LEARNING COMPACT PARTIAL DIFFERENTIAL EQUATIONS FOR COLOR IMAGES WITH EFFICIENCY

*Zhenyu Zhao, Chenping Hou, and Bo Lin**

College of Liberal Arts and Sciences
National University of Defense Technology
Changsha, P. R. China

Cong Fang†

Key Lab. of Machine Perception (MOE)
School of EECS, Peking University
Beijing, P. R. China

ABSTRACT

Learning Partial Differential Equations (LPDEs) from training data for particular tasks has been successfully applied to many image processing problems. In this paper, we aim to learn compact Partial Differential Equations (LCPDEs) for color image tasks by proposing a more effective algorithm. The LCPDEs system is formulated as a linear combination of fundamental differential invariants and simplified by omitting the PDE which works as an indicate function. We replace L2-norm with L1-norm to regularize the coefficients with respect to the invariants. As the objective function is non-smooth, we resort to proximal algorithm to optimize it, which ensures convergence in an at least sub-linear rate. Experiments demonstrate the advantages of the proposed method over other PDE-based methods in terms of both quality and efficiency.

Index Terms— Partial differential equations, L1-norm regularization, Proximal algorithm, Color images.

1. INTRODUCTION

Partial Differential Equations (PDEs) have shown their superiority in computer vision and image processing [1, 2], e.g., denoising [3], enhancement [4], segmentation [5], stereo and optical flow computation [6]. According to [7], there are usually two kinds of methods used to design PDEs. For one kind, PDEs are written down directly, based on some mathematical understandings on the properties of the PDEs (e.g., anisotropic diffusion [3] and shock filter [4]). Another kind of methods basically define an energy functional first, which collects the wish list of the desired properties of the output image, and then deducts the evolution equations by computing the Euler-Lagrange variation of the energy functional (e.g. CTV [8], GTV [9] and PL [10]). However, both methods for designing PDEs require high mathematical skills and good insight into the problems.

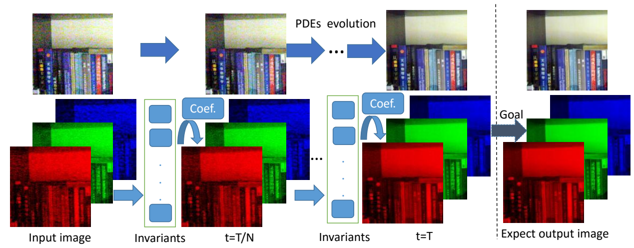

Fig. 1. Illustration of the proposed approach. The PDEs (evolution equations) are formulated as a linear combination of fundamental translational and rotational invariants. We learn the coefficients with respect to the invariants (Coef. is short for coefficients).

Recently, Liu et al. [7] proposed a method that learned partial differential equations (LPDEs) from training image pairs (i.e., the input image and the expect output image). In [11, 7], they successfully apply LPDEs to image restoration, debluring and denoising tasks. They also use LPDEs to solve some mid-and-high-level tasks that the traditional PDE-based methods cannot. In [7], they apply LPDEs to object detection, color2gray, and demosaicking. For saliency detection, Liu et al. [12, 13] propose to learn the boundary of a PDEs system. Zhao et al. [14] extend the LPDEs method to text detection tasks and Fang et al. [15] propose a learning PDEs model to feature learning. Although they have made great progress, they mainly focus on gray images or particular tasks.

In this paper, we focus on color images and propose a Learning Compact PDEs (LCPDEs) model with an effective solving algorithm. An illustration of our approach is shown in Figure 1. We assume that all the PDEs in our system are evolution equations [16] and the evolutionary rates of the PDEs are formulated as linear combination of fundamental translational and rotational invariants. When given training image pairs, we learn the coefficients with respect to the fundamental invariants to obtain data-driven PDEs. Yet, our model has the following distinctions from others. First, we design our PDEs system only has three coupled PDEs (for R, G, B three

*E-mail: zhaozhenyu01@nudt.edu.cn, hcpnudt@hotmail.com, linbo@nudt.edu.cn.

This work was supported by the National Natural Science Foundation of China (11701566, 61473302, 61503396, 61601479).

†E-mail: tjufangcong19911231@gmail.com.

Table 1. Fundamental differential invariants.

j	$\mathbf{inv}_j(\mathbf{u})$ $\left(\mathbf{u}=\{u^1,u^2,u^3\}\right)$
0-3	$1, u^i; i\in\{1,2,3\}$.
4-9	$\left(\nabla u^i\right)^T\nabla u^j = u^i_x u^j_x + u^i_y u^j_y$; $i,j\in\{1,2,3\}$ and $j\geq i$.
10-12	$tr\left(\mathbf{H}_{u^i}\right) = u^i_{xx}+u^i_{yy}; i=1,2,3.$
13-30	$\left(\nabla u^i\right)^T\mathbf{H}_{v^k}\nabla u^j = u^i_x u^j_x u^k_{xx} + \left(u^i_x u^j_y + u^i_y u^j_x\right) u^k_{xy} + u^i_y u^j_y u^k_{yy}; i,j,k\in\{1,2,3\}$ and $j\geq i$.
31-36	$tr\left(\mathbf{H}_{u^i}\mathbf{H}_{u^j}\right) = u^i_{xx}u^j_{xx}+2u^i_{xy}u^j_{xy}+u^i_{yy}u^j_{yy}$; $i,j\in\{1,2,3\}$ and $j\geq i$.

channels, respectively, see in Figure 1), omitting the PDE with respect to indication function which was introduced for collecting global information in Liu et al. [7]. As it is redundant for color image and omitting it can reduce the optimization difficulties. Second, we use L1-norm (also called sparsity regularization), instead of L2-norm, to regularize the coefficients with respect to the fundamental invariants, since we find that for color images these coefficients are usually in a sparse representation. Third, we resort to the proximal algorithm [17, 18] to optimize the objective function as the optimal control approach proposed by [7] cannot optimize non-smooth function. Note that LCPDEs can handle many color image tasks as one just need to prepare the training image pairs for each task.

In summary, the advantages of our common model are as follows:

1. A learning compact PDEs (LCPDEs) model is proposed by omitting the PDE which works as an indication function and replacing L_2-norm regularization to L_1-norm regularization. It is much suitable for color image tasks.
2. We resort to the effective proximal algorithm for learning the compact PDEs. The algorithm can guarantee the convergence in an at least sub-linear rate.

2. LEARNING COMPACT PDES

In this section, we introduce the LCPDEs model and present the whole problem formulation.

2.1. PDEs formulation

We assume that the color image process can be described as three evolution equations. Considering the translational and rotational invariance of computer vision and image processing problems, it is proved [7] that the evolutionary rates are functions of fundamental differential invariants, which form "bases" of all differential invariants that are invariant with respect to translation and rotation. As shown in Table 1, there are 37 fundamental differential invariants $\{\text{inv}_i(\mathbf{u}), i=0,\cdots,36\}$ up to the second order.

Considering the simplest case, we choose the evolutionary rates as a linear combination coefficients of fundamental differential invariants. Then the coefficients respect to fundamental invariants are provably functions of time t only and independent of spatial variables [7]. When training, we minimize the difference between the output of PDEs and the ground truth (the last two columns in Figure 1), which is a PDEs constrained optimal control problem, formulated as:

$$\min_{\mathbf{a}} E\left(\mathbf{a}(t)\right) = \frac{1}{2}\sum_{m=1}^{M}\sum_{c=1}^{3}\int_\Omega \left(O^c_m - u^c_m(x,y,T)\right)^2 \,\mathrm{d}\Omega$$
$$s.t. \begin{cases} \frac{\partial u^c_m}{\partial t} = \sum\limits_{i=0}^{36} a^c_i(t)\text{inv}_i(\mathbf{u}_m(t)), & (x,y,t)\in Q,\\ u^c_m(x,y,t)=0, & (x,y,t)\in\Gamma,\\ u^c_m|_{t=0}(x,y,t) = I^c_m, & (x,y)\in\Omega, \end{cases} \tag{1}$$

where $\{(\mathbf{I}_m,\mathbf{O}_m) = (I^c_m, O^c_m)|c=1,2,3, m=1,\cdots,M\}$ denote the M input/output training image pairs, $u_m(x,y,t)$ is the evolution image at time t with respect to the input image I_m, $\Omega\subset\mathbb{R}^2$ is the (rectangular) region occupied by the image[1], T is the temporal span of evolution which can be normalized as 1, $Q=\Omega\times[0,T]$, $\Gamma=\partial\Omega\times[0,T]$, and $\partial\Omega$ denotes the boundary of Ω.

Note that we omit the PDE with respect to indicate function which was introduced for collecting the global information [7] as it can reduce the optimization difficulties and the PDE is also redundant for color image tasks. Experiments show that our model is more suitable than LPDEs [7] for color images.

2.2. Problem formulation with L1-norm regularization

For most color image problems, we observe that a few fundamental invariants should be enough to represent the governing functions. That means the coefficient functions $\{\mathbf{a}^c_i(t)|i=0,1,\cdots,36, c=1,2,3\}$ should be sparse. For example, the mostly used diffusion equations $F=\text{div}(cu)$ just has one term $tr(\mathbf{H}_u)$ if c is a constant. For anisotropic diffusion, we have that $\text{div}(c\nabla u) = tr(c\mathbf{H}_u) + (\nabla u)^T\nabla c$, which can be considered as a combination of $tr(\mathbf{H}u)$ and ∇u with the pre-defined function c. Another reason we choose sparsity regularization is that when there are more terms in the governing functions, the numerical stability of PDEs is extremely harder to preserve.

With replacing L2-norm [7] to L1-norm regularization, the whole LCPDEs formulation can be stated as follows:

$$\begin{aligned} E(\mathbf{a}(t)) =& \frac{1}{2}\sum_{m=1}^{M}\sum_{c=1}^{3}\int_\Omega \left(O^c_m - u^c_m(x,y,T)\right)^2 \,\mathrm{d}\Omega \\ &+\lambda\sum_{i=0}^{36}\sum_{c=1}^{3}\int_0^T |a^c_i(t)|\,\mathrm{d}t, \end{aligned} \tag{2}$$

[1] The images are padded with zeros of several pixels width around them.

where $u_m^c(x,y,T)$ is the evolution image by (1) at time T with respect to the input image I_m.

3. PROXIMAL ALGORITHM

In this section, we introduce the proximal algorithm to learn the compact PDEs. It is motivated by the structure of the objective function (2), which ensures a closed-form solution on each iteration and guarantees convergence in an at least sub-linear rate when applying the proximal algorithm.

3.1. G$\hat{a}$teaux derivatives and discretization

3.1.1. Gâteaux derivatives

This subsection is dedicated to the gâteaux derivatives [19] of the functional E which is the key for derivation of a suitable minimization scheme. Here we only compute the gâteaux derivatives of the first term of (2) as the second term is nonsmooth.

With the help of the adjoint equations, the derivatives of $E(\mathbf{a})$ in (2) with respect to $\mathbf{a}^c$ are as follows:

$$\frac{DE}{Da_i^c} = -\sum_{m=1}^{M}\int_\Omega \varphi_m^c \mathrm{inv}_i(\mathbf{u}_m)\,\mathrm{d}\Omega, i=0,\cdots,36, c=1,2,3. \tag{3}$$

First, we need to deduce the adjoint equations for computing the gâteaux derivatives, which is widely used in optimal control methods [20]. The adjoint equations with respect to $\phi=[\varphi^1,\varphi^2,\varphi^3]$ for $\mathbf{u}=[u^1,u^2,u^3]$ can be deduced as

$$\begin{cases} \frac{\partial\varphi^c}{\partial t} - E^c(\mathbf{u},\phi)=0, & (x,y,t)\in Q,\\ \varphi^c(x,y,t)=0, & (x,y,t)\in\Gamma,\\ \varphi^c(x,y,0)=O^c-u^c(1), & (x,y)\in\Omega, \end{cases} \tag{4}$$

where

$$E^c(\mathbf{u},\phi)=\sum_{(p,q)\in\mathcal{P}}(-1)^{p+q}\frac{\left(\partial^{p+q}\sum_{i=1}^{3}\sigma_{pq}^c(u^i)\varphi^i\right)}{\partial x^p\partial y^q},$$

$$\sigma_{pq}^c(u^i)=\frac{\partial\left(\sum_{j=0}^{36}a_j^i\mathrm{inv}^T(\mathbf{u})\right)}{\partial u_{pq}^c}, u_{pq}^c=\frac{\partial^{p+q}u^c}{\partial x^p\partial y^q}.$$

Then the derivatives of $E(\mathbf{a})$ in (2) with respect to $\mathbf{a}^c$ are as follows:

$$\frac{DE}{Da_i^c} = -\sum_{m=1}^{M}\int_\Omega \varphi_m^c \mathrm{inv}_i(\mathbf{u}_m)\,\mathrm{d}\Omega, i=0,\cdots,36, c=1,2,3. \tag{5}$$

We omit the deduction, since it is similar to the existing theory of optimal control governed by PDEs [7, 20].

3.2. Learning compact PDEs by proximal algorim

We use a finite difference scheme to make the discretization for the PDEs [7]. Then we can solve the LCPDEs model by the proximal algorithm. Proximal algorithm was first introduced as an approximation regularization method in convex optimization and then it showed the superiority in nonconvex minimization, such as in decision process of economics and decision sciences (procedural rationality) [17]. It is especially propitious to the case where the objective function E can be split as

$$E=f+g, \tag{6}$$

in which $f:\mathcal{R}^n\to\mathcal{R}$ is a C^1 function whose gradient ∇f is Lipschitz continuous, and $g:\mathcal{R}^n\to\mathcal{R}$ is non-smooth but has a closed-form minimizer when it sums with an independent quadratic auxiliary function. In this case, the algorithm is also known as iterative shrinkage-thresholding algorithm (ISTA) [21].

For any $L>0$, the proximal algorithm considers the following quadratic approximation of $E(\mathbf{a})$ at a given point $\mathbf{y}$:

$$Q_L(\mathbf{a},\mathbf{y}) := f(\mathbf{y})+\langle\mathbf{a}-\mathbf{y},\nabla f(\mathbf{y})\rangle+\frac{L}{2}\|\mathbf{a}-\mathbf{y}\|^2+g(\mathbf{a}). \tag{7}$$

When gotten the gradient of f, the algorithm goes to directly minimize $Q_L(\mathbf{a},\mathbf{a}_k)$ at each iteration. Especially, when $g(\mathbf{a})$ is the L1-norm regularization, minimizing $Q_L(\mathbf{a},\mathbf{a}_k)$ can be given a closed-form solution, stated as

$$\mathbf{a}_k=p_L(\mathbf{a}_{k-1})=\mathcal{T}_{\frac{1}{L}}(\mathbf{a}_{k-1}-\frac{1}{L}\nabla f(a_{k-1})), \tag{8}$$

where $\mathcal{T}_\epsilon(\omega)=\max(|\omega|-\epsilon)\mathrm{sgn}(\omega)$ is the shrinkage operator.

We denote $f(\mathbf{a})$ as the first term of (2) (the loss of the control problem) and $g(\mathbf{a})$ as the second term (the L_1-norm regularization). The iteration steps of optimizing objective function (2) meets the case of (8). We compute the gradient of f by (5). Usually, L is taken as the Lipschitz constant. But in our case, the Lipschitz constant of the smooth part (f) is not easily computable, so we use backtracking to search L. We use the same initialization method as [7] did and summarize the optimization process for LCPDEs in Algorithm 1.

Algorithm 1 LCPDEs by proximal algorithm.

Input Training image pairs $\{(\mathbf{I}_m,\mathbf{O}_m)\}_{m=1}^M$.
Step 0 Take $L_0>0$, some $\eta>1$ and $\mathbf{a}_0$.
Step k ($k\geq 1$) Find the smallest nongetive integers i_k such that with $\overline{L}=\eta^{i_k}L_{k-1}$.

$$F(p_{\overline{L}}(\mathbf{a}_{k-1}))\leq Q_{\overline{L}}(p_{\overline{L}}(\mathbf{a}_{k-1}),\mathbf{a}_{k-1}). \tag{9}$$

Set $L_k=\eta^{i_k}L_{k-1}$ and compute

$$\mathbf{a}_k=p_{L_k}(\mathbf{a}_{k-1}). \tag{10}$$

3.3. Convergence Analysis

We present convergence analysis of Algorithm 1 in this subsection. Main results are shown in Theorem 1. For space limitations, the proofs are omitted.

Theorem 1. *The sequence* $\{\mathbf{a}_k\}$ *generated by Algorithm 1 satisfies the following properties:*

1. *The sequence* $\{\mathbf{a}_k\}$ *is a Cauchy sequence and converges to a critical point of our optimization problem* (2).
2. *The sequence* $\{\mathbf{a}_k\}$ *converges to the critical point* $\{\mathbf{a}_k^*\}$ *of our optimization problem* (2) *in an at least sub-linear convergence rate, i.e., we meet the worse case when* $\theta \in (\frac{1}{2}, 1)$*, and there exists* $\omega > 0$*, such that*

$$\|\mathbf{a}_k - \mathbf{a}_k^*\| \leq \omega k^{-\frac{1-\theta}{2\theta-1}}. \tag{11}$$

4. EXPERIMENTS

As denoising is one of the most fundamental vision problems, we test on this task to demonstrate the superiority of our framework from two aspects: the performance of the model and the efficiency of the algorithm.

4.1. Natural color image denoising

For this task, experiments are verified on images with unknown natural noise. As LPDEs [7] method is the most related work to ours, we use the same dataset as [7] did. We also compare our method with existing PDE-based methods: color total variation (CTV [8]) and Parallel Level Sets (PL [10]). For LPDEs [7] and LCPDEs, we randomly choose 6 objects and for each object 10 noisy images are taken as samples. These noisy images and their ground truth images are used for training. Then we compare all the methods on images of the remaining objects. All the parameters among the methods are tuned to the best.

The experiment results are shown in Figure 2. Overall, we can see that color total variation (CTV) [8] performs much worse than LCPDEs, LPDEs [7] and PL [10]. PL can remove noise effectively, but it gives more blur results than LCPDEs and LPDEs. Although LPDEs gives sharper result than PL, it fails to remove some noise. The proposed LCPDEs has done the best balance at the two aspects. Moreover, we present the average PSNRs and SSIMs [22] of the remaining images in Table 2. Both of them are higher than all of other methods.

Table 2. The average PSNRs and SSIMs of the remaining images.

Method	CTV [8]	PL [10]	LPDEs [7]	LCPDEs
PSNR	23.61dB	23.99dB	23.97dB	**24.89**dB
SSIM	78.93%	77.89%	78.99%	**79.04**%

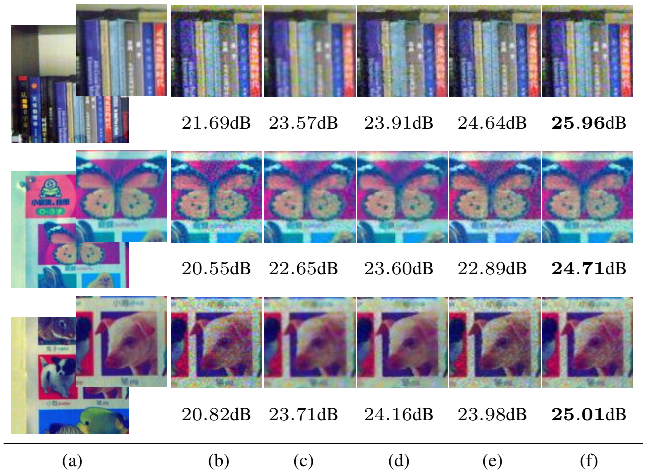

Fig. 2. The results of denoising images with natural noise. (a-b) Noiseless and noisy images. (c-f) Denoised images using CTV [8], PL [10], LPDEs [7] and LCPDEs, respectively. The PSNRs are presented below each image.

4.2. Comparison of training error and time

As both LPDEs [7] and our proposed LCPDEs have the training process and they have the same training error formulation (the objective function without regularization), we compare the training error and training time between LPDEs [7] and LCPDEs in this subsection. The training error is measured by (1) with $M = 60$.

Table 3. Comparison of training error and training time of LPDEs and our proposed LCPDEs.

Error		Time (s)		
LPDEs	LCPDEs	LPDEs	LCPDEs	Improve
4806	**4491**	43349	**515**	$O(10^2)$

The training error and time of LPDEs [7] and LCPDEs are shown in Table 3. Obviously, the training time is greatly reduced by LCPDEs. We can see that the training speed is accelerated by hundred times and the training errors of LCPDEs is lower than those of LPDEs in the meanwhile. So we conclude that our method is much more effective on learning PDEs for color images than LPDEs.

5. CONCLUSION

In this paper, we mainly focus on LPDEs for color images. A learning-based compact PDEs (LCPDEs) model is proposed by omitting the PDE which works as an indicate function and changing to L1-norm regularization on the coefficients. We resort to the proximal algorithm for learning the compact PDEs, which is more effective than LPDEs [7]. In the future, we hope to apply the LCPDEs to solve some high level tasks.

6. REFERENCES

[1] Gilles Aubert and Pierre Kornprobst, *Mathematical problems in image processing: partial differential equations and the calculus of variations*, vol. 147, Springer, 2006.

[2] Guillermo Sapiro, *Geometric partial differential equations and image analysis*, Cambridge university press, 2006.

[3] Pietro Perona and Jitendra Malik, "Scale-space and edge detection using anisotropic diffusion," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 12, no. 7, pp. 629–639, 1990.

[4] Stanley Osher and Leonid I Rudin, "Feature-oriented image enhancement using shock filters," *SIAM Journal on Numerical Analysis*, vol. 27, no. 4, pp. 919–940, 1990.

[5] Chunming Li, Chenyang Xu, Changfeng Gui, and Martin D Fox, "Level set evolution without re-initialization: a new variational formulation," in *Computer Vision and Pattern Recognition*. IEEE, 2005, vol. 1, pp. 430–436.

[6] Steven S. Beauchemin and John L. Barron, "The computation of optical flow," *ACM Computing Surveys (CSUR)*, vol. 27, no. 3, pp. 433–466, 1995.

[7] Risheng Liu, Zhouchen Lin, Wei Zhang, Kewei Tang, and Zhixun Su, "Toward designing intelligent PDEs for computer vision: An optimal control approach," *Image and Vision Computing*, vol. 31, no. 1, pp. 43–56, 2013.

[8] P. Blomgren and T F Chan, "Color TV: total variation methods for restoration of vector-valued images.," *Image Processing, IEEE Transactions on*, vol. 7, no. 3, pp. 304 – 309, 1998.

[9] Ivan W Selesnick, "Generalized total variation: Tying the knots," *IEEE Signal Processing Letters*, vol. 22, no. 11, pp. 2009–2013, 2015.

[10] Matthias Joachim Ehrhardt and Simon R. Arridge, "Vector-valued image processing by parallel level sets," *Image Processing, IEEE Transactions on*, vol. 23, no. 1, pp. 9–18, 2014.

[11] Risheng Liu, Zhouchen Lin, Wei Zhang, Kewei Tang, and Zhixun Su, "Learning PDEs for image restoration via optimal control," in *ECCV*, 2010.

[12] Risheng Liu, Junjie Cao, Zhouchen Lin, and Shiguang Shan, "Adaptive partial differential equation learning for visual saliency detection," in *CVPR*, 2014.

[13] Risheng Liu, Guangyu Zhong, Junjie Cao, Zhouchen Lin, Shiguang Shan, and Zhongxuan Luo, "Learning to diffuse: A new perspective to design pdes for visual analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 12, pp. 2457–2471, 2016.

[14] Zhenyu Zhao, Cong Fang, Zhouchen Lin, and Yi Wu, "A robust hybrid method for text detection in natural scenes by learning-based partial differential equations," *Neurocomputing*, vol. 168, pp. 23–34, 2015.

[15] Cong Fang, Zhenyu Zhao, Pan Zhou, and Zhouchen Lin, "Feature learning via partial differential equation with applications to face recognition," *Pattern Recognition*, vol. 69, no. C, pp. 14–25, 2017.

[16] John A Walker, *Dynamical Systems and Evolution Equations: Theory and Applications*, vol. 20, Springer Science & Business Media, 2013.

[17] Hedy Attouch, "On the convergence of the proximal algorithm for nonsmooth functions involving analytic features," *Mathematical Programming*, vol. 116, no. 1-2, pp. 5–16, 2009.

[18] Neal Parikh and Stephen Boyd, "Proximal algorithms," *Optimization*, vol. 1, no. vember, pp. 123–231, 2013.

[19] Eberhard Zeidler, *Nonlinear Functional Analysis and Its Applications: III: Variational Methods and Optimization*, Springer Science & Business Media, 2013.

[20] Jacques Louis Lions, *Optimal control of systems governed by partial differential equations*, vol. 170, Springer Verlag, 1971.

[21] Amir Beck and Marc Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM journal on imaging sciences*, vol. 2, no. 1, pp. 183–202, 2009.

[22] Zhou Wang, Alan Conrad Bovik, Hamid Rahim Sheikh, and Eero P Simoncelli, "Image quality assessment: from error visibility to structural similarity," *Image Processing, IEEE Transactions on*, vol. 13, no. 4, pp. 600–612, 2004.