ON MATHEMATICAL MODELING IN IMAGE RECONSTRUCTION AND BEYOND

BIN DONG

ABSTRACT

Imaging has been playing a vital role in the development of natural sciences. Advances in sensory, information, and computer technologies have further extended the scope of influence of imaging, making digital images an essential component of our daily lives. Image reconstruction is one of the most fundamental problems in imaging. For the past three decades, we have witnessed phenomenal developments of mathematical models and algorithms in image reconstruction. In this paper, we will first review some progress of the two prevailing mathematical approaches, i.e., the wavelet frame-based and PDE-based approaches, for image reconstruction. We shall discuss the connections between the two approaches and the implications and impact of the connections. Furthermore, we will review how the studies of the links between the two approaches lead us to a mathematical understanding of deep convolutional neural networks, which has led to further developments in modeling and algorithmic design in deep learning and new applications of machine learning in scientific computing.

ICM INTERNATIONAL CONGRESS OF MATHEMATICIANS 2022 JULY 6—14 SAINT PETERSBURG

Published by EMS Press. DOI 10.4171/ICM2022/2
Proc. Int. Cong. Math. 2022, Vol. ?, pp. 2–30

1. INTRODUCTION

The development of natural sciences has been heavily relying on visual examinations. Through observations on natural phenomenons made by our naked eyes or via instruments such as cameras, microscopes, telescopes, etc., scientists make a scientific hypothesis on the underlying principles hidden in the phenomenon, and they later conduct more experiments or resort to mathematical deductions to further verify their hypothesis. Therefore, images play a central role since they can accurately record the phenomenon of interest and be further processed and analyzed by algorithms to assist human decision-making. In the past few decades, we are experiencing rapid advances in information technology, which contribute significantly to the exponential growth of data. Digital images are of no doubt one of the essential components of data. Advanced computer technology has made it possible to apply some of the most sophisticated developments in mathematics and machine learning to the design and implementation of efficient algorithms to process and analyze image data. As a result, the impact of images has now gone far beyond natural sciences. Image processing and analysis techniques are now widely adopted in engineering, medicine, technical disciplines, and social media, and digital images have become an essential element of our daily lives.

Among all tasks within the scope of computer vision, image reconstruction, such as image denoising, deblurring, inpainting, medical imaging, etc., is one of the most fundamental ones. Its objective is to obtain high-quality reconstructions of images that are corrupted in various ways during the process of acquisition, storage, and communication, and enable us to see crucial but subtle objects that reside in the images. Mathematics has been the main driven force in the advancement of image reconstruction for the past few decades [7, 33, 53]. Conversely, image reconstruction also brings to mathematics new challenging problems and fascinating applications that gave birth to new mathematical tools, whose application has even gone beyond the scope of image reconstruction.

Image reconstruction can be formulated as the following inverse problem:

$$\boldsymbol{f} = \boldsymbol{A}\boldsymbol{u} + \boldsymbol{\eta}. \tag{1.1}$$

Here, $\boldsymbol{A}$ is a linear operator corresponding to the imaging process. For example, $\boldsymbol{A}$ is an identity operator for image denoising; a convolution operator for image deblurring; a restriction operator for image inpainting [13]; a subsampled Fourier transform for magnetic resonance imaging (MRI) [19]; a subsampled Radon transform for X-ray-based computed tomography (CT) [22]. Variable $\boldsymbol{u}$ is the unknown image to be reconstructed, and $\boldsymbol{f}$ is the measurements that are contaminated by additive noise $\boldsymbol{\eta}$ with known or partially known statistics, e.g., Gaussian, Laplacian, Poisson, etc. The main challenge in solving the linear inverse problem (1.1) is the ill-posedness of the problem. A naive inversion of $\boldsymbol{A}$, such as pseudoinversion or via Tikhonov regularization [154], may result in a reconstructed image with amplified noise and smeared-out edges.

Many existing image reconstruction models and algorithms are transformation-based. One of the earliest transforms was the Fourier transform, which is effective on signals that are smooth and sinusoidal-like. However, the Fourier transform is not adequate on images with multiple localized frequency components. Windowed Fourier transforms [72]

were introduced to overcome the poor spatial localization of the Fourier transform. However, the high-frequency coefficients in the transform domain are not ideally sparse for images due to the fixed time-frequency resolution of the windowed Fourier transforms. This is why wavelets and wavelet frames are much more effective for images than Fourier or windowed Fourier transforms because of their varied time-frequency resolution, which enables them to provide a better sparse approximation to piecewise smooth functions [45, 51, 110].

Another influential class of methods for image reconstruction that have been developed through a rather different path from wavelets is the PDE-based approach [33, 119, 136], which includes variational and (nonlinear) PDE methods. The basic idea of variational methods is to characterize images as functions living in a certain function space, such as the BV space [115, 131] (space of functions with bounded variations), and an energy functional is designed according to the function space assumption. PDE methods, on the other hand, often take the observed low-quality image or a coarsely reconstructed image as the initialization and enhance it by evolving a carefully designed nonlinear PDE that conducts smoothing in homogeneous regions and edge-preservation or enhancement near edges [120, 123].

The two approaches have been developing independently for decades. Although studies were showing the links between the two approaches [84, 148] using specific models and algorithms, their general connections were still unknown. Later in [24, 26, 42, 52], fundamental connections between wavelet frame-based approach and variational methods were established. Connections of wavelet frame-based models to the total variation model were established in [24], to the Mumford–Shah model were established in [26], and to some more general variational models such as the total generalized variation model [18] were established in [42, 52]. On the other hand, [49] established a generic connection between iterative wavelet frame shrinkage and general nonlinear evolution PDEs. We showed that wavelet frame shrinkage algorithms could be viewed as discrete approximations of nonlinear evolution PDEs. Such connection led to new understandings of both the wavelet frame- and PDE-based approach and expanded the scope of applications for both. The series of papers [24, 26, 42, 49, 52] essentially merged the two seemingly unrelated areas: wavelet frame-based and PDE-based approach for image reconstruction, and gave birth to many new image reconstruction models and algorithms.

For the past decade, the landscape of research and technological development of image reconstruction and computer vision is experiencing a significant transformation due to the advances in machine learning, especially deep learning [71, 91, 145]. A new set of models call the convolutional neural networks (CNNs) [65, 92] were introduced, where the AlexNet [89], U-Net [130], ResNet [77], and DenseNet [79] are well-known examples. Most CNNs have millions to billions of parameters that are trained (or optimized) on large data sets via stochastic algorithms. One remarkable property of deep neural networks (DNNs) in general is that they can well approximate nonlinear functions in high-dimensional spaces without suffering from the curse of dimensionality [36, 104, 114, 142–144, 163, 164, 170]. CNNs were first shown to be extremely effective in image classification [77, 89]. They were later adopted in image reconstruction and significantly advanced its state-of-the-art (see, e.g., [38, 113, 156, 159, 172]).

Why CNNs perform so well in practice and where their capability boundary locates is arguably the biggest mystery in deep learning for the moment. One possible way of unraveling such mystery, at least for image reconstruction, is to explore the connection between CNNs and mathematical models we now have a systematic understanding of. More importantly, what do CNNs do differently to outperform these mathematical models significantly, and can we combine the wisdom from both sides? Answering these questions can bring new insights into CNN models and further extend the scope of their applications.

Let $\mathcal{F}$ be an image reconstruction operator for the problem (1.1) that takes a coarse reconstruction of the image as input and the reconstructed image as output. For both wavelet frame-based and PDE-based models, this mapping $\mathcal{F}$ is a discrete dynamical system. As shown by **[49]**, most of these discrete dynamical systems are various discrete approximations to differential equations. CNNs, on the other hand, are formed by consecutive compositions of relatively simple functions, which makes them discrete dynamical systems as well. We use $\mathcal{F}_{\boldsymbol{\Theta}}$ to denote a CNN, which is a parameterized dynamical system. One apparent difference between $\mathcal{F}$ and $\mathcal{F}_{\boldsymbol{\Theta}}$ is that the former is entirely design-based using human knowledge while the latter has minimal human design and its actual form mostly relies on a large number of parameters $\boldsymbol{\Theta}$ that are optimized through empirical risk minimization. The dynamics $\mathcal{F}$ and $\mathcal{F}_{\boldsymbol{\Theta}}$ are two extremes of modeling where the former advocates human knowledge, which grants solid theoretical foundations and adequate interpretability, while the latter promotes data-driven modeling which can extract features and principles from data that may be unknown to humans to better assist in decision making. However, in practice, neither extreme is ideal, which is especially the case in science, economics, and medicine. In these disciplines, interpretability is mostly required. Also, we have some knowledge to describe a particular phenomenon but still largely not enough, and we have observational or simulation data but limited in quantity. Therefore, we need to balance between the two extremes depending on the specific application of interest. Finding connections between $\mathcal{F}$ and $\mathcal{F}_{\boldsymbol{\Theta}}$ may better assist us in this regard.

This motivated us to study connections between CNNs and differential equations. From the standpoint of dynamical systems, we explored the structural similarities between numerical differential equations and CNNs in **[101,102,107]**. In **[107]**, we showed that not only ResNet could be viewed as a forward-Euler approximation to differential equations as first pointed out by **[74,162]**, but many other CNNs with bypass structures (or skip connections) can also be viewed as a discrete approximation of differential equations. Furthermore, **[107]** was the first to draw connections between residual-type CNNs with random perturbations and stochastic differential equations (SDEs). In fact, **[107]** suggested numerical ODEs/SDEs as a systematic framework for designing CNNs for image classification. In **[101, 102]**, we were among the earliest to explore the structural similarity between CNNs and numerical PDEs. The key to such structural similarity is also the key to the connections between wavelet frame-based and PDE-based approaches for image reconstruction. By exploiting such structural similarity, we proposed a set of new CNNs called PDE-Nets, which can estimate the analytical form of (time-dependent) PDEs from observed dynamical data with minor prior knowledge on the underlying mechanism that drives the dynamics. Once trained,

the PDE-Net also serves as a simulator that can generate more dynamical data accurately and efficiently.

This paper will review the development of the wavelet frame-based and PDE-based approaches for image reconstruction. We shall discuss the connections between the two approaches and demonstrate how the connections lead to new models for image reconstruction. Furthermore, we will show how these theoretical studies inspired our exploration of structural similarities between differential equations and CNNs. These findings lead to further developments in modeling and algorithmic design in deep learning and new applications of machine learning in scientific computing.

2. WAVELET FRAME-BASED APPROACH FOR IMAGE RECONSTRUCTION

We start with a brief introduction to the concept of wavelet frame transform in a discrete setting. The interested readers should consult **[45, 46, 128, 129]** for theories of frames and wavelet frames, **[51, 140]** for a short survey on the theory and applications of frames, and **[53]** for a more detailed survey.

In the discrete setting, let an image $\boldsymbol{f}$ be a d-dimensional array. We denote by $\mathcal{I}_d = \mathbb{R}^{N_1 \times N_2 \times \cdots \times N_d}$ the set of all d-dimensional images. We denote the d-dimensional fast $(L+1)$-level wavelet frame transform/decomposition with filters $\{\boldsymbol{q}^{(0)}, \boldsymbol{q}^{(1)}, \ldots, \boldsymbol{q}^{(r)}\}$ (see, e.g., **[53]**) by

$$\boldsymbol{W}\boldsymbol{u} = \left\{\boldsymbol{W}_{\ell,l}\boldsymbol{u} : (\ell, l) \in \mathbb{B}\right\}, \quad \boldsymbol{u} \in \mathcal{I}_d, \tag{2.1}$$

where $\mathbb{B} = \{(\ell, l) : 1 \le \ell \le r,\ 0 \le l \le L\} \cup \{(0, L)\}$. The wavelet frame coefficients $\boldsymbol{W}_{\ell,l}\boldsymbol{u} \in \mathcal{I}_d$ are computed by $\boldsymbol{W}_{\ell,l}\boldsymbol{u} = \boldsymbol{q}_{\ell,l}[-\cdot] \circledast \boldsymbol{u}$, where $\circledast$ denotes the convolution operator with a certain boundary condition, e.g., periodic boundary condition, and $\boldsymbol{q}_{\ell,l}$ is defined as

$$\boldsymbol{q}_{\ell,l} = \breve{\boldsymbol{q}}_{\ell,l} \circledast \breve{\boldsymbol{q}}_{l-1,0} \circledast \cdots \circledast \breve{\boldsymbol{q}}_{0,0} \quad \text{with} \quad \breve{\boldsymbol{q}}_{\ell,l}[\boldsymbol{k}] = \begin{cases} \boldsymbol{q}_\ell[2^{-l}\boldsymbol{k}], & \boldsymbol{k} \in 2^l\mathbb{Z}^d, \\ 0, & \boldsymbol{k} \notin 2^l\mathbb{Z}^d. \end{cases} \tag{2.2}$$

Similarly, we can define $\tilde{\boldsymbol{W}}\boldsymbol{u}$ and $\tilde{\boldsymbol{W}}_{\ell,l}\boldsymbol{u}$ given a set of dual filters $\{\tilde{\boldsymbol{p}}, \tilde{\boldsymbol{q}}_1, \ldots, \tilde{\boldsymbol{q}}_r\}$. We denote the inverse wavelet frame transform (or wavelet frame reconstruction) as $\tilde{\boldsymbol{W}}^\top$, which is the adjoint operator of $\tilde{\boldsymbol{W}}$. When the primal filters $\{\boldsymbol{p}, \boldsymbol{q}^{(1)}, \ldots, \boldsymbol{q}^{(r)}\}$ and dual filters $\{\tilde{\boldsymbol{p}}, \tilde{\boldsymbol{q}}_1, \ldots, \tilde{\boldsymbol{q}}_r\}$ satisfy the extension principles **[128, 129]**, we have the perfect reconstruction formula

$$\boldsymbol{u} = \tilde{\boldsymbol{W}}^\top \boldsymbol{W}\boldsymbol{u}, \quad \text{for all } \boldsymbol{u} \in \mathcal{I}_d.$$

In particular, when the dual filters are the same as the primal filters with the extension principle satisfied, $\boldsymbol{W}$ is the transform associated to a tight frame system, and we simply have that

$$\boldsymbol{u} = \boldsymbol{W}^\top \boldsymbol{W}\boldsymbol{u}, \quad \text{for all } \boldsymbol{u} \in \mathcal{I}_d. \tag{2.3}$$

For simplicity, we will mostly focus our discussions on the case $d = 2$.

Two simple but useful examples of filters for univariate tight frame systems, i.e., Haar and piecewise linear tight frame system, constructed from B-splines [129] are given as follows.

Example 2.1. Filters of B-spline tight frame systems.

(1) *Haar*. Let $\boldsymbol{p} = \frac{1}{2}[1, 1]$ be the refinement mask of the piecewise constant B-spline $B_1(x) = 1$ for $x \in [0, 1]$ and 0 otherwise. Define $\boldsymbol{q}_1 = \frac{1}{2}[1, -1]$.

(2) *Piecewise linear*. Let $\boldsymbol{p} = \frac{1}{4}[1, 2, 1]$ be the refinement mask of the piecewise linear B-spline $B_2(x) = \max(1 - |x|, 0)$. Define $\boldsymbol{q}_1 = \frac{\sqrt{2}}{4}[1, 0, -1]$ and $\boldsymbol{q}_2 = \frac{1}{4}[-1, 2, -1]$.

The key to the success of wavelet frames in image reconstruction is their capability to provide a sparse approximation to images. In other words, the high-frequency band $\mathbb{B}\backslash\{(0, L)\}$ of the wavelet frame transform $\boldsymbol{W}\boldsymbol{u}$ of a typical image $\boldsymbol{u}$ is sparse. Large (in magnitude) wavelet frame coefficients encode image features such as edges, while the coefficients are small in smooth regions. This is mainly due to the short support and high order of vanishing moments of wavelet frames that make them behave like differential operators (we will come back to this in Section 4).

Wavelet frame-based image reconstruction started from the seminal work [32]. The basic idea is as follows: Consider the linear inverse problem (1.1). After an initial reconstruction of the image $\boldsymbol{u}$, edges might be blurred, and noise is still present in the image. Since a clean image should be sparse in the wavelet frame domain, one of the simplest ways to sharpen the image and remove noise at the same time is to set small high-frequency coefficients to zero. When we reconstruct the image using the processed wavelet frame coefficients, it will no longer be consistent with the data, i.e., $\boldsymbol{A}\boldsymbol{u}$ may be far away from $\boldsymbol{f}$. The simplest way to correct it is by moving $\boldsymbol{u}$ closer to the hyperplane $\boldsymbol{A}\boldsymbol{u} = \boldsymbol{f}$. Then, we iterate this procedure till convergence. This leads to a wavelet frame-based iterative algorithm, which was later analyzed by [23] and revealed its relation to the following wavelet frame-based balanced model:

$$\min_{\boldsymbol{d}} \frac{1}{2}\left\|\boldsymbol{A}\boldsymbol{W}^{\top}\boldsymbol{d} - \boldsymbol{f}\right\|_2^2 + \frac{\kappa}{2}\left\|(\boldsymbol{I} - \boldsymbol{W}\boldsymbol{W}^{\top})\boldsymbol{d}\right\|_2^2 + \|\boldsymbol{\lambda} \cdot \boldsymbol{d}\|_1. \tag{2.4}$$

The balanced model also takes the analysis model [25, 55, 147]

$$\min_{\boldsymbol{u}} \frac{1}{2}\|\boldsymbol{A}\boldsymbol{u} - \boldsymbol{f}\|_2^2 + \|\boldsymbol{\lambda} \cdot \boldsymbol{W}\boldsymbol{u}\|_1, \tag{2.5}$$

and the synthesis model [47, 59, 60, 63, 64]

$$\min_{\boldsymbol{d}} \frac{1}{2}\left\|\boldsymbol{A}\boldsymbol{W}^{\top}\boldsymbol{d} - \boldsymbol{f}\right\|_2^2 + \|\boldsymbol{\lambda} \cdot \boldsymbol{d}\|_1 \tag{2.6}$$

as special cases. The balanced, analysis and synthesis models (and their variants) are among the most commonly used models in image reconstruction.

The objective functions in (2.4)–(2.6) are all convex, and can be efficiently optimized by convex optimization algorithms. For example, both the balanced and synthesis models can be solved efficiently by proximal forward–backward splitting (PFBS) [20, 35, 44,

[122, 155] and can be further accelerated by Nesterov's approach [10, 141]. The analysis model can be solved efficiently using the alternating direction method of multipliers (ADMM) [16, 25, 66, 68, 69] and the primal dual hybrid gradient (PDHG) method [30, 58, 176].

3. PDE-BASED APPROACH FOR IMAGE RECONSTRUCTION

In the past few decades, many variational and PDE models have been proposed with success in different tasks in image reconstruction. In this section, we shall refer to them both as the PDE-based approach. Successful examples of the PDE-based approach include the total variation (TV) model [131], total generalized variation model [18], Mumford–Shah model [115], shock-filter [120], Perona–Malik (PM) equation [123], anisotropic diffusion models [161], fluid dynamics model [12], etc. In this section, we will recall the TV model and the PM equation.

Regularization is crucial in solving ill-posed inverse problems. In 1963, Tikhonov proposed the so-called Tikhonov regularization [154] that penalizes the H^1 seminorm of the image to be reconstructed. Tikhonov regularization can effectively remove noise while it smears out important image features such as edges as well. This is essentially because H^1 is not an appropriate function space to model images. It has such a strong regularity requirement that functions with jump discontinuities are not allowed in the function space. To overcome such drawbacks, Rudin, Osher, and Fatemi proposed the refined TV model that penalizes the total variation of the function to be reconstructed so that jump discontinuities can be well-preserved and noise can be adequately removed. This is because the BV space is large enough to include functions with discontinuities but not too large, so that noise is still excluded.

Now, we first recall the definition of TV and the BV space. Let $\Omega \subset \mathbb{R}^2$ be an open set and $u \in L_1(\Omega)$. Then, the total variation of u is defined as

$$\mathrm{TV}(u) := \sup \left\{ \int_\Omega u \operatorname{div} v \, \mathrm{d}x : v \in C_c^1(\Omega, \mathbb{R}^2),\ \|v\|_{L_\infty(\Omega)} \le 1 \right\}, \tag{3.1}$$

where $C_c^1(\Omega, \mathbb{R}^2)$ is the space of all compactly supported continuously differentiable functions on Ω. Another convenient notation for the TV of a function u is $\mathrm{TV}(u) = \int_\Omega |Du(x)| \mathrm{d}x$, where Du is the distributional derivative of u. Intuitively speaking, the TV of a function u records the total amount of fluctuation of the function on domain Ω. If u is differentiable, then $\mathrm{TV}(u) = \int_\Omega |\nabla u(x)| \, \mathrm{d}x$. We define the BV space as

$$\mathrm{BV}(\Omega) = \{u \in L_1(\Omega) : \mathrm{TV}(u) < +\infty\}.$$

We now consider the function version of the image reconstruction problem (1.1), namely

$$f = Au + \eta.$$

We use nonbold characters to denote functions and linear operators in contrast to the bold characters that denote arrays and matrices. Then, the TV model for image reconstruction

reads as follows:

$$\min_{u\in \mathrm{BV}} \mathrm{TV}(u) + \frac{\lambda}{2}\int_{\Omega} \big(Au(x) - f(x)\big)^2 \mathrm{d}x, \tag{3.2}$$

where $\lambda > 0$ is a preselected hyperparameter that balances the amount of regularization from the first term and data consistency from the second term. Ways of solving the TV model include solving the associated Euler–Lagrange equation or the gradient flow, or we can discretize the model first and then use a convex optimization algorithm (e.g., one of those described in the previous section). Note that (3.2) is similar in form to (2.5). The difference is that in (2.5), we penalize the ℓ_1-norm of the wavelet frame transform of $\boldsymbol{u}$, while in (3.2) we penalize the L_1-norm of Du. This is an indication that (3.2) and (2.5) may be closely related. For convenience, we shall call the variational model (3.2) and its variants as differential operator-based analysis model, and (2.5) the wavelet frame-based analysis model.

In contrast to variational models for image reconstruction, designing PDE models is less restrictive and more intuitive to incorporate local geometric structures of images in the design. The scale-space theory tells us that using PDEs to model image reconstruction is a reasonable option. Let us use a set of nonlinear operators $\{\boldsymbol{T}_t\}_{t\geq 0}$ with $u(t,x) = (T_t u_0)(x)$ to denote the flow of image reconstruction starting from an initial estimation $u_0(x)$. If the set of operators satisfies certain axioms, such as recursivity, regularity, locality, translation invariance, etc., then there exists a second-order nonlinear evolution PDE such that $u(t,x)$ is its viscosity solution **[3]**. The PM equation is one of the well-known PDE models that are effective in image reconstruction (originally for image denoising but can be extended to general image reconstruction problems). It imposes a different amount of diffusion, even backward diffusion, in different regions of the images depending on local regularity and the orientation of edges. In the following, we will recall the idea of the original design of the PM equation for image denoising. Interested readers should consult **[7, 123]** for more details.

Given an observed noisy image $u_0(x)$, the PM equation takes the following form:

$$\begin{cases} u_t = \mathrm{div}\big(g\big(|\nabla u|^2\big)\nabla u\big), & \text{on } (0,T)\times\Omega, \\ \frac{\partial u}{\partial \vec{n}}(t,x) = 0, & \text{on } (0,T)\times\partial\Omega, \\ u(0,x) = u_0(x), & \text{on } \Omega, \end{cases}$$

where the diffusivity function g is a scalar function satisfying

$$\begin{cases} g : [0,\infty) \mapsto (0,\infty) \text{ is monotonically decreasing;} \\ g(0) = 1; \qquad g(x) \to 0 \quad \text{as} \quad x \to \infty; \\ g(x) + 2xg'(x) > 0 \quad \text{for } x \leq K; \qquad g(x) + 2xg'(x) < 0 \quad \text{for } x > K. \end{cases} \tag{3.3}$$

The specific design of the diffusivity function g is to impose not only a spatially variant diffusion, but also different amount of diffusion in different directions at any given location. Commonly used examples of the diffusivity function g include

$$g(s) = e^{-\frac{s}{2\sigma^2}}, \quad \text{or} \quad g(s) = \frac{1}{1 + s^p/\lambda^2}, \quad p > \frac{1}{2},\ \lambda > 0.$$

From (3.3), we can see that $g(|\nabla u|^2)$ is relatively large at smooth regions of the image u where $|\nabla u|$ is relatively small. Thus, the PM equation applies stronger smoothing

in smooth regions of the image. In contrast, $|\nabla u|$ is relatively large near edges, and hence $g(|\nabla u|^2)$ is relatively small. Then, the PM equation applies less smoothing near edges which can reduce the amount of blurring. On the other hand, if we decompose the PM equation along the tangential and normal direction of the level sets of u, we can rewrite the original PM equation as

$$u_t = g\big(|\nabla u|^2\big)u_{TT} + \tilde{g}\big(|\nabla u|^2\big)u_{NN},$$

with

$$\tilde{g}(x) = g(x) + 2xg'(x), \quad N = \frac{\nabla u}{|\nabla u|}, \quad \text{and} \quad T = N^{\perp}, \quad |T| = 1.$$

Here, T and N are two unit vector fields that record, respectively, the tangential and normal directions of the level sets of function u. Further, u_{TT} and u_{NN} are the second-order derivatives along the tangential direction T and normal direction N, respectively. We can see from (3.3) that the PM equation imposes forward diffusion along the tangential direction to remove noise, while imposing backward diffusion along the normal direction near edges for enhancement. This, however, makes the PM equation an ill-posed PDE. This problem was later resolved by **[28]** where a modification of the PM equation was proposed and analyzed.

4. CONNECTIONS BETWEEN WAVELET FRAME-BASED AND PDE-BASED APPROACHES

In this section, we will summarize the main findings from the work **[24]** that established the connections between the differential operator-based and wavelet frame-based analysis models, and the work **[49]** that established the connections between nonlinear evolution PDEs and iterative wavelet frame-based shrinkage algorithms. Extensions of these results can be found in **[26, 42, 52]**.

4.1. Wavelet frame transform and differential operators

Wavelet frame transform is a collection of convolution operators with both low- and high-pass filters. For a given multiresolution analysis (MRA) based wavelet frame system, the low-pass filters are associated with the refinable functions, while the high-pass filters are associated with wavelet functions. Key properties of both refinable and wavelet functions, such as smoothness and vanishing moments, can be characterized by their associated filters. The key observation that eventually leads to the connections between wavelet frame and PDE-based approaches is the link between vanishing moments of wavelet functions and differential operators in discrete and continuum settings. This observation was first made in **[24]** and was further exploited in **[26, 42, 49, 52]**.

For a high-pass filter $\boldsymbol{q}$, let $\widehat{\boldsymbol{q}}(\boldsymbol{\omega}) = \sum_{\boldsymbol{k}\in\mathbb{Z}^2} \boldsymbol{q}[\boldsymbol{k}]e^{-i\boldsymbol{k}\boldsymbol{\omega}}$ be its two-scale symbol. Throughout this paper, for a multiindex $\boldsymbol{\alpha} = (\alpha_1, \alpha_2) \in \mathbb{Z}_+^2$ and $\boldsymbol{\omega} \in \mathbb{R}^2$, write

$$\boldsymbol{\alpha}! = \alpha_1!\alpha_2!, \quad |\boldsymbol{\alpha}| = \alpha_1 + \alpha_2, \quad \boldsymbol{D}_{\boldsymbol{\alpha}} = \frac{\partial^{\boldsymbol{\alpha}}}{\partial \boldsymbol{\omega}^{\boldsymbol{\alpha}}} = \frac{\partial^{\alpha_1+\alpha_2}}{\partial \omega_2^{\alpha_2} \partial \omega_1^{\alpha_1}}.$$

We say that $\boldsymbol{q}$ (and $\widehat{\boldsymbol{q}}(\boldsymbol{\omega})$) has vanishing moments of order $\boldsymbol{\alpha} = (\alpha_1, \alpha_2)$, where $\boldsymbol{\alpha} \in \mathbb{Z}_+^2$, provided that

$$\sum_{\boldsymbol{k}\in\mathbb{Z}^2} \boldsymbol{k}^{\boldsymbol{\beta}} \boldsymbol{q}[\boldsymbol{k}] = i^{|\boldsymbol{\beta}|} \frac{\partial^{\boldsymbol{\beta}}}{\partial \boldsymbol{\omega}^{\boldsymbol{\beta}}} \widehat{\boldsymbol{q}}(\boldsymbol{\omega})\Big|_{\boldsymbol{\omega}=\boldsymbol{0}} = 0 \tag{4.1}$$

for all $\boldsymbol{\beta} \in \mathbb{Z}_+^2$ with $|\boldsymbol{\beta}| < |\boldsymbol{\alpha}|$ and for all $\boldsymbol{\beta} \in \mathbb{Z}_+^2$ with $|\boldsymbol{\beta}| = |\boldsymbol{\alpha}|$ but $\boldsymbol{\beta} \neq \boldsymbol{\alpha}$. We say that $\boldsymbol{q}$ has total vanishing moments of order K with $K \in \mathbb{Z}_+$ if (4.1) holds for all $\boldsymbol{\beta} \in \mathbb{Z}_+^2$ with $|\boldsymbol{\beta}| < K$. Suppose $K \geq 1$. If (4.1) holds for all $\boldsymbol{\beta} \in \mathbb{Z}_+^2$ with $|\boldsymbol{\beta}| < K$ except for $\boldsymbol{\beta} \neq \boldsymbol{\beta}_0$ with certain $\boldsymbol{\beta}_0 \in \mathbb{Z}_+^2$ and $|\boldsymbol{\beta}_0| = J < K$, then we say that $\boldsymbol{q}$ has total vanishing moments of order $K\backslash\{J+1\}$.

To have a better understanding of the concept of vanishing moments, let us look at one example. Let $\widehat{\boldsymbol{q}}_1(\boldsymbol{\omega}) = e^{i\omega_1} - e^{-i\omega_1}$, which is the first high-pass filter of the piecewise linear B-spline tight wavelet frame system in Example 2.1. Then,

$$\widehat{\boldsymbol{q}}_1(\boldsymbol{0}) = 0, \quad \frac{\partial}{\partial\omega_1}\widehat{\boldsymbol{q}}_1(\boldsymbol{0}) = 2i \neq 0, \quad \frac{\partial}{\partial\omega_2}\widehat{\boldsymbol{q}}_1(\boldsymbol{0}) = 0.$$

Thus $\widehat{\boldsymbol{q}}_1(\boldsymbol{\omega})$ has vanishing moments of order $(1,0)$. In addition, we have

$$\frac{\partial^2}{\partial\omega_1^2}\widehat{\boldsymbol{q}}_1(\boldsymbol{0}) = 0, \quad \frac{\partial^2}{\partial\omega_1\partial\omega_2}\widehat{\boldsymbol{q}}_1(\boldsymbol{0}) = 0, \quad \frac{\partial^2}{\partial\omega_2^2}\widehat{\boldsymbol{q}}_1(\boldsymbol{0}) = 0.$$

Therefore, $\boldsymbol{q}_1$ has total vanishing moments of order $3\backslash\{|(1,0)|+1\}$, or $3\backslash\{2\}$ (it does not have total vanishing moments of order $4\backslash\{2\}$ since $\frac{\partial^3}{\partial\omega_1^3}\widehat{\boldsymbol{q}}_1(\boldsymbol{0}) = -2i \neq 0$).

The following proposition from **[49]** describes the relation between the vanishing moments of high-pass filters and finite difference approximations of differential operators. This proposition was later applied to the work of PDE-Net **[101,102]** that explores and exploits structure similarities between deep convolutional neural networks and numerical PDEs.

Proposition 4.1. *Let $\boldsymbol{q}$ be a high-pass filter with vanishing moments of order $\boldsymbol{\alpha} \in \mathbb{Z}_+^2$. Then for a smooth function $F(\boldsymbol{x})$ on $\mathbb{R}^2$, we have*

$$\frac{1}{\varepsilon^{|\boldsymbol{\alpha}|}} \sum_{\boldsymbol{k}\in\mathbb{Z}^2} \boldsymbol{q}[\boldsymbol{k}] F(\boldsymbol{x} + \varepsilon\boldsymbol{k}) = C_{\boldsymbol{\alpha}} \frac{\partial^{\boldsymbol{\alpha}}}{\partial \boldsymbol{x}^{\boldsymbol{\alpha}}} F(\boldsymbol{x}) + O(\varepsilon),$$

where $C_{\boldsymbol{\alpha}}$ is the constant defined by

$$C_{\boldsymbol{\alpha}} = \frac{1}{\boldsymbol{\alpha}!} \sum_{\boldsymbol{k}\in\mathbb{Z}^2} \boldsymbol{k}^{\boldsymbol{\alpha}} \boldsymbol{q}[\boldsymbol{k}] = \frac{i^{|\boldsymbol{\alpha}|}}{\boldsymbol{\alpha}!} \frac{\partial^{\boldsymbol{\alpha}}}{\partial \boldsymbol{\omega}^{\boldsymbol{\alpha}}} \widehat{\boldsymbol{q}}(\boldsymbol{\omega})\Big|_{\boldsymbol{\omega}=\boldsymbol{0}}.$$

If, in addition, $\boldsymbol{q}$ has total vanishing moments of order $K\backslash\{|\boldsymbol{\alpha}|+1\}$ for some $K > |\boldsymbol{\alpha}|$, then

$$\frac{1}{\varepsilon^{|\boldsymbol{\alpha}|}} \sum_{\boldsymbol{k}\in\mathbb{Z}^2} \boldsymbol{q}[\boldsymbol{k}] F(\boldsymbol{x} + \varepsilon\boldsymbol{k}) = C_{\boldsymbol{\alpha}} \frac{\partial^{\boldsymbol{\alpha}}}{\partial \boldsymbol{x}^{\boldsymbol{\alpha}}} F(\boldsymbol{x}) + O(\varepsilon^{K-|\boldsymbol{\alpha}|}).$$

Similar results can be written in terms of wavelet frame functions which is given by the following proposition of **[42]**. Note that a version of the same result for B-splines wavelet frames was proposed earlier in **[24]**.

Proposition 4.2. *Let a tensor product wavelet frame function $\psi_{\boldsymbol{\alpha}} \in L_2(\mathbb{R}^2)$ have vanishing moments of order $\boldsymbol{\alpha}$ with $|\boldsymbol{\alpha}| \le s$, and let* $\operatorname{supp}(\psi_{\boldsymbol{\alpha}}) = [a_1, a_2] \times [b_1, b_2]$. *Then, there exists a unique $\varphi_{\boldsymbol{\alpha}} \in L_2(\mathbb{R}^2)$ such that $\varphi_{\boldsymbol{\alpha}}$ is differentiable up to order $\boldsymbol{\alpha}$ a.e.,*

$$c_{\boldsymbol{\alpha}} = \int_{\mathbb{R}^2} \varphi_{\boldsymbol{\alpha}} \neq 0 \quad \text{and} \quad \psi_{\boldsymbol{\alpha}} = \partial^{\boldsymbol{\alpha}} \varphi_{\boldsymbol{\alpha}}.$$

Furthermore, for $n \in \mathbb{N}$ and $\boldsymbol{k} \in \mathbb{Z}^2$ with $\operatorname{supp}(\psi_{\boldsymbol{\alpha},n-1,\boldsymbol{k}}) \subseteq \overline{\Omega}$, *we have*

$$\langle u, \psi_{\boldsymbol{\alpha},n-1,\boldsymbol{k}} \rangle = (-1)^{|\boldsymbol{\alpha}|} 2^{|\boldsymbol{\alpha}|(1-n)} \langle \partial^{\boldsymbol{\alpha}} u, \varphi_{\boldsymbol{\alpha},n-1,\boldsymbol{k}} \rangle$$

for every u belonging to the Sobolev space $W_1^s(\Omega)$. Here,

$$\psi_{\boldsymbol{\alpha},n-1,\boldsymbol{k}} = 2^{n-2} \psi_{\boldsymbol{\alpha}}\big(2^{n-1} \cdot - \boldsymbol{k}/2\big)$$

and $\varphi_{\boldsymbol{\alpha},n-1,\boldsymbol{k}}$ is defined similarly.

Note that Proposition 4.1 is more convenient to use in addressing the connections between wavelet frame shrinkage algorithm and nonlinear evolution PDEs. In contrast, Proposition 4.2 is more convenient to use in addressing the connections between wavelet frame-based and differential operator-based analysis models.

4.2. Connections between wavelet frame-based analysis model and TV model

The wavelet frame-based analysis model considered by **[24]** is given as

$$\inf_{u \in W_1^s(\Omega)} E_n(u) := \nu \|\boldsymbol{\lambda}_n \cdot \boldsymbol{W}\boldsymbol{T}_n u\|_1 + \frac{1}{2} \|\boldsymbol{A}_n \boldsymbol{T}_n u - \boldsymbol{T}_n f\|_2^2, \tag{4.2}$$

and the differential operator-based analysis model is given as

$$\inf_{u \in W_1^s(\Omega)} E(u) := \nu \|\boldsymbol{D}u\|_1 + \frac{1}{2} \|Au - f\|_{L_2(\Omega)}^2. \tag{4.3}$$

Here, $\boldsymbol{W}$ denotes the wavelet frame transform defined by (2.1) and (2.2), $\boldsymbol{T}_n$ is the sampling operator generated by the refinable function corresponding to the underlying wavelet frame system, $\boldsymbol{A}_n$ is a discrete approximation of the operator A, $\boldsymbol{D}$ is a certain linear differential operator with highest order s (e.g., for the TV model, $\boldsymbol{D} = \nabla$ and $s = 1$). We denote by $W_p^r(\Omega)$ the Sobolev space with functions whose rth order weak derivatives belong to $L_p(\Omega)$ and which is equipped with the norm $\|f\|_{W_p^r(\Omega)} := \sum_{|\boldsymbol{\alpha}| \le r} \|D_{\boldsymbol{\alpha}} f\|_p$.

From the form of E_n and E, we can see a similarity between the two functionals. It was proved in **[24]** that E_n converges to E pointwise on $W_1^s(\Omega)$. However, since we are interested in the (approximated) minimizers of these functionals, pointwise convergence does not guarantee a relation between their associated (approximated) minimizers. Therefore, Γ-convergence **[17]** was used in **[24]** to draw a connection between the problems $\min_u E_n(u)$ and $\min_u E(u)$. We first recall the definition of Γ-convergence.

Definition 4.1. Given $E_n(u) : W_1^s(\Omega) \mapsto \bar{\mathbb{R}}$ and $E(u) : W_1^s(\Omega) \mapsto \bar{\mathbb{R}}$, we say that E_n Γ-converges to E if:

(i) for every sequence $u_n \to u$ in $W_1^s(\Omega)$, $E(u) \le \liminf_{n\to\infty} E_n(u_n)$;

(ii) for every $u \in W_1^s(\Omega)$, there is a sequence $u_n \to u$ in $W_1^s(\Omega)$, such that $E(u) \ge \limsup_{n\to\infty} E_n(u_n)$.

Then, based on the link between wavelet frame transform and differential operators given by Proposition 4.2, the main result of [24] is given as follows:

Theorem 4.1. *Given the variational problem* (4.3)*, there exists a set of coefficients* $\boldsymbol{\lambda}_n$*, such that the functional* E_n *of the problem* (4.2) Γ*-converges to the functional* E *of the problem* (4.3) *in* $W_1^s(\Omega)$*. Let* $u_n^\star$ *be an* ε*-optimal solution to the problem* (4.2)*, i.e.,* $E_n(u_n^\star) \leq \inf_u E_n(u) + \epsilon$ *(*$\epsilon > 0$*). We have that*

$$\limsup_{n\to\infty} E_n(u_n^\star) \leq \inf_u E(u) + \epsilon,$$

and any cluster point of $\{u_n^\star\}_n$ *is an* ε*-optimal solution to the problem* (4.3)*.*

Theorem 4.1 goes beyond the theoretical justifications of the linkage of (4.2) and (4.3). Since the differential operator-based analysis model (4.3) has strong geometric interpretations, this connection brings geometric interpretations to the wavelet frame-based approach (4.2) as well. This also leads to even wider applications of the wavelet frame-based approach, e.g., image segmentation [27, 48, 99] and 3D surface reconstruction [50]. Conversely, the theorem also grants a new perspective of sparse approximation to the PDE-based approach supplementing its current function space perspective. On the other hand, not only the wavelet frame-based analysis model can be viewed as a discrete approximation of the differential operator-based analysis model, but such discretization can also be superior to standard finite difference discretization commonly used in PDE-based methods. Taking the Haar wavelet frame-based analysis model as an example, its regularization term has the property of 45-degree rotation invariance. In contrast, the standard finite difference discretization for TV regularization does not have such an invariance. This enables Haar wavelet frame-based analysis model to generate better reconstructed images than the TV model with the standard discretization.

4.3. Connections between wavelet shrinkage algorithms and nonlinear evolutional PDEs

In [49], general connections between wavelet frame shrinkage algorithms and nonlinear evolution PDEs (e.g., PM equation, shock-filters, anisotropic diffusions) were established. The links between the two approaches provide new and inspiring interpretations of themselves that enable us to derive new PDE models and (better) wavelet frame shrinkage algorithms for image restoration. Here, we will recall some of the main results from [49].

Let $\boldsymbol{d} := \boldsymbol{W}\boldsymbol{u}$ be the wavelet frame transform of $\boldsymbol{u}$, $\tilde{\boldsymbol{W}}^\top \boldsymbol{d}$ be the inverse wavelet frame transform defined by (2.1) and (2.2) with the corresponding filters satisfying the extension principles [128,129]. Then, we have $\tilde{\boldsymbol{W}}^\top \boldsymbol{W} = \boldsymbol{I}$. For simplicity, we only consider 1-level wavelet frame transform. Given wavelet frame coefficients $\boldsymbol{d} = \{d_{\ell,\boldsymbol{n}} : \boldsymbol{n} \in \mathbb{Z}^2, 0 \leq \ell \leq L\}$ and a threshold $\boldsymbol{\lambda}(\boldsymbol{d}) = \{\lambda_{\ell,\boldsymbol{n}}(\boldsymbol{d}) : \boldsymbol{n} \in \mathbb{Z}^2, 0 \leq \ell \leq L\}$, the shrinkage operator $\boldsymbol{S}_{\boldsymbol{\lambda}}(\boldsymbol{d})$ is defined as follows:

$$\boldsymbol{S}_{\boldsymbol{\lambda}}(\boldsymbol{d}) = \big\{S_{\lambda_{\ell,\boldsymbol{n}}(\boldsymbol{d})}(d_{\ell,\boldsymbol{n}}) = d_{\ell,\boldsymbol{n}}\big(1 - \lambda_{\ell,\boldsymbol{n}}(\boldsymbol{d})\big) : \boldsymbol{n} \in \mathbb{Z}^2, 0 \leq \ell \leq L\big\}. \tag{4.4}$$

Two well-known examples of the shrinkage operator (4.4) are the isotropic and anisotropic soft-thresholding operators [24, 49, 54].

Given the shrinkage operator $\boldsymbol{S}_{\boldsymbol{\lambda}}$, a generic wavelet frame shrinkage algorithm takes the form

$$\boldsymbol{u}^k = \tilde{\boldsymbol{W}}^\top \boldsymbol{S}_{\boldsymbol{\lambda}^{k-1}}(\boldsymbol{W}\boldsymbol{u}^{k-1}), \quad k = 1, 2, \dots \tag{4.5}$$

Note that, for simplicity, we have dropped the term of data fidelity. More general versions of the algorithm can be found in [49]. Now, consider the following nonlinear evolution PDE:

$$u_t = \sum_{\ell=1}^{L} \frac{\partial^{\boldsymbol{\alpha}_\ell}}{\partial x^{\boldsymbol{\alpha}_\ell}} \Phi_\ell(\boldsymbol{D}u, u), \quad \boldsymbol{D} = \left(\frac{\partial^{\boldsymbol{\beta}_1}}{\partial x^{\boldsymbol{\beta}_1}}, \dots, \frac{\partial^{\boldsymbol{\beta}_L}}{\partial x^{\boldsymbol{\beta}_L}} \right). \tag{4.6}$$

The PDE (4.6) is defined on $\mathbb{R}^2$, and $|\boldsymbol{\alpha}_\ell|, |\boldsymbol{\beta}_\ell| \geq 0$, $1 \leq \ell \leq L$. Thus, it covers most nonlinear parabolic and hyperbolic equations that we use for image reconstruction.

One key results of [49] can be summarized as follows: Given a PDE that takes the form (4.6), then we can construction wavelet frame transforms $\boldsymbol{W}$ and $\tilde{\boldsymbol{W}}$, and a shrinkage operator $\boldsymbol{S}_{\boldsymbol{\lambda}}$ such that the wavelet frame shrinkage algorithm (4.5) is an approximation of the PDE (4.6). When the PDE (4.6) is a well-posed anisotropic diffusion, the discrete solution obtained from (4.5) converges to the solution of the PDE. This result is a consequence of Proposition 4.1.

Let us consider a simple example. Consider the PDE

$$u_t = \frac{\partial \Phi_1}{\partial x_1}\left(\frac{\partial u}{\partial x_1}, \frac{\partial u}{\partial x_2}, u \right) + \frac{\partial \Phi_2}{\partial x_2}\left(\frac{\partial u}{\partial x_1}, \frac{\partial u}{\partial x_2}, u \right).$$

Let $\boldsymbol{W}_\ell$, $\ell = 1, 2$, be the Haar wavelet frame transform corresponding to the first two high-frequency bands. By Proposition 4.1, we have the following discretization of the above PDE:

$$\begin{aligned} \boldsymbol{u}^k = \boldsymbol{u}^{k-1} &- \tau\tilde{\gamma}_1 \boldsymbol{W}_1^\top \Phi_1(\gamma_1 \boldsymbol{W}_1 \boldsymbol{u}^{k-1}, \gamma_2 \boldsymbol{W}_2 \boldsymbol{u}^{k-1}, \boldsymbol{u}^{k-1}) \\ &- \tau\tilde{\gamma}_2 \boldsymbol{W}_2^\top \Phi_2(\gamma_1 \boldsymbol{W}_1 \boldsymbol{u}^{k-1}, \gamma_2 \boldsymbol{W}_2 \boldsymbol{u}^{k-1}, \boldsymbol{u}^{k-1}), \end{aligned}$$

with parameters γ_ℓ and $\tilde{\gamma}_\ell$ being properly chosen such that $\gamma_\ell \boldsymbol{W}_\ell \approx \frac{\partial}{\partial x_\ell}$ and $\tilde{\gamma}_\ell \boldsymbol{W}_\ell^\top \approx \frac{\partial}{\partial x_\ell}$. On the other hand, the iterative algorithm (4.5) can be rewritten as

$$\begin{aligned} \boldsymbol{u}^k = \boldsymbol{u}^{k-1} &- \boldsymbol{W}_1^\top \big[\boldsymbol{W}_1 \boldsymbol{u}^{k-1} - \mathcal{S}_1(\boldsymbol{W}_1 \boldsymbol{u}^{k-1}, \boldsymbol{W}_2 \boldsymbol{u}^{k-1}, \boldsymbol{u}^{k-1}) \big] \\ &- \boldsymbol{W}_2^\top \big[\boldsymbol{W}_2 \boldsymbol{u}^{k-1} - \mathcal{S}_2(\boldsymbol{W}_1 \boldsymbol{u}^{k-1}, \boldsymbol{W}_2 \boldsymbol{u}^{k-1}, \boldsymbol{u}^{k-1}) \big]. \end{aligned}$$

Comparing the above two iterative formulas, we can see that if we define the operator $\mathcal{S} = \{S^\ell : \ell = 1, 2\}$ as

$$S^\ell(\xi_1, \xi_2, \zeta) := \xi_\ell - \tau\tilde{\gamma}_\ell \Phi_\ell(\xi_1, \xi_2, \zeta) = \xi_\ell \big(1 - \tau\tilde{\gamma}_\ell \Phi_\ell(\xi_1, \xi_2, \zeta)/\xi_\ell \big), \quad \xi_\ell, \zeta \in \mathbb{R},$$

(whenever $\Phi_\ell(\xi_1, \xi_2, \zeta)/\xi_\ell$ is well defined), then there is an exact correspondence between the two iterative formulas. Note that the threshold level in the original definition (4.4) is given by $\tau\tilde{\gamma}_\ell \Phi_\ell(\xi_1, \xi_2, \zeta)/\xi_\ell$. In particular, when

$$\Phi_\ell\left(\frac{\partial u}{\partial x_1}, \frac{\partial u}{\partial x_2}, u \right) = g_\ell\big(|\nabla u|^2, u \big) \frac{\partial u}{\partial x_\ell},$$

we have

$$S^{\ell}(\xi_1, \xi_2, \zeta) = \xi_{\ell}\big(1 - \tau\tilde{\gamma}\ell g_{\ell}(\xi_1^2 + \xi_2^2, \zeta)\big).$$

It is interesting to observe that the threshold level given by $\tau\tilde{\gamma}\ell g_{\ell}(\xi_1^2 + \xi_2^2, \zeta)$ is proportional to the diffusivity g_{ℓ}.

Other than showing that the wavelet frame shrinkage algorithms can be viewed as a discrete approximation of PDEs, **[49]** also presented examples of new PDEs that can be derived from wavelet frame shrinkage algorithms. Conversely, new wavelet shrinkage algorithms that better exploit local image geometry can also be derived. Here, we recall one such example.

Consider the accelerated wavelet frame shrinkage algorithm **[93, 116, 117]**

$$\begin{aligned} \boldsymbol{u}^k = (I - \mu \boldsymbol{A}^{\top}\boldsymbol{A})\boldsymbol{W}^{\top}\boldsymbol{S}_{\boldsymbol{\alpha}^{k-1}}\big((1+\gamma^{k-1})\boldsymbol{W}\boldsymbol{u}^{k-1} - \gamma^{k-1}\boldsymbol{W}\boldsymbol{u}^{k-2}\big) \\ + \mu \boldsymbol{A}^{\top}\boldsymbol{f}, \quad k = 1, 2, \ldots \end{aligned} \tag{4.7}$$

When we properly choose the wavelet frame transform $\boldsymbol{W}$ and the parameters μ and γ^k, the iterative algorithm (4.7) leads to the following PDE:

$$\begin{aligned} u_{tt} + Cu_t = \sum_{\ell=1}^{L}(-1)^{1+|\boldsymbol{\beta}_{\ell}|}\frac{\partial^{\boldsymbol{\beta}_{\ell}}}{\partial \boldsymbol{x}^{\boldsymbol{\beta}_{\ell}}}\left[g_{\ell}\left(u, \frac{\partial^{\boldsymbol{\beta}_1}u}{\partial \boldsymbol{x}^{\boldsymbol{\beta}_1}}, \ldots, \frac{\partial^{\boldsymbol{\beta}_L}u}{\partial \boldsymbol{x}^{\boldsymbol{\beta}_L}}\right)\frac{\partial^{\boldsymbol{\beta}_{\ell}}}{\partial \boldsymbol{x}^{\boldsymbol{\beta}_{\ell}}}u\right] \\ - \kappa A^{\top}(Au - f). \end{aligned} \tag{4.8}$$

What makes equation (4.8) interesting is the presence of both u_t and u_{tt}. The term u_t makes the PDE parabolic-like so that the first term on the right-hand side regularizes the solution u; the term u_{tt} makes the PDE hyperbolic-like so that the evolution of u is accelerated. The idea of using a hyperbolic equation to speed up convergence was proposed in **[111]** for sparse signal reconstruction from noisy, blurry observations. Furthermore, related findings was also given by **[149, 150]**. It also inspired more recent studies in machine learning that established connections between numerical ODEs and CNNs **[107]**.

5. GOING BEYOND IMAGE RECONSTRUCTION

Differential equations, especially partial differential equations (PDEs), play a prominent role in physics, chemistry, biology, economics, engineering, etc., to describe the governing laws underlying virtually every physical, technical, or biological process. The application of differential equations in image reconstruction and computer vision is a relatively new field that started around 1990. In Section 4, we have unified the prevailing models in image reconstruction, from which we can see that most effective image reconstruction algorithms are various discrete approximations of differential equations. In this section, we shall bridge the design of certain types of CNNs with numerical differential equations. More specifically, the bridge between numerical ODEs/SDEs and CNNs was established by **[107]** and the bridge between numerical PDEs and CNNs was established by **[101, 102]**. In this line of work, we regard CNNs as a discrete dynamical system, and the flow of features from the very first layer to the last layer of the CNNs is the underlying dynamical process. We argue that different

numerical schemes of differential equations lead to different architectures of CNNs, which inherit certain properties from the differential equations. By connecting CNNs with numerical differential equations, we can bring in tools from applied mathematics and physics to shed light on the interpretability of CNNs; and we can also bring in tools from deep learning to further advance not only image reconstruction but also a much broader field of scientific computing.

5.1. ODE-Nets: exploring structural similarity between numerical ODEs and CNNs

One of the central tasks in deep learning is designing effective deep architectures with strong generalization potential and are easy to train. The first ultra-deep CNN is the ResNet [77] where skip connections were introduced to keep feature maps in different layers on the same scale and to avoid gradient vanishing. Structures other than the skip connections of the ResNet were also introduced to prevent gradient vanishing, such as the dense connections [79] and fractal path [90].

Observe that each residual block of ResNet can be written as

$$\boldsymbol{u}^{k+1} = \boldsymbol{u}^k + \Delta t F(\boldsymbol{u}^k, t_k), \tag{5.1}$$

where k is the index for the residual block F. As first suggested by [74, 162], each residual block of ResNet is one step of the forward-Euler discretization of the ODE $\dot{u} = F(u, t)$. In [107], we further showed that many state-of-the-art deep network architectures, such as PolyNet [174], FractalNet [90], and RevNet [70], which can be considered as different discretizations of ODEs. From the perspective of [107], the success of these networks is mainly due to their ability to efficiently approximate dynamical systems.

Taking PolyNet as an example, a PolyInception module was introduced in each residual block. The PolyInception model includes polynomial compositions that can be described as

$$(I + F + F^2) \cdot x = x + F(x) + F\big(F(x)\big).$$

We observed in [107] that PolyInception model can be interpreted as an approximation to one step of the backward-Euler (implicit) scheme, $\boldsymbol{u}^{k+1} = (I - \Delta t F)^{-1}\boldsymbol{u}^k$. Indeed, we can formally rewrite $(I - \Delta t F)^{-1}$ as

$$I + \Delta t F + (\Delta t F)^2 + \cdots + (\Delta t F)^k + \cdots .$$

Therefore, the architecture of PolyNet can be viewed as an approximation to the backward-Euler scheme solving the ODE $\boldsymbol{u}_t = F(\boldsymbol{u}, t)$. Note that the implicit scheme allows a larger step size [6], which in turn allows fewer residual blocks.

Furthermore, for residual-type networks with random perturbations, such as ResNet with shake-shake regularization [67] and stochastic depth [80], it was shown by [107] that these networks can be viewed as weak approximations [118] to certain SDEs, which links the

training of such networks with mean-field stochastic optimal control

$$\min \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{P}}\left(\mathbb{E}\left(\ell(X_T,\boldsymbol{y})+\int_0^T r(s,X_s,\theta_s)\right)ds\right)$$
$$\text{s.t. } dX_t = f(X_t,\theta_t)dt + g(X_t,\theta_t)dB_t, \quad X_0 = \boldsymbol{x},$$

where $\ell(\cdot,\cdot)$ is a certain loss function measuring the distance between the two input arguments, $r(\cdot,\cdot,\cdot)$ is a running cost that regularizes the dynamics and $\mathcal{P}$ is the distribution of the data. Note that the SDE and stochastic optimal control perspective on ResNet with dropout **[146]** was later proposed by **[151]**.

In **[107]**, we argued that we could exploit numerical ODEs to design new residual-type CNNs with state-of-the-art classification accuracy. Here, we shall call these deep residual-type CNNs inspired by numerical schemes of ODEs as ODE-Nets. As an example, we proposed to use a linear two-step scheme for ODEs to design a new ODE-Net, called LM-ResNet, as follows:

$$\boldsymbol{u}^{k+1} = (1-\alpha_k)\boldsymbol{u}^k + \alpha_k\boldsymbol{u}^{k-1} + F(\boldsymbol{u}^k, t_k), \quad \alpha_k \in \mathbb{R}. \tag{5.2}$$

The difference between the LM-ResNet (5.2) and the original ResNet (5.1) is revealed by the modified equation analysis **[160]**. Modified equations are commonly used in numerical analysis to describe numerical behaviors of numerical schemes. The modified equations of the ResNet and the LM-ResNet are as follows:

$$\begin{cases} \dot{u}^k + \frac{\Delta t}{2}\ddot{u}^k = F(u^k, t_k), & \text{ResNet;} \\ (1+\alpha_k)\dot{u}^k + (1-\alpha_k)\frac{\Delta t}{2}\ddot{u}^k = F(u^k, t_k), & \text{LM-ResNet.} \end{cases} \tag{5.3}$$

Here, $u^k = u(t_k)$ and similarly for $\dot{u}^k$ and $\ddot{u}^k$. Comparing the two modified equations in (5.3), we can see that when $\alpha_k \le 0$, the second-order term $\ddot{u}$ of the modified equation of LM-ResNet is bigger than that of the original ResNet. Note that the term $\ddot{u}$ represents acceleration which leads to acceleration of the convergence of $\boldsymbol{u}^k$ when $F = -\nabla G$, which was observed earlier for $F(u)$ taking a particular form in (4.8). This was our original motivation to select (5.2) among numerous other numerical ODE schemes, since we believed the depth of the corresponding ODE-Net could be reduced compared to the original ResNet because of the acceleration mechanisms induced by the term $\ddot{u}^k$. It turned out that it was indeed the case, and LM-ResNet managed to reduce the depth of the original ResNet (the versions with stochastic perturbations as well) by a factor of 2–10 without hurting classification accuracy. This was empirically validated on image classification benchmarks CIFAR10/100 and ImageNet.

The bridge between numerical schemes and architectures of neural networks can not only inspire various designs of ODE-Nets **[34, 94, 108, 177]**, concepts from the numerical analysis can also be introduced to enforce the ODE-Nets to satisfy certain desired properties. For example, **[41]** utilized a symplectic scheme to enforce the learned network to preserve the physic structure, and **[173]** boosted the stability and adversarial robustness of ResNet through stability analysis on the underlying dynamical system. The bridge also inspired the work on the neural ODE **[37, 97]** in which ODEs and SDEs were used as machine learning models, and they have achieved huge success in generative modeling. The validity of

using dynamical systems as machine learning models was provided by [96] where the universal approximation property of these models was established. This line of research also inspired more applications of ODE-Nets in time series prediction [87] and physical system identification [73, 127, 168]. By regarding training ResNet as an optimal control problem, [95] discovered that the BP-based optimization algorithm could be viewed as an iterative solver for the maximal principle of the optimal control problem. Based on this observation, [98, 171] designed new accelerated training algorithms for ODE-Nets inspired by the theory of optimal control. Although the structural similarity between numerical ODEs and CNNs is mostly formal, theoretical analysis regarding the depth limit of ODE-Nets has become a vibrant and fast-moving field of research [43, 106, 121, 125, 153].

5.2. PDE-Nets: exploring structural similarity between numerical PDEs and CNNs

The original motivation of the work PDE-Nets [101, 102] was to design transparent CNNs to uncover hidden PDE models from observed dynamical data with minor prior knowledge on the mechanisms of the dynamics and to perform accurate predictions at the same time. Learning PDEs from observation or measurement data is a typical task in inverse problems in which machine learning methods have recently attracted tremendous attention [5]. However, existing CNNs designed for computer vision tasks primarily emphasize prediction accuracy. They are generally considered black-boxes and cannot reveal the hidden PDE model that drives the dynamical data. Therefore, we need to carefully design the CNN by exploring the structure similarity between numerical PDEs and CNNs.

Assume that the PDE to be uncovered takes the following generic form:

$$u_t = F(u, \boldsymbol{D}u), \quad x \in \Omega \subset \mathbb{R}^2,\ t \in [0, T],$$

where $\boldsymbol{D}$ was defined in (4.6). In a nutshell, PDE-Nets are designed as feedforward networks by discretizing the above PDE using forward-Euler (or any other temporal discretization) in time and finite-difference in space. The forward-Euler approximation of the temporal derivative makes PDE-Nets residual-type neural networks. As has been extensively discussed in Section 4, the finite-difference approximation to the differential operator $\boldsymbol{D}$ can realized by convolutions with properly chosen convolution kernels (i.e., filters). In fact, not only finite-difference approximations can be realized by convolutions, any discretization of $\boldsymbol{D}$ based on an approximation of u using translation-invariant basis functions can also be realized by convolutions [39]. The nonlinear response function F is approximated by a symbolic neural network denoted as SymNet (or a regular DNN as in [102] that is more expressive but less interpretable). Let $\boldsymbol{u}^{k+1}$ be the predicted value at time $t_k + \Delta t$ based on $\boldsymbol{u}^k$. Then, the PDE-Nets take the following dynamical form:

$$\boldsymbol{u}^{k+1} = \boldsymbol{u}^k + \Delta t \cdot \mathrm{SymNet}_m^n(Q_{00}\boldsymbol{u}^k, Q_{01}\boldsymbol{u}^k, Q_{10}\boldsymbol{u}^k, \dots), \quad k = 0, 1, \dots, K-1. \tag{5.4}$$

Here, the operators $\{Q_{ij}\}$ denote convolution operators with the underlying filters denoted by $\boldsymbol{q}_{ij}$, i.e., $Q_{ij}u = \boldsymbol{q}_{ij} \circledast \boldsymbol{u}$. The operators Q_{10}, Q_{01}, Q_{11}, etc., approximate differential operators, i.e., $Q_{ij}u \approx \frac{\partial^{i+j}\boldsymbol{u}}{\partial^i x \partial^j y}$. In particular, Q_{00} is a certain averaging operator. The symbolic

neural network SymNet_m^n is introduced to approximate the multivariate nonlinear response function F. The design of SymNet_m^n is motivated by [135]. It can accurately estimate function F that is formed or can be well approximated by multivariate polynomials. Details on SymNet_m^n and its properties can be found in [101]. All the parameters of the SymNet and the filters $\{\boldsymbol{q}_{ij}\}$ are jointly learned from data.

A key difference from existing works (e.g., [14,21,100,133,137,138,167]) on discovering PDE models from observation data prior to [101,102] is that the filters corresponding to the specific finite-difference approximations to $\boldsymbol{D}$ are learned jointly with the estimation of the nonlinear response function F. The benefits of doing such joint learning in both system identification and prediction were empirically demonstrated in [101]. More importantly, in order to grant desired interpretability to the PDE-Nets, proper constraints are enforced on the filters. These constraints are motivated from Proposition 4.1 which we now elaborate.

In [101,102], the moment matrix associated to a given filter $\boldsymbol{q}$ was introduced to easily enforce constraints on the filter during training. Recall that the moment matrix $M(\boldsymbol{q})$ of an $N \times N$ filter $\boldsymbol{q}$ is defined by

$$M(\boldsymbol{q}) = (m_{i,j})_{N\times N}, \tag{5.5}$$

where

$$m_{i,j} = \frac{1}{i!j!} \sum_{k_1,k_2=-\frac{N-1}{2}}^{\frac{N-1}{2}} k_1^i k_2^j \boldsymbol{q}[k_1,k_2], \quad i,j = 0,1,\ldots,N-1. \tag{5.6}$$

Then, by examining (5.6), (4.1), and Proposition 4.1, it is not hard to see that, with a properly chosen N, filter $\boldsymbol{q}$ can be designed to approximate any differential operator with prescribed order of accuracy by imposing constraints on $M(q)$.

For example, if we want to approximate $\frac{\partial u}{\partial x}$ (up to a constant) by convolution $\boldsymbol{q} \circledast \boldsymbol{u}$ where $\boldsymbol{q}$ is a 3×3 filter and $\boldsymbol{u}$ is the evaluation of u on a regular grid, we can consider the following constrains on $M(q)$:

$$\begin{pmatrix} 0 & 0 & \star \\ 1 & \star & \star \\ \star & \star & \star \end{pmatrix} \quad \text{or} \quad \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & \star \\ 0 & \star & \star \end{pmatrix}. \tag{5.7}$$

Here, $\star$ means no constraint on the corresponding entry which allows one degree of freedom for learning. The constraints described by the moment matrix on the left of (5.7) guarantee that the approximation accuracy is at least of first order, and that on the right guarantees an approximation of at least second order. In particular, when all entries of $M(q)$ are constrained, the corresponding filter is uniquely determined. In the PDE-Nets, all filters are learned subjected to partial constraints on their associated moment matrices. Similar ideas on learning constrained filters to approximate differential operators were later used in [9] to design data-driven solvers for PDEs, and in [31] to design data-driven discretizations for total variations. A more extended discussion on the connections between numerical PDEs and neural networks was given in [2].

Exploiting the links between PDEs and CNNs has become a popular line of research that has led to many new designs of CNN models for machine learning and computer vision

tasks [4, 76, 113, 134, 152, 175]. It can also be used to improve the efficiency of CNNs [57]. On the other hand, this line of research also drives the development of data-driven modeling in scientific computing including efficient solvers for PDEs [9, 11, 39, 40, 85, 105, 158], model reduction of complex systems [109, 112, 126, 165, 166, 169], system identification from observation or simulation data [8, 15, 40, 75, 81, 83, 103, 124, 132], control of physical systems [78, 157], inverse problems [5, 61, 62, 86], and applications in seismology [88]. In addition, building PDE models on unstructured data for machine learning and scientific computing tasks is now an emerging branch of research [1, 29, 56, 82, 139].

FUNDING

This work was partially supported by the National Natural Science Foundation of China (grant No. 12090022), Beijing Natural Science Foundation (grant No. 180001) and Beijing Academy of Artificial Intelligence (BAAI).

REFERENCES

[1] F. Alet, A. K. Jeewajee, M. B. Villalonga, A. Rodriguez, T. Lozano-Perez, and L. Kaelbling, Graph element networks: adaptive, structured computation and memory. In *International Conference on Machine Learning*, pp. 212–222, Proceedings of Machine Learning Research, 2019.

[2] T. Alt, K. Schrader, M. Augustin, P. Peter, and J. Weickert, Connections between numerical algorithms for PDEs and neural networks. 2021, arXiv:2107.14742.

[3] L. Alvarez, F. Guichard, P. Lions, and J. Morel, Axioms and fundamental equations of image processing. *Arch. Ration. Mech. Anal.* **123** (1993), no. 3, 199–257.

[4] S. Arridge and A. Hauptmann, Networks for nonlinear diffusion problems in imaging. *J. Math. Imaging Vision* **62** (2020), no. 3, 471–487.

[5] S. Arridge, P. Maass, O. Öktem, and C.-B. Schönlieb, Solving inverse problems using data-driven models. *Acta Numer.* **28** (2019), 1–174.

[6] U. M. Ascher and L. R. Petzold, *Computer methods for ordinary differential equations and differential-algebraic equations*. SIAM: Society for Industrial and Applied Mathematics, 1997.

[7] G. Aubert and P. Kornprobst, *Mathematical problems in image processing: partial differential equations and the calculus of variations*. Appl. Math. Sci. 147, Springer, New York, 2006.

[8] I. Ayed, E. de Bézenac, A. Pajot, J. Brajard, and P. Gallinari, Learning dynamical systems from partial observations. 2019, arXiv:1902.11136.

[9] Y. Bar-Sinai, S. Hoyer, J. Hickey, and M. P. Brenner, Learning data-driven discretizations for partial differential equations. *Proc. Natl. Acad. Sci. USA* **116** (2019), no. 31, 15344–15349.

[10] A. Beck and M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sci.* **2** (2009), no. 1, 183–202.

[11] C. Beck, S. Becker, P. Cheridito, A. Jentzen, and A. Neufeld, Deep splitting method for parabolic PDEs. 2019, arXiv:1907.03452

[12] M. Bertalmio, A. L. Bertozzi, and G. Sapiro, Navier–Stokes, fluid dynamics, and image and video inpainting. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. I–I, 1, IEEE, 2001.

[13] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, Image inpainting. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 417–424, ACM Press/Addison-Wesley Publishing Co., 2000.

[14] J. Bongard and H. Lipson, Automated reverse engineering of nonlinear dynamical systems. *Proc. Natl. Acad. Sci. USA* **104** (2007), no. 24, 9943–9948.

[15] G.-J. Both, S. Choudhury, P. Sens, and R. Kusters, Deepmod: deep learning for model discovery in noisy data. *J. Comput. Phys.* **428** (2021), 109985.

[16] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Signal Process.* **3** (2011), no. 1, 1–122.

[17] A. Braides, *Gamma-convergence for beginners*. Oxford Lecture Series in Mathematics and Its Applications, Vol. 22, Oxford University Press, 2002.

[18] K. Bredies, K. Kunisch, and T. Pock, Total generalized variation. *SIAM J. Imaging Sci.* **3** (2010), 492.

[19] R. W. Brown, E. M. Haacke, Y.-C. N. Cheng, M. R. Thompson, and R. Venkatesan, *Magnetic resonance imaging: physical principles and sequence design*. John Wiley & Sons, 2014.

[20] R. E. Bruck Jr, On the weak convergence of an ergodic iteration for the solution of variational inequalities for monotone operators in hilbert space. *J. Math. Anal. Appl.* **61** (1977), no. 1, 159–164.

[21] S. L. Brunton, J. L. Proctor, and J. N. Kutz, Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proc. Natl. Acad. Sci. USA* (2016), 201517384.

[22] T. M. Buzug, *Computed tomography: from photon statistics to modern cone-beam CT*. Springer, Berlin, 2008.

[23] J. Cai, R. Chan, and Z. Shen, A framelet-based image inpainting algorithm. *Appl. Comput. Harmon. Anal.* **24** (2008), no. 2, 131–149.

[24] J. Cai, B. Dong, S. Osher, and Z. Shen, Image restorations: total variation, wavelet frames and beyond. *J. Amer. Math. Soc.* **25** (2012), no. 4, 1033–1089.

[25] J. Cai, S. Osher, and Z. Shen, Split Bregman methods and frame based image restoration. *Multiscale Model. Simul.* **8** (2009), no. 2, 337–369.

[26] J.-F. Cai, B. Dong, and Z. Shen, Image restoration: a wavelet frame based model for piecewise smooth functions and beyond. *Appl. Comput. Harmon. Anal.* **41** (2016), no. 1, 94–138.

[27] X. Cai, R. Chan, S. Morigi, and F. Sgallari, Vessel segmentation in medical imaging using a tight-frame–based algorithm. *SIAM J. Imaging Sci.* **6** (2013), no. 1, 464–486.

[28] F. Catté, P.-L. Lions, J.-M. Morel, and T. Coll, Image selective smoothing and edge detection by nonlinear diffusion. *SIAM J. Numer. Anal.* **29** (1992), no. 1, 182–193.

[29] B. P. Chamberlain, J. Rowbottom, M. Gorinova, S. Webb, E. Rossi, and M. M. Bronstein, GRAND: graph neural diffusion. 2021, arXiv:2106.10934.

[30] A. Chambolle and T. Pock, A first-order primal-dual algorithm for convex problems with applications to imaging. *J. Math. Imaging Vision* **40** (2011), no. 1, 120–145.

[31] A. Chambolle and T. Pock, Learning consistent discretizations of the total variation. *SIAM J. Imaging Sci.* **14** (2021), no. 2, 778–813.

[32] R. Chan, T. Chan, L. Shen, and Z. Shen, Wavelet algorithms for high-resolution image reconstruction. *SIAM J. Sci. Comput.* **24** (2003), no. 4, 1408–1432.

[33] T. F. Chan and J. Shen, *Image processing and analysis: variational, PDE, wavelet, and stochastic methods*. SIAM, 2005.

[34] B. Chang, L. Meng, E. Haber, L. Ruthotto, D. Begert, and E. Holtham, Reversible architectures for arbitrarily deep residual neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32, No. 1, 2018.

[35] G. H. Chen and R. Rockafellar, Convergence rates in forward–backward splitting. *SIAM J. Optim.* **7** (1997), no. 2, 421–444.

[36] L. Chen and C. Wu, A note on the expressive power of deep rectified linear unit networks in high-dimensional spaces. *Math. Methods Appl. Sci.* **42** (2019), no. 9, 3400–3404.

[37] R. T. Chen, Y. Rubanova, J. Bettencourt, and D. Duvenaud, Neural ordinary differential equations. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pp. 6572–6583, 2018.

[38] T. Chen, X. Chen, W. Chen, H. Heaton, J. Liu, Z. Wang, and W. Yin, Learning to optimize: a primer and a benchmark. 2021, arXiv:2103.12828.

[39] Y. Chen, B. Dong, and J. Xu, Meta-MgNet: Meta multigrid networks for solving parameterized partial differential equations. 2020, arXiv:2010.14088.

[40] Y. Chen, B. Hosseini, H. Owhadi, and A. M. Stuart, Solving and learning nonlinear PDEs with Gaussian processes. 2021, arXiv:2103.12959.

[41] Z. Chen, J. Zhang, M. Arjovsky, and L. Bottou, Symplectic recurrent neural networks. 2019, arXiv:1909.13334.

[42] J. K. Choi, B. Dong, and X. Zhang, An edge driven wavelet frame model for image restoration. *Appl. Comput. Harmon. Anal.* **48** (2020), no. 3, 993–1029.

[43] A. Cohen, R. Cont, A. Rossier, and R. Xu, Scaling properties of deep residual networks. 2021, arXiv:2105.12245.

[44] P. Combettes and V. Wajs, Signal recovery by proximal forward–backward splitting. *Multiscale Model. Simul.* **4** (2006), no. 4, 1168–1200.

[45] I. Daubechies, *Ten lectures on wavelets*. CBMS-NSF Lecture Notes, SIAM, 61, Society for Industrial and Applied Mathematics, 1992.

[46] I. Daubechies, B. Han, A. Ron, and Z. Shen, Framelets: MRA-based constructions of wavelet frames. *Appl. Comput. Harmon. Anal.* **14** (2003), no. 1, 1–46.

[47] I. Daubechies, G. Teschke, and L. Vese, Iteratively solving linear inverse problems under general convex constraints. *Inverse Probl. Imaging* **1** (2007), no. 1, 29.

[48] B. Dong, A. Chien, and Z. Shen, Frame based segmentation for medical images. *Commun. Math. Sci.* **9** (2010), no. 2, 551–559.

[49] B. Dong, Q. Jiang, and Z. Shen, Image restoration: wavelet frame shrinkage, nonlinear evolution PDEs, and beyond. *Multiscale Model. Simul.* **15** (2017), no. 1, 606–660.

[50] B. Dong and Z. Shen, Frame based surface reconstruction from unorganized points. *J. Comput. Phys.* **230** (2011), 8247–8255.

[51] B. Dong and Z. Shen, Image restoration: a data-driven perspective. In *Proceedings of the International Congress of Industrial and Applied Mathematics (ICIAM)*, pp. 65–108, Citeseer, 2015.

[52] B. Dong, Z. Shen, and P. Xie, Image restoration: a general wavelet frame based model and its asymptotic analysis. *SIAM J. Math. Anal.* **49** (2017), no. 1, 421–445.

[53] B. Dong, Z. Shen et al., *MRA-based wavelet frames and applications*. Summer Program on "The Mathematics of Image Processing". IAS Lect. Notes Ser. 19, Park City Mathematics Institute, 2010.

[54] D. L. Donoho, De-noising by soft-thresholding. *IEEE Trans. Inf. Theory* **41** (1995), no. 3, 613–627.

[55] M. Elad, J. Starck, P. Querre, and D. Donoho, Simultaneous cartoon and texture image inpainting using morphological component analysis (MCA). *Appl. Comput. Harmon. Anal.* **19** (2005), no. 3, 340–358.

[56] M. Eliasof, E. Haber, and E. Treister, PDE-GCN: novel architectures for graph neural networks motivated by partial differential equations. 2021, arXiv:2108.01938.

[57] J. Ephrath, M. Eliasof, L. Ruthotto, E. Haber, and E. Treister, Leanconvnets: low-cost yet effective convolutional neural networks. *IEEE J. Sel. Top. Signal Process.* **14** (2020), no. 4, 894–904.

[58] E. Esser, X. Zhang, and T. F. Chan, A general framework for a class of first order primal–dual algorithms for convex optimization in imaging science. *SIAM J. Imaging Sci.* **3** (2010), no. 4, 1015–1046.

[59] M. Fadili and J. Starck, Sparse representations and Bayesian image inpainting. *Proc. SPARS* **5** (2005).

[60] M. Fadili, J. Starck, and F. Murtagh, Inpainting and zooming using sparse representations. *Comput. J.* **52** (2009), no. 1, 64.

[61] Y. Fan and L. Ying, Solving inverse wave scattering with deep learning. 2019, arXiv:1911.13202.

[62] Y. Fan and L. Ying, Solving electrical impedance tomography with deep learning. *J. Comput. Phys.* **404** (2020), 109119.

[63] M. Figueiredo and R. Nowak, An EM algorithm for wavelet-based image restoration. *IEEE Trans. Image Process.* **12** (2003), no. 8, 906–916.

[64] M. Figueiredo and R. Nowak, A bound optimization approach to wavelet-based image deconvolution. In *IEEE International Conference on Image Processing, 2005. ICIP 2005*, pp. II–782, 2, IEEE, 2005.

[65] K. Fukushima, Neocognitron: a self organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biol. Cybernet.* **36** (1980), no. 4, 193–202.

[66] D. Gabay and B. Mercier, A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Comput. Math. Appl.* **2** (1976), no. 1, 17–40.

[67] X. Gastaldi, Shake-shake regularization. 2017, arXiv:1705.07485.

[68] R. Glowinski and A. Marroco, Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problèmes de dirichlet non linéaires. *Rev. Fr. Autom. Inform. Rech. Opér., Anal. Numér.* **9** (1975), no. R2, 41–76.

[69] T. Goldstein and S. Osher, The split Bregman method for l_1-regularized problems. *SIAM J. Imaging Sci.* **2** (2009), no. 2, 323–343.

[70] A. N. Gomez, M. Ren, R. Urtasun, and R. B. Grosse, The reversible residual network: backpropagation without storing activations. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pp. 2211–2221, 2017.

[71] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT Press, Cambridge, 2016.

[72] K. Gröchenig, *Foundations of time-frequency analysis*. Birkhäuser, 2001.

[73] V. L. Guen and N. Thome, Disentangling physical dynamics from unknown factors for unsupervised video prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11474–11484, 2020.

[74] E. Haber and L. Ruthotto, Stable architectures for deep neural networks. *Inverse Probl.* **34** (2017), no. 1, 014004.

[75] J. Han, C. Ma, Z. Ma, and E. Weinan, Uniformly accurate machine learning-based hydrodynamic models for kinetic equations. *Proc. Natl. Acad. Sci. USA* **116** (2019), no. 44, 21983–21991.

[76] J. He and J. Xu, Mgnet: a unified framework of multigrid and convolutional neural network. *Sci. China Math.* **62** (2019), no. 7, 1331–1354.

[77] K. He, X. Zhang, S. Ren, and J. Sun, Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

[78] P. Holl, N. Thuerey, and V. Koltun, Learning to control PDEs with differentiable physics. 2020, arXiv:2001.07457

[79] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4700–4708, 2017.

[80] G. Huang, Y. Sun, Z. Liu, D. Sedra, and K. Q. Weinberger, Deep networks with stochastic depth. In *European Conference on Computer Vision*, pp. 646–661, Springer, 2016.

[81] J. Huang, Z. Ma, Y. Zhou, and W.-A. Yong, Learning thermodynamically stable and galilean invariant partial differential equations for non-equilibrium flows. *J. Non-Equilib. Thermodyn.* (2021).

[82] V. Iakovlev, M. Heinonen, and H. Lähdesmäki, Learning continuous-time PDEs from sparse data with graph neural networks. 2020, arXiv:2006.08956.

[83] J. Jia and A. R. Benson, Neural jump stochastic differential equations. *Adv. Neural Inf. Process. Syst.* **32** (2019), 9847–9858.

[84] Q. Jiang, Correspondence between frame shrinkage and high order nonlinear diffusion. *Appl. Numer. Math.* (2011).

[85] Y. Khoo, J. Lu, and L. Ying, Solving parametric PDE problems with artificial neural networks. *European J. Appl. Math.* **32** (2021), no. 3, 421–435.

[86] Y. Khoo and L. Ying, Switchnet: a neural network model for forward and inverse scattering problems. *SIAM J. Sci. Comput.* **41** (2019), no. 5, A3182–A3201.

[87] P. Kidger, J. Morrill, J. Foster, and T. Lyons, Neural controlled differential equations for irregular time series. 2020, arXiv:2005.08926.

[88] Q. Kong, D. T. Trugman, Z. E. Ross, M. J. Bianco, B. J. Meade, and P. Gerstoft, Machine learning in seismology: turning data into insights. *Seismol. Res. Lett.* **90** (2019), no. 1, 3–14.

[89] A. Krizhevsky, I. Sutskever, and G. E. Hinton, ImageNet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **25** (2012), 1097–1105.

[90] G. Larsson, M. Maire, and G. Shakhnarovich, FractalNet: ultra-deep neural networks without residuals. 2016, arXiv:1605.07648.

[91] Y. Lecun, Y. Bengio, and G. Hinton, Deep learning. *Nature* **521** (2015), 436–444.

[92] Y. LeCun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, and L. Jackel, Handwritten digit recognition with a back-propagation network. *Adv. Neural Inf. Process. Syst.* **2** (1989).

[93] M. Li, Z. Fan, H. Ji, and Z. Shen, Wavelet frame based algorithm for 3D reconstruction in electron microscopy. *SIAM J. Sci. Comput.* **36** (2014), no. 1, B45–B69.

[94] M. Li, L. He, and Z. Lin, Implicit Euler skip connections: enhancing adversarial robustness via numerical stability. In *International Conference on Machine Learning*, pp. 5874–5883, Proceedings of Machine Learning Research, 2020.

[95] Q. Li, L. Chen, and C. Tai, Maximum principle based algorithms for deep learning. *J. Mach. Learn. Res.* **18** (2018), 1–29.

[96] Q. Li, T. Lin, and Z. Shen, Deep learning via dynamical systems: an approximation perspective. *J. Eur. Math. Soc. (JEMS)* (to appear).

[97] X. Li, T.-K. L. Wong, R. T. Chen, and D. Duvenaud, Scalable gradients for stochastic differential equations. In *International Conference on Artificial Intelligence and Statistics*, pp. 3870–3882, Proceedings of Machine Learning Research, 2020.

[98] G.-H. Liu, T. Chen, and E. A. Theodorou, Differential dynamic programming neural optimizer. 2020, arXiv:2002.08809.

[99] J. Liu, X. Zhang, B. Dong, Z. Shen, and L. Gu, A wavelet frame method with shape prior for ultrasound video segmentation. *SIAM J. Imaging Sci.* **9** (2016), no. 2, 495–519.

[100] R. Liu, Z. Lin, W. Zhang, and Z. Su, Learning PDEs for image restoration via optimal control. In *European Conference on Computer Vision*, pp. 115–128, Springer, 2010.

[101] Z. Long, Y. Lu, and B. Dong, PDE-Net 2.0: Learning PDEs from data with a numeric-symbolic hybrid deep network. *J. Comput. Phys.* (2019), 108925.

[102] Z. Long, Y. Lu, X. Ma, and B. Dong, PDE-Net: Learning PDEs from data. In *International Conference on Machine Learning*, pp. 3214–3222, 2018.

[103] F. Lu, M. Maggioni, and S. Tang, Learning interaction kernels in heterogeneous systems of agents from multiple trajectories. 2019, arXiv:1910.04832.

[104] J. Lu, Z. Shen, H. Yang, and S. Zhang, Deep network approximation for smooth functions. 2020, arXiv:2001.03040.

[105] L. Lu, X. Meng, Z. Mao, and G. E. Karniadakis, Deepxde: a deep learning library for solving differential equations. *SIAM Rev.* **63** (2021), no. 1, 208–228.

[106] Y. Lu, C. Ma, Y. Lu, J. Lu, and L. Ying, A mean field analysis of deep ResNet and beyond: Towards provably optimization via overparameterization from depth. In *International Conference on Machine Learning*, pp. 6426–6436, Proceedings of Machine Learning Research, 2020.

[107] Y. Lu, A. Zhong, Q. Li, and B. Dong, Beyond finite layer neural networks: bridging deep architectures and numerical differential equations. In *International Conference on Machine Learning*, pp. 3276–3285, 2018.

[108] Z. Luo, Z. Sun, W. Zhou, and S.-i. Kamata, Rethinking ResNets: improved stacking strategies with high order schemes. 2021, arXiv:2103.15244.

[109] M. Lutter, C. Ritter, and J. Peters, Deep Lagrangian networks: using physics as model prior for deep learning. 2019, arXiv:1907.04490.

[110] S. Mallat, *A wavelet tour of signal processing: the sparse way*. Academic Press, 2008.

[111] Y. Mao, S. Osher, and B. Dong, A nonlinear PDE-based method for sparse deconvolution. *Multiscale Model. Simul.* **8** (2010), no. 3.

[112] A. Mohan, D. Daniel, M. Chertkov, and D. Livescu, Compressed convolutional LSTM: an efficient deep learning framework to model high fidelity 3D turbulence. 2019, arXiv:1903.00033.

[113] V. Monga, Y. Li, and Y. C. Eldar, Algorithm unrolling: interpretable, efficient deep learning for signal and image processing. *IEEE Signal Process. Mag.* **38** (2021), no. 2, 18–44.

[114] H. Montanelli, H. Yang, and Q. Du, Deep ReLu networks overcome the curse of dimensionality for bandlimited functions. 2019, arXiv:1903.00735.

[115] D. Mumford and J. Shah, Optimal approximations by piecewise smooth functions and associated variational problems. *Comm. Pure Appl. Math.* **42** (1989), no. 5, 577–685.

[116] Y. Nesterov, A method of solving a convex programming problem with convergence rate $o(1/k^2)$. *Sov. Math., Dokl.* **27** (1983), no. 2, 372–376.

[117] Y. Nesterov, On an approach to the construction of optimal methods for minimizing smooth convex functions. *Èkon. Mat. Metody* **24** (1988), no. 3, 509–517.

[118] B. Oksendal, *Stochastic differential equations: an introduction with applications*. Springer, Berlin, 2013.

[119] S. Osher and R. Fedkiw, *Level set methods and dynamic implicit surfaces*. Appl. Math. Sci. 153, Springer, New York, 2003.

[120] S. Osher and L. Rudin, Feature-oriented image enhancement using shock filters. *SIAM J. Numer. Anal.* **27** (1990), no. 4, 919–940.

[121] K. Ott, P. Katiyar, P. Hennig, and M. Tiemann, ResNet after all: neural ODEs and their numerical solution. In *International Conference on Learning Representations*, 2020.

[122] G. B. Passty, Ergodic convergence to a zero of the sum of monotone operators in Hilbert space. *J. Math. Anal. Appl.* **72** (1979), 383–290.

[123] P. Perona and J. Malik, Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. Pattern Anal. Mach. Intell.* **12** (1990), no. 7, 629–639.

[124] T. Qin, K. Wu, and D. Xiu, Data driven governing equations approximation using deep neural networks. *J. Comput. Phys.* **395** (2019), 620–635.

[125] A. F. Queiruga, N. B. Erichson, D. Taylor, and M. W. Mahoney, Continuous-in-depth neural networks. 2020, arXiv:2008.02389.

[126] C. Rackauckas, Y. Ma, J. Martensen, C. Warner, K. Zubov, R. Supekar, D. Skinner, A. Ramadhan, and A. Edelman, Universal differential equations for scientific machine learning. 2020, arXiv:2001.04385.

[127] M. Raissi, P. Perdikaris, and G. E. Karniadakis, Multistep neural networks for data-driven discovery of nonlinear dynamical systems. 2018, arXiv:1801.01236.

[128] A. Ron and Z. Shen, Affine systems in $L_2(\mathbb{R}^d)$ II: dual systems. *J. Fourier Anal. Appl.* **3** (1997), no. 5, 617–638.

[129] A. Ron and Z. Shen, Affine systems in $L_2(\mathbb{R}^d)$: the analysis of the analysis operator. *J. Funct. Anal.* **148** (1997), no. 2, 408–447.

[130] O. Ronneberger, P. Fischer, and T. Brox, U-Net: convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pp. 234–241, 2015.

[131] L. Rudin, S. Osher, and E. Fatemi, Nonlinear total variation based noise removal algorithms. *Phys. D* **60** (1992), 259–268.

[132] S. Rudy, A. Alla, S. L. Brunton, and J. N. Kutz, Data-driven identification of parametric partial differential equations. *SIAM J. Appl. Dyn. Syst.* **18** (2019), no. 2, 643–660.

[133] S. H. Rudy, S. L. Brunton, J. L. Proctor, and J. N. Kutz, Data-driven discovery of partial differential equations. *Sci. Adv.* **3** (2017), no. 4, e1602614.

[134] L. Ruthotto and E. Haber, Deep neural networks motivated by partial differential equations. *J. Math. Imaging Vision* **62** (2020), no. 3, 352–364.

[135] S. Sahoo, C. Lampert, and G. Martius, Learning equations for extrapolation and control. In *International Conference on Machine Learning*, pp. 4442–4450, Proceedings of Machine Learning Research, 2018.

[136] G. Sapiro, *Geometric partial differential equations and image analysis*. Cambridge University Press, 2001.

[137] H. Schaeffer, Learning partial differential equations via data discovery and sparse optimization. *Proc. R. Soc. A, Math. Phys. Eng. Sci.* **473** (2017), no. 2197, 20160446.

[138] M. Schmidt and H. Lipson, Distilling free-form natural laws from experimental data. *Science* **324** (2009), no. 5923, 81–85.

[139] S. Seo, C. Meng, and Y. Liu, Physics-aware difference graph networks for sparsely-observed dynamics. In *International Conference on Learning Representations*, 2019.

[140] Z. Shen, Wavelet frames and image restorations. In *Proceedings of the International Congress of Mathematicians*, Vol. 4, pp. 2834–2863, World Scientific, 2010.

[141] Z. Shen, K.-C. Toh, and S. Yun, An accelerated proximal gradient algorithm for frame-based image restoration via the balanced approach. *SIAM J. Imaging Sci.* **4** (2011), no. 2, 573–596.

[142] Z. Shen, H. Yang, and S. Zhang, Deep network with approximation error being reciprocal of width to power of square root of depth. *Neural Comput.* **33** (2021), no. 4, 1005–1036.

[143] Z. Shen, H. Yang, and S. Zhang, Neural network approximation: three hidden layers are enough. *Neural Netw.* **141** (2021), 160–173.

[144] J. W. Siegel and J. Xu, Optimal approximation rates and metric entropy of ReLU^k and cosine networks. 2021, arXiv:2101.12365.

[145] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot et al., Mastering the game of Go with deep neural networks and tree search. *Nature* **529** (2016), no. 7587, 484–489.

[146] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **15** (2014), no. 1, 1929–1958.

[147] J. Starck, M. Elad, and D. Donoho, Image decomposition via the combination of sparse representations and a variational approach. *IEEE Trans. Image Process.* **14** (2005), no. 10, 1570–1582.

[148] G. Steidl, J. Weickert, T. Brox, P. Mrázek, and M. Welk, On the equivalence of soft wavelet shrinkage, total variation diffusion, total variation regularization, and sides. *SIAM J. Numer. Anal.* (2005), 686–713.

[149] W. Su, S. Boyd, and E. Candes, A differential equation for modeling Nesterov's accelerated gradient method: theory and insights. *Adv. Neural Inf. Process. Syst.* **27** (2014), 2510–2518.

[150] W. Su, S. Boyd, and E. J. Candes, A differential equation for modeling Nesterov's accelerated gradient method: theory and insights. *J. Mach. Learn. Res.* **17** (2016), no. 153, 1–43.

[151] Q. Sun, Y. Tao, and Q. Du, Stochastic training of residual networks: a differential equation viewpoint. 2018, arXiv:1812.00174.

[152] Y. Sun, L. Zhang, and H. Schaeffer, Neupde: neural network based ordinary and partial differential equations for modeling time-dependent data. In *Mathematical and Scientific Machine Learning*, pp. 352–372, Proceedings of Machine Learning Research, 2020.

[153] M. Thorpe and Y. van Gennip, Deep limits of residual neural networks. 2018, arXiv:1810.11741.

[154] A. Tikhonov, V. Arsenin, and F. John, *Solutions of ill-posed problems*. VH Winston, Washington, DC, 1977.

[155] P. Tseng, Applications of a splitting algorithm to decomposition in convex programming and variational inequalities. *SIAM J. Control Optim.* **29** (1991), no. 1, 119–138.

[156] H. Wang, Y. Wu, M. Li, Q. Zhao, and D. Meng, A survey on rain removal from video and single image. 2019, arXiv:1909.08326.

[157] W. Wang, S. Axelrod, and R. Gómez-Bombarelli, Differentiable molecular simulations for control and learning. 2020, arXiv:2003.00868.

[158] Y. Wang, Learning to discretize: solving 1D scalar conservation laws via deep reinforcement learning. *Commun. Comput. Phys.* **28** (2020), no. 5, 2158–2179.

[159] Z. Wang, J. Chen, and S. C. Hoi, Deep learning for image super-resolution: a survey. *IEEE Trans. Pattern Anal. Mach. Intell.* (2020), Vol. 43, pp. 3365–3387.

[160] R. F. Warming and B. Hyett, The modified equation approach to the stability and accuracy analysis of finite-difference methods. *J. Comput. Phys.* **14** (1974), no. 2, 159–179.

[161] J. Weickert, *Anisotropic diffusion in image processing*. Teubner, Stuttgart, 1998.

[162] E. Weinan, A proposal on machine learning via dynamical systems. *Commun. Math. Stat.* **5** (2017), no. 1, 1–11.

[163] E. Weinan, C. Ma, and L. Wu, A priori estimates of the population risk for two-layer neural networks. *Commun. Math. Sci.* **17** (2019), no. 5, 1407–1425.

[164] E. Weinan and Q. Wang, Exponential convergence of the deep neural network approximation for analytic functions. *Sci. China Math.* **61** (2018), no. 10, 1733–1740.

[165] S. Wiewel, M. Becher, and N. Thuerey, Latent space physics: towards learning the temporal evolution of fluid flow. *Comput. Graph. Forum* **38** (2019), 71–82. Wiley Online Library.

[166] J.-L. Wu, K. Kashinath, A. Albert, D. Chirila, H. Xiao et al., Enforcing statistical constraints in generative adversarial networks for modeling chaotic dynamical systems. *J. Comput. Phys.* **406** (2020), 109209.

[167] Z. Wu and R. Zhang, Learning physics by data for the motion of a sphere falling in a non-Newtonian fluid. *Commun. Nonlinear Sci. Numer. Simul.* **67** (2019), 577–593.

[168] X. Xie, F. Bao, T. Maier, and C. Webster, Analytic continuation of noisy data using Adams Bashforth residual neural network. *Discrete Contin. Dyn. Syst. Ser. S* (2021), doi: 10.3934/dcdss.2021088.

[169] Y. Xie, E. Franz, M. Chu, and N. Thuerey, tempoGAN: a temporally coherent, volumetric GAN for super-resolution fluid flow. *ACM Trans. Graph.* **37** (2018), no. 4, 1–15.

[170] D. Yarotsky and A. Zhevnerchuk, The phase diagram of approximation rates for deep neural networks. 2019, arXiv:1906.09477.

[171] D. Zhang, T. Zhang, Y. Lu, Z. Zhu, and B. Dong, You only propagate once: accelerating adversarial training via maximal principle. *Adv. Neural Inf. Process. Syst.* **32** (2019), 227–238.

[172] H.-M. Zhang and B. Dong, A review on deep learning in medical image reconstruction. *J. Oper. Res. Soc. China* (2020), 1–30.

[173] J. Zhang, B. Han, L. Wynter, L. K. Hsiang, and M. Kankanhalli, Towards robust ResNet: a small step but a giant leap. In *28th International Joint Conference on Artificial Intelligence*, 2019.

[174] X. Zhang, Z. Li, C. Change Loy, and D. Lin, Polynet: a pursuit of structural diversity in very deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 718–726, 2017.

[175] X. Zhang, J. Liu, Y. Lu, and B. Dong, Dynamically unfolding recurrent restorer: a moving endpoint control method for image restoration. In *International Conference on Learning Representations*, 2019.

[176] M. Zhu and T. Chan, An efficient primal–dual hybrid gradient algorithm for total variation image restoration. *UCLA CAM Report* **34** (2008).

[177] M. Zhu, B. Chang, and C. Fu, Convolutional neural networks combined with Runge–Kutta methods. 2018, arXiv:1802.08831.

BIN DONG

Beijing International Center for Mathematical Research, National Engineering Laboratory for Big Data Analysis and Applications, National Biomedical Imaging Center, Institute for Artificial Intelligence, Peking University, Beijing, China, dongbin@math.pku.edu.cn