A Prarallel Linearized ADMM with Application to Multichannel TGV-Based Image Restoration

Chuan He, Changhua Hu

the High-tech Institute of Xi'an

Xuelong Li

Xi'an Institute of Optics and Precision Mechanics

Outline

- Introduction
- Proposed Method
- Multichannel TGV-Based Image Restoration
- Concluding Remarks

Introduction

f :Observation *K*: Blur Matirx *u*: Orignial Scene? *n*: Noise

$$f = Ku + n$$

Matrix -Vector

The key to the success of image restoration:

- **Regularization model incorporating image prior knowledge**
- **Automatic, accurate, concise, and fast solution algorithm**

Introduction

Convex Objective Function for Imaging Inverse Problems:

$$\min_{\boldsymbol{x}\in X} g(\boldsymbol{x}) + \sum_{h=1}^{H} f_h(\boldsymbol{L}_h \boldsymbol{x}) \qquad (1)$$

- $\boldsymbol{g}$ and $\boldsymbol{f_h}$ are convex functions whose proximity operators possess closed-forms or at least can be solved efficiently by existent methods;
- L_h is a bounded linear operator with adjoint $\boldsymbol{L}_h^*$.

The solution of (1) usually suffers from two aspects:

- Data space X in a practical application is typically of high dimension;
- Function $\boldsymbol{g}$ and the linear-operator-coupled $\boldsymbol{f_h}$ may be nondifferentiable.

Introduction

Our Strategy-Parallel LADMM with “full splitting”:

$$\min_{\boldsymbol{x}\in X} g(\boldsymbol{x}) + \sum_{h=1}^{H} f_h(\boldsymbol{L}_h \boldsymbol{x}) \qquad (1)$$

- At each iteration, only the proximity operators of the convex functions and the linear operators are involved. It possesses a highly parallel structure and can be accelerated by parallel calculation techniques.
- The linear inverse operator, which usually exists in methods dealing with inverse problems. is excluded. It is not partial to a particular data boundary condition.
- It achieves a worst-case O (1/k) convergence rate by exploiting only the first-order information of the functions.

Proposed Method

Similar to ADMM, the AL functional of (1) is as follows

$$\mathcal{L}_{\mathcal{A}}\left(\boldsymbol{x},\boldsymbol{a}_1,\ldots,\boldsymbol{a}_H;\boldsymbol{v}_1,\ldots,\boldsymbol{v}_H\right)=g\left(\boldsymbol{x}\right)+\sum_{h=1}^{H}\left(f_h\left(\boldsymbol{a}_h\right)+\left\langle \boldsymbol{v}_h,\boldsymbol{L}_h\boldsymbol{x}-\boldsymbol{a}_h\right\rangle+\frac{\beta_h}{2}\left\|\boldsymbol{L}_h\boldsymbol{x}-\boldsymbol{a}_h\right\|_2^2\right). \quad (2)$$

- v_h is the Lagrange multiplier and $\beta_h > 0$ is the penalty parameter.

The proposed PLADMM finding the saddle point of (2):

$$\begin{cases}\boldsymbol{a}_h^{k+1}=\operatorname{prox}_{f_h/\beta_h}\left(\boldsymbol{L}_h\boldsymbol{x}^{k+1}+\dfrac{\boldsymbol{v}_h^k}{\beta_h}\right) & h=1,\ldots,H;\\ \boldsymbol{v}_h^{k+1}=\boldsymbol{v}_h^k+\beta_h\left(\boldsymbol{L}_h\boldsymbol{x}^{k+1}-\boldsymbol{a}_h^{k+1}\right) & h=1,\ldots,H;\\ \boldsymbol{x}^{k+1}=\operatorname{prox}_{tg}\left(\boldsymbol{x}^k-t\sum_{h=1}^{H}\beta_h\boldsymbol{L}_h^*\left(\boldsymbol{L}_h\boldsymbol{x}^k-\boldsymbol{a}_h^{k+1}+\dfrac{\boldsymbol{v}_h^{k+1}}{\beta_h}\right)\right), & 0<t\le\left(1\Big/\sum_{h=1}^{H}\beta_h\left\|\boldsymbol{L}_h^*\boldsymbol{L}_h\right\|\right).\end{cases}$$

Proposed Method

With the Moreau decomposition in convex analysis:

$$\operatorname{prox}_{\beta f^*}\boldsymbol{v} = \boldsymbol{v} - \beta\operatorname{prox}_{f/\beta}\left(\boldsymbol{v}/\beta\right)$$

The iterative scheme of PLADMM is transformed into:

$$\begin{cases} \boldsymbol{v}_h^{k+1} = \operatorname{prox}_{\beta_h f_h^*}\left(\beta_h \boldsymbol{L}_h \boldsymbol{x}^{k+1} + \boldsymbol{v}_h^k\right) \quad h = 1,\ldots,H; \\ \boldsymbol{x}^{k+1} = \operatorname{prox}_{tg}\left(\boldsymbol{x}^k - t\sum_{h=1}^{H}\boldsymbol{L}_h^*\left(2\boldsymbol{v}_h^{k+1} - \boldsymbol{v}_h^k\right)\right),\ 0 < t \le \left(1\Big/\sum_{h=1}^{H}\beta_h\left\|\boldsymbol{L}_h^*\boldsymbol{L}_h\right\|\right). \end{cases}$$

Proposed Method

According to the convergence analysis of LADMM (Theorem1 in the paper)

$\left\{ \boldsymbol{x}^k, \boldsymbol{a}_1^k, \ldots, \boldsymbol{a}_H^k; \boldsymbol{v}_1^k, \ldots, \boldsymbol{v}_H^k \right\}$ converges to a saddle point of

$\mathcal{L}_{\mathcal{A}}\left(\boldsymbol{x}, \boldsymbol{a}_1, \ldots, \boldsymbol{a}_H; \boldsymbol{v}_1, \ldots, \boldsymbol{v}_H \right)$;

$\left\{ \boldsymbol{x}^k \right\}$ converges to a solution of problem (1);

PLADMM possesses a worst-case $O(1/k)$ convergence rate.

Multichannel TGV-Based Image Restoration

Objective Function: $$\left(\boldsymbol{u}^*, \boldsymbol{p}^*\right) = \arg\min_{\boldsymbol{u},\boldsymbol{p}} \ \alpha_1 \left\|\nabla \boldsymbol{u} - \boldsymbol{p}\right\|_1 + \alpha_2 \left\|\mathcal{E}\boldsymbol{p}\right\|_1$$

$$\text{s.t.} \quad \left\{\boldsymbol{u} \in \Omega \triangleq \{\boldsymbol{u} : \boldsymbol{0} \le \boldsymbol{u} \le \boldsymbol{255}\} \cap \Psi \triangleq \left\{\boldsymbol{u} : \left\|\boldsymbol{K}\boldsymbol{u} - \boldsymbol{f}\right\|_2^2 \le c\right\}\right\}.$$

PLADMM Scheme:

$$\begin{cases} \boldsymbol{v}_{1,i,j,l}^{k+1} = P_{B_{\alpha_1}}\left(\beta_1\left(\left(\nabla \boldsymbol{u}^k\right)_{i,j,l} - \boldsymbol{p}_{i,j,l}^k\right) + \boldsymbol{v}_{1,i,j,l}^k\right) \\ \boldsymbol{v}_{2,i,j,l}^{k+1} = P_{B_{\alpha_2}}\left(\beta_2\left(\mathcal{E}\boldsymbol{p}^k\right)_{i,j,l} + \boldsymbol{v}_{2,i,j,l}^k\right) \\ \boldsymbol{v}_3^{k+1} = \beta_3 S_{\sqrt{c}}\left(\dfrac{\boldsymbol{v}_3^k}{\beta_3} + \boldsymbol{K}\boldsymbol{u}^k - \boldsymbol{f}\right) \\ \boldsymbol{u}^{k+1} = P_{\Omega}\left(\boldsymbol{u}^k - t\left(\nabla^T \tilde{\boldsymbol{v}}_1^{k+1} + + \boldsymbol{K}^T \tilde{\boldsymbol{v}}_3^{k+1}\right)\right) \\ \boldsymbol{p}_1^{k+1} = \boldsymbol{p}_1^k - t\left(\nabla_1^T \tilde{\boldsymbol{v}}_{2,1}^{k+1} + \nabla_2^T \tilde{\boldsymbol{v}}_{2,3}^{k+1} - \tilde{\boldsymbol{v}}_{1,1}^{k+1}\right) \\ \boldsymbol{p}_2^{k+1} = \boldsymbol{p}_2^k - t\left(\nabla_1^T \tilde{\boldsymbol{v}}_{2,3}^{k+1} + \nabla_2^T \tilde{\boldsymbol{v}}_{2,2}^{k+1} - \tilde{\boldsymbol{v}}_{1,2}^{k+1}\right) \end{cases}$$

$$\tilde{\boldsymbol{v}}_h^{k+1} = 2\boldsymbol{v}_h^{k+1} - \boldsymbol{v}_h^k$$

Multichannel TGV-Based Image Restoration

The experiment was performed in MATLAB on a PC with an Intel Core i5 CPU (3.20GHz) and 8GB of RAM.



Images: **Lena (256×256), Peppers (512×512), and Monarch (768 × 512)**

Multichannel TGV-Based Image Restoration

Problem	Image	Blur Kernels	σ	PSNR	SSIM
1	Lena	Set 1	3	20.05	0.5239
2	Peppers	Set 2	8	17.55	0.5140
3	Monarch	Set 3	10	17.95	0.4608

The three blurs are generated: (1). Generate 9 kernels: {A(13), A(15), A(17), G(11, 9), G(21, 11), G(31, 13), M(21, 45), M(41, 90), M(61, 135)}; (2). Assign the above 9 kernels to {$K11$, $K12$, $K13$; $K21$, $K22$, $K23$; $K31$, $K32$, $K33$}; (3). then with the above kernels, we generate the final three sets of blurs for comparison by multiplying relative weights {1, 0, 0; 0, 1,0; 0, 0, 1} (Set 1), {0.6, 0.2, 0.2; 0.15, 0.7, 0.15; 0.1, 0.1, 0.8} (Set 2), and {0.7, 0.15, 0.15; 0.1, 0.8, 0.1; 0.2, 0.2, 0.6} (Set 3) to the corresponding blur kernels.

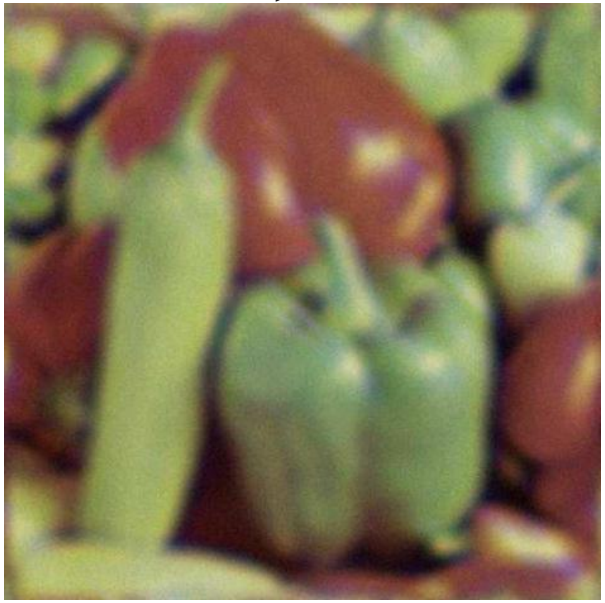
Multichannel TGV-Based Image Restoration

Comparison in PSNR, SSIM, and CPU time

Problem	Method	PSNR	SSIM	CPU
1	PLADMM-TGV	**26.21**	**0.7680**	32.24
	APEADMM-TGV	**26.21**	0.7649	35.62
	FTVD-v4	26.04	0.7583	10.37
2	PLADMM-TGV	**25.57**	**0.7632**	155.46
	APEADMM-TGV	25.56	0.7623	199.93
	FTVD-v4	25.25	0.7507	54.69
3	PLADMM-TGV	**23.85**	**0.8083**	234.45
	APEADMM-TGV	23.83	0.8063	302.68
	FTVD-v4	23.61	0.7965	84.43

APEADMM: He2014(IEEE-TIP)

PSNR=17.55, SSIM=0.5140

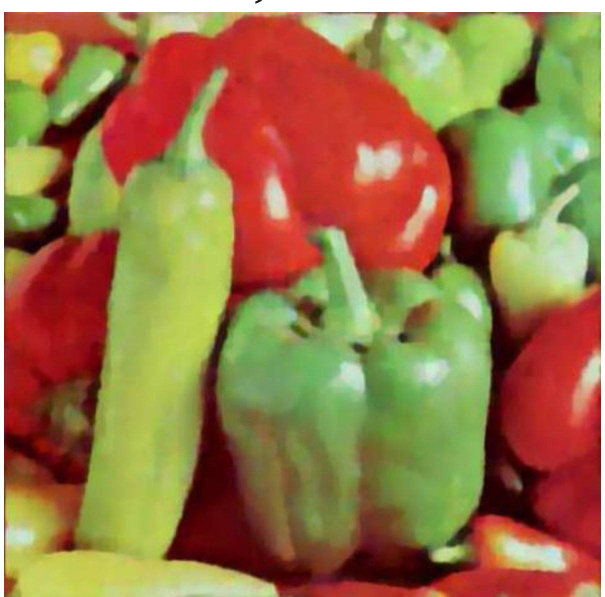


Degraded

PSNR=25.57, SSIM=0.7632

PLADMM-TGV

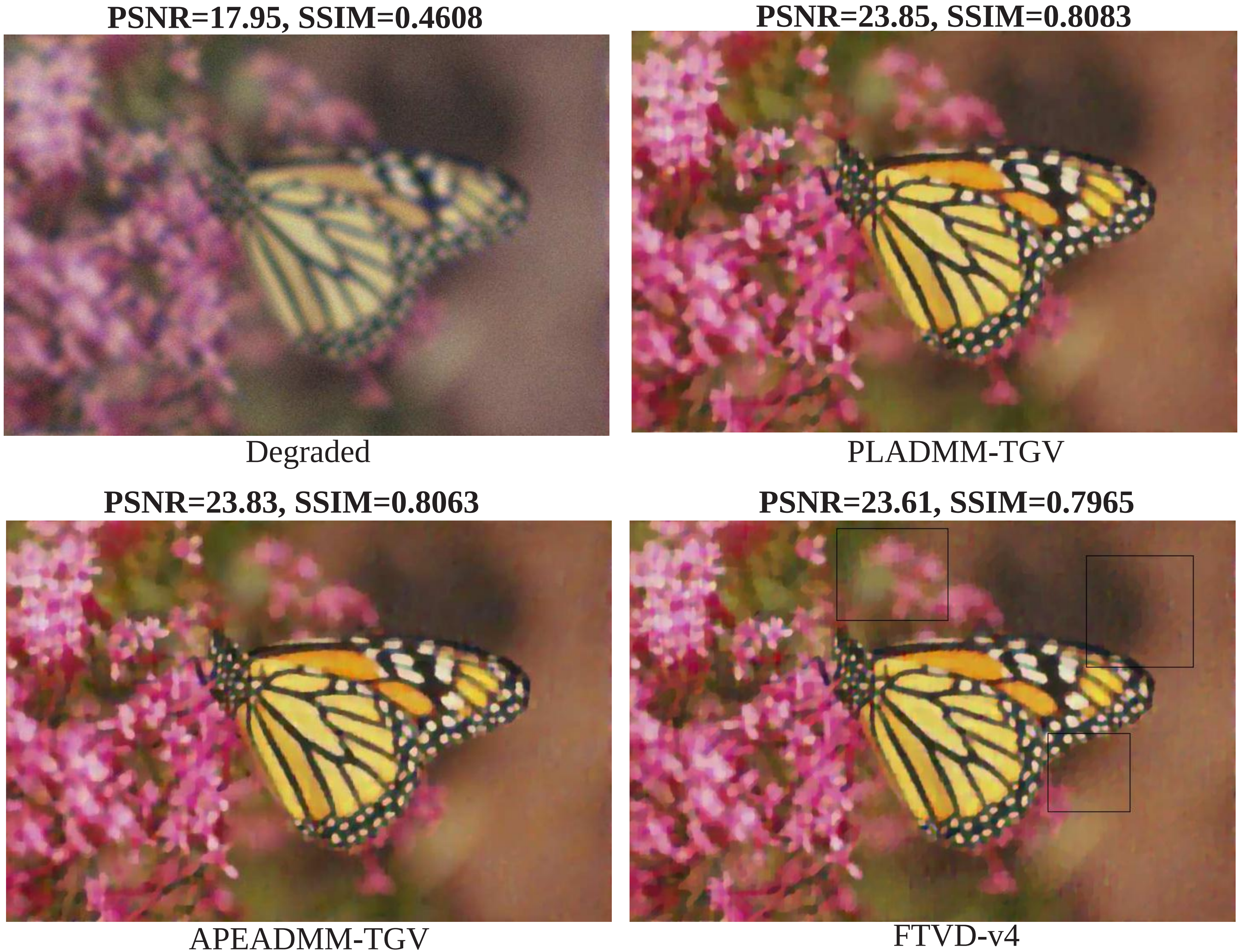
PSNR=25.56, SSIM=0.7623



APEADMM-TGV

PSNR=25.25, SSIM=0.7507

FTVD-v4


PSNR=17.95, SSIM=0.4608
Degraded
PSNR=23.85, SSIM=0.8083
PLADMM-TGV
PSNR=23.83, SSIM=0.8063
APEADMM-TGV
PSNR=23.61, SSIM=0.7965
FTVD-v4

Some Related Works

1. **He Chuan**, Hu Changhua, Zhang Wei, et al. A fast adaptive parameter estimation for total variation image restoration [J]. *IEEE Transactions on Image Processing*, 2014, 23(12): 4954–4967. (SCI, IF: 4.828, EI)
2. **He Chuan**, Hu Changhua, Li Xuelong, et al. A parallel alternating direction method with application to compound l_1-regularized imaging inverse problems. *Information Sciences*, 2016, 348: 179-197. (SCI, IF: 4.832, EI)
3. **He Chuan**, Hu Changhua, Li Xuelong, et al. A parallel primal-dual splitting method for image restoration [J]. *Information Sciences*, 2016, 358-359: 73-91. (SCI, IF: 4.832, EI)
4. **He Chuan**, Hu Changhua, Li Xuelong. A parallel linearized ADMM with application to multichannel TGV-based image restoration [C]. *IEEE International Conference on Image Processing*, Beijing, China, September, 2017.(EI)
5. **He Chuan**, Hu Changhua, Zhang Wei. Adaptive shearlet-regularized image deblurring via alternating direction method [C]. *IEEE International Conference on Multimedia & Expo* (IEEE ICME 2014), Chengdu, China, July, 2014. (EI)

The related MATLAB codes can be found on my Researchgate.

Conclusion

- Full Splitting
  - More Parallel
- Excluding Matrix Inverse Operation
  - More concise and more fast
- Extension
  - Other regularizations and other image inverse problems

THANK YOU!

Dr. Chuan He

hechuan8512@163.com