7)
end for
Return: $\mathcal{L}_T$ and $\mathcal{S}_T$

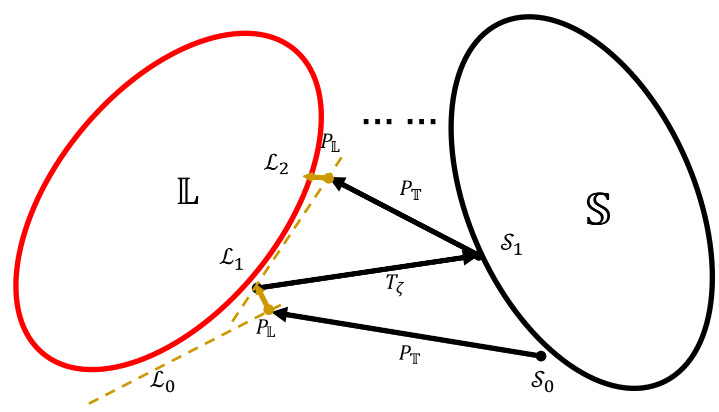
Proposition 4.2. *Projection of $P_{\mathbb{T}}(\mathcal{Z})$ onto low multi-rank space $\mathbb{L}$ can be executed by*

$$P_{\mathbb{L}} \circ P_{\mathbb{T}}(\mathcal{Z}) = \begin{bmatrix} \mathcal{U} & \mathcal{Q}_1 \end{bmatrix} \diamond_{\Phi} P_{\mathbb{L}}(\mathcal{M}) \diamond_{\Phi} \begin{bmatrix} \mathcal{V}^H \\ \mathcal{Q}_2^H \end{bmatrix}, \tag{6}$$

where

$$\mathcal{M} = \begin{bmatrix} \mathcal{U}^H \diamond_{\Phi} \mathcal{Z} \diamond_{\Phi} \mathcal{V} & \mathcal{R}_2^H \\ \mathcal{R}_1 & \mathbf{0} \end{bmatrix} \in \mathbb{R}^{2r \times 2r \times q},$$

$\mathcal{Q}_1 \diamond_{\Phi} \mathcal{R}_1 = (\mathcal{I}_{\Phi} - \mathcal{U} \diamond_{\Phi} \mathcal{U}^H) \diamond_{\Phi} \mathcal{Z} \diamond_{\Phi} \mathcal{V}$ *and* $\mathcal{Q}_2 \diamond_{\Phi} \mathcal{R}_2 = (\mathcal{I}_{\Phi} - \mathcal{V} \diamond_{\Phi} \mathcal{V}^H) \diamond_{\Phi} \mathcal{Z} \diamond_{\Phi} \mathcal{U}$ *are t-QR (Theorem B.2).*

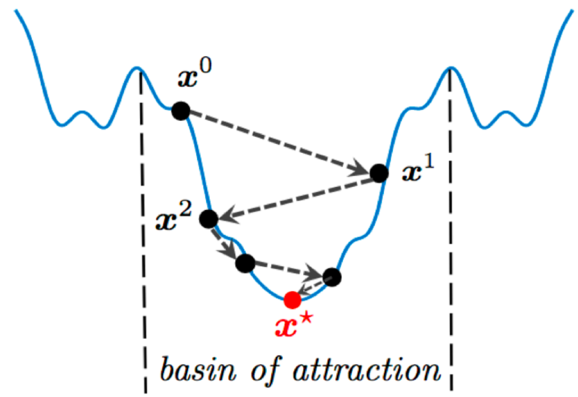


Initialization of APT and EAPT

- **Goal**
  - construct an initial guess that is sufficient close to the ground truth and is inside the "basin of attraction"
- **Initialization algorithm**

Algorithm 1 Initialization

$\mathcal{S}_{-1} = T_{\zeta_{-1}}(\mathcal{D})$ where $\zeta_{-1} = \beta_{\text{init}} \cdot \bar{\sigma}_1(\mathcal{D})$
$\mathcal{L}_0 = P_{\mathbb{L}}(\mathcal{D} - \mathcal{S}_{-1})$
$\mathcal{S}_0 = T_{\zeta_0}(\mathcal{D} - \mathcal{L}_0)$ where $\zeta_0 = \beta_0 \cdot \bar{\sigma}_1(\mathcal{D} - \mathcal{S}_{-1})$
Return: $\mathcal{L}_0$ and $\mathcal{S}_0$



  - $\mathcal{L}_0$ and $\mathcal{S}_0$ are the initialization values for both APT and EAPT

Theoretical Results

Some assumptions:

- assumption for low-rank tensor (incoherence)

Assumption 5.1. Given the transformed t-SVD of a tensor $\mathcal{L} = \mathcal{U} \diamond_\Phi \Sigma \diamond_\Phi \mathcal{V}^H \in \mathbb{R}^{n\times n\times q}$ with multi-rank $\boldsymbol{r}$, $\mathcal{L}$ is said to satisfy the tensor incoherent condition, if there exists $\mu > 0$ such that

$$\textit{Tensor-column:} \quad \frac{nq}{s_r} \max_{i\in[n]} \|\mathcal{U}^H \diamond_\Phi \dot{e}_i\|_F^2 \leq \mu;$$
$$\textit{Tensor-row:} \quad \frac{nq}{s_r} \max_{j\in[n]} \|\mathcal{V}^H \diamond_\Phi \dot{e}_j\|_F^2 \leq \mu.$$

- assumption for sparse tensor

Assumption 5.2. A sparse tensor $\mathcal{S}\in\mathbb{R}^{n\times n\times q}$ is α-sparse, i.e., $\|\mathcal{S}(:,i,:)\|_0 \leq \alpha nq$ and $\|\mathcal{S}(i,:,:)\|_0 \leq \alpha nq$ for $i\in[n]$.

Guarantee for the initialization close to ground truth

Proposition 5.3 (Algorithm 1 for initialization). *Assume that a low multi-rank $\boldsymbol{r}$ tensor $\mathcal{L}^*$ satisfies Assumption 5.1 and a sparse $\mathcal{S}^*$ satisfies Assumption 5.2 with $\alpha\mu \lesssim \frac{1}{s_r\kappa\sqrt{q}}$. For hyperparameters obeying $\frac{\mu s_r \bar{\sigma}_1(\mathcal{L}^*)}{nq\bar{\sigma}_1(\mathcal{D})} \leq \beta_{init} \leq \frac{3\mu s_r \bar{\sigma}_1(\mathcal{L}^*)}{nq\bar{\sigma}_1(\mathcal{D})}$ and $\beta_0 = \frac{\mu s_r}{2nq}$, the outputs of Algorithm 1 satisfy $\|\mathcal{L} - \mathcal{L}_0\| \leq 8\alpha\mu s_r \bar{\sigma}_1(\mathcal{L}^*)$ and $\|\mathcal{S} - \mathcal{S}_0\|_\infty \leq \frac{\mu s_r}{nq}\bar{\sigma}_1(\mathcal{L}^*)$.*

Recovery guarantee of APT

Theorem 5.4 (Exact recovery of Algorithm 2). *Under the assumption of Proposition 5.3, for any $\epsilon > 0$, we have $\|\mathcal{L}_T - \mathcal{L}^*\| \leq 8\alpha\epsilon$ and $\|\mathcal{S}_T - \mathcal{S}^*\|_\infty \leq {}^{4\epsilon}/_{nq}$ with $T = \mathcal{O}\left(\log\left({}^1/_\epsilon\right)\right), \beta = {}^{2\mu s_r}/_{nq}$.*

Recovery guarantee of EAPT

Theorem 5.5 (Exact recovery of Algorithm 4). *Under the assumption of Proposition 5.3 except that $\alpha \lesssim \min\{ {}^1/_{\mu s_r^2\kappa^3}, {}^{q^{0.5}}/_{\mu^{1.5}s_r^2\kappa^2}, {}^{q^{0.5}}/_{\mu^2 s_r^2\kappa}\}$[1], for any $\epsilon > 0$, we have $\|\mathcal{L}_T - \mathcal{L}^*\|_\infty \leq 8\alpha\epsilon$ and $\|\mathcal{S}_T - \mathcal{S}^*\|_\infty \leq {}^\epsilon/_{nq}$ with $T = \mathcal{O}\left(\log\left({}^1/_\epsilon\right)\right), \beta = {}^{\mu s_r}/_{2nq}$.*

Comparison with other methods

t-SVD methods	effectiveness (recovery performance)			efficiency (optimization time)		
	transformed	rank	recovery guarantee	convergence rate	iteration complexity	
					FFT	DCT
TRPCA (Lu et al., 2019)	✗	tubal	✓	$\mathcal{O}(1/\epsilon)$	$\mathcal{O}\left(n^2 q \log q + n^3 q\right)$	—
ETRPCA (Gao et al., 2020)	✗	tubal	✗	✗	$\mathcal{O}\left(n^2 q \log q + n^3 q\right)$	—
T-TRPCA (Lu, 2021)	✓	tubal	✓	$\mathcal{O}(1/\epsilon)$	—	$\mathcal{O}(n^2 q^2 + n^3 q)$
APT	✓	multi	✓	$\mathcal{O}\left(\log(1/\epsilon)\right)$	$\mathcal{O}(n^2 q \log q + n^3 q)$	$\mathcal{O}(n^2 q^2 + n^3 q)$
EAPT	✓	multi	✓	$\mathcal{O}\left(\log(1/\epsilon)\right)$	$\mathcal{O}(n^2 q \log q + n^2 s_r)$	$\mathcal{O}(n^2 q^2 + n^2 s_r)$

- Our methods have linear convergence rate and recovery guarantee
- Our methods can make use of information in the frequency domain
- EAPT has less iteration complexity

More comparison

	decomposition	Convergence	Iteration complexity	Algorithm	Recovery
TRPCA (Lu et al., 2019)	t-SVD	$\mathcal{O}(1/\epsilon)$	$\mathcal{O}\left(n^3 \log n + n^4\right)$	ADMM	✓
Transformed TRPCA (Lu, 2021)	T t-SVD	$\mathcal{O}(1/\epsilon)$	$\mathcal{O}\left(n^4\right)$	ADMM	✓
ETRPCA (Gao et al., 2020)	t-SVD	✓	$\mathcal{O}\left(n^3 \log n + n^4\right)$	ALM	✗
RTCUR (Cai et al., 2021)	Fiber CUR	✗	$\mathcal{O}\left(nr^2 \log^2(n) + r^4 \log^4(n)\right)$	AP	✗
TTNN (Yang et al., 2020)	TT	✓	$\mathcal{O}\left(n^4\right)$	ADMM	✗
SNN (Huang et al., 2015)	Tucker	$\mathcal{O}(1/\epsilon)$	$\mathcal{O}\left(n^4\right)$	ADMM	✓
Atomic Norm (Driggs et al., 2019)	CP	✗	$\mathcal{O}(n^4 r)$	LBFGS	✗
RTD (Anandkumar et al., 2016)	CP	$\mathcal{O}\left(\log(1/\epsilon)\right)$	$\mathcal{O}\left(n^{4+c} r^2\right)$	AP	✓
APT	T t-SVD	$\mathcal{O}\left(\log(1/\epsilon)\right)$	$\mathcal{O}(n^4)$	AP	✓
EAPT	T t-SVD	$\mathcal{O}\left(\log(1/\epsilon)\right)$	$\mathcal{O}(n^4 + n^2 s_r)$	AP	✓

The computational complexities are calculated on tensors in $\mathbb{R}^{n\times n\times n}$

Experiments: synthetic data

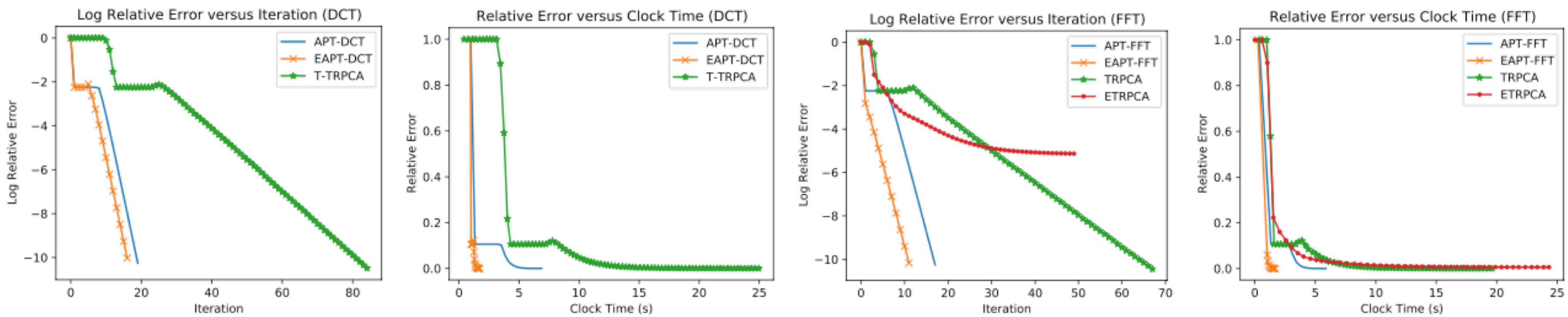


Figure 1. Comparison between different t-SVD based tensor RPCA methods.

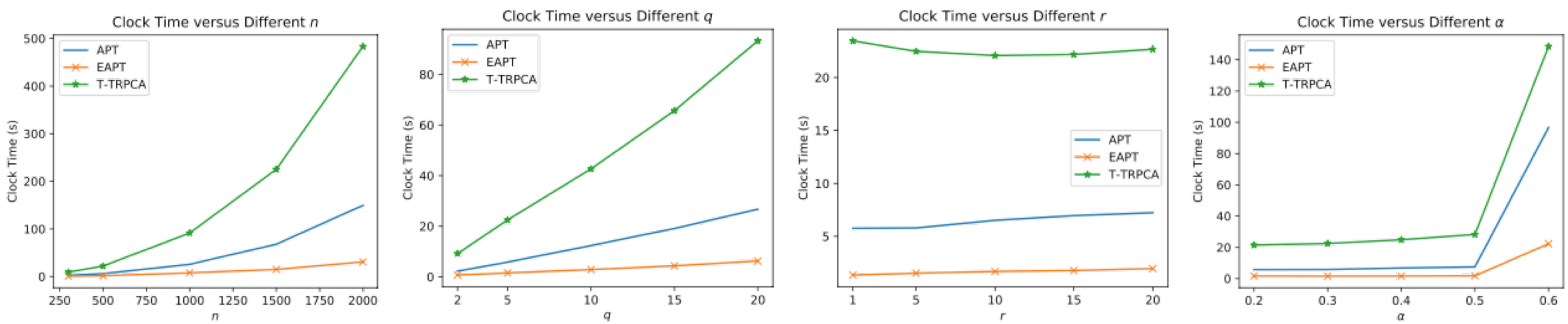


Figure 2. Consistent advantages over T-TRPCA with various parameters. $\alpha = 0.6$ fails to successfully recovery.

Experiments: real data

	TT		Tucker		CP		CP		CP		t-SVD									
	TTNN		SNN		Atomic Norm		RTD		KBR-TRPCA		TRPCA		T-TRPCA		ETRPCA		EAPT-FFT		EAPT-DCT	
	PSNR	Time	PSNR	Time	PSNR	Time	PSNR	Time	PSNR	Time	PSNR	Time	PSNR	Time	PSNR	Time	PSNR	Time	PSNR	Time
toys	29.39	85.4	28.47	297.3	16.96	223.1	24.18	368.6	36.11	305.8	34.04	63.6	34.09	75.5	38.47	121.7	39.95	36.5	**41.87**	**35.8**
feathers	29.00	84.5	28.00	297.8	17.89	331.4	24.42	486.8	36.23	189.9	31.62	59.9	31.36	70.7	36.72	120.8	38.01	33.4	**39.61**	**32.3**
sponges	36.90	83.3	37.38	305.1	19.48	337.4	28.24	249.8	**44.16**	227.8	31.52	59.9	30.28	70.8	34.81	122.2	38.88	33.6	40.05	**22.5**
watercolors	28.74	85.9	28.31	284.7	18.33	377.2	23.49	353.1	35.77	316.4	36.28	61.6	36.3	71.3	40.59	121.6	41.43	**34.2**	**41.66**	35.9
paints	30.35	83.1	30.33	291.5	18.98	336.0	25.16	457.5	32.20	248.6	33.83	61.1	33.72	71.5	38.15	123.7	39.45	**34.3**	**39.53**	35.6
sushi	31.60	84.0	31.57	312.2	17.42	320.9	29.96	492.3	36.03	187.8	33.40	62.5	33.50	72.5	35.67	120.3	36.03	36.3	**39.30**	**32.3**

Results about HSI denoising

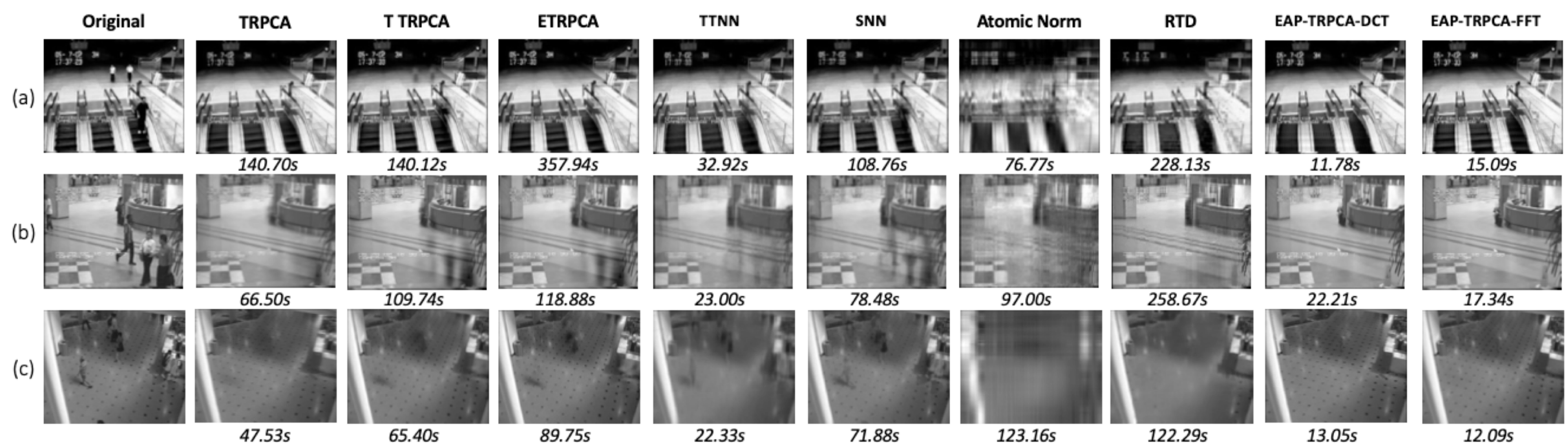


Figure 3. Video background subtraction results of different methods and their corresponding reconstruction clock time. (a) Escalator with 200 frames; (b) Hall with 100 frames; (c) ShoppingMall with 50 frames.

apply smaller rank to high frequency term

Summary

- We propose two alternating projection algorithms for tensor RPCA. Specifically, EAPT is more efficient since it uses the tangent space of low-rank tensor to reduce iteration complexity.
- Linear convergence to the ground-truth can be guaranteed under suitable tensor incoherence conditions.
- Experiments on synthetic data and real data demonstrate both efficiency and effectiveness of our methods.
- Several interesting problems need to be further investigated: higher order case, initialization free procedure, …

Thank you!