Image Denoising using Convolutional Sparse Coding Network with Dry Friction*

Yali Zhang[1], Xiaofan Wang[1], Fengpin Wang[1], and Jinjia Wang[1,2][0000−0002−2210−5570]

[1] School of Information Science and Engineering, Yanshan University, Qinhuangdao, 066004, China
wjj@ysu.edu.cn
[2] Hebei Key Laboratory of Information Transmission and Signal Processing, Yanshan University, Qinhuangdao, 066004, China

Abstract. Convolutional sparse coding model has been successfully used in some tasks such as signal or image processing and classification. The recently proposed supervised convolutional sparse coding network (CSC-Net) model based on the Minimum Mean Square Error (MMSE) approximation shows the similar PSNR value for image denoising problem with state of the art methods while using much fewer parameters. The CSC-Net uses the learning convolutional iterative shrinkage-thresholding algorithms (LISTA) based on the convolutional dictionary setting. However, LISTA methods are known to converge to local minima. In this paper we proposed one novel algorithm based on LISTA with dry friction, named LISTDFA. The dry friction enters the LISTDFA algorithm through proximal mapping. Due to the nature of dry friction, the LISTDFA algorithm is proven to converge in a finite time. The corresponding iterative neural network preserves the computational simplicity of the original CSCNet, and can reach a better local minima practically.

Keywords: Image denoising · Convolutional sparse coding · Iterative shrinkage thresholding algorithms · Dry Friction.

1 Introduction

Noise is the pollution of image in the process of acquisition, compression and transmission, which is easy to cause the loss of image information and bring adverse effects on image processing[13, 27]. Image denoising is the process of removing noise from the image polluted by noise and restoring the original image. In recent years, it has been paid attention as the basis of other image processing and is one of the key issues in the field of image processing. In order to achieve

* This work was supported in part by Basic Research Cooperation Projects of Beijing, Tianjin and Hebei (F2019203583,19JCZDJC65600), and the Central Funds Guiding the Local Science and Technology Development (Basic Research Projects)(206Z5001G).

image denoising, we must know the prior information of the original signal. Image priors, also known as image models, involve the mathematical description of the true distribution of images. Convolutional sparse coding (CSC) is a popular and important prior model in the field of signal processing and machine learning [29]. CSC with the banded convolutional structure constrained forms has a solid theoretical foundation and an uniqueness of the solution. CSC has many applications in the field of image processing, such as inverse problems [21], image reconstruction [20, 31, 36, 37], image denoising [28, 30], image inpainting [16, 20, 37] and so on. Along with the application of deep learning, some advanced image reconstruction methods based on convolutional neural networks have achieved excellent performance in image denoising [10, 33–35]. In state of the art studies, the improvement of neural network module makes the reconstruction model more interactive in industrial and commercial scenarios [17, 18].

Iterative neural network [7] is a kind of unfold network based on the iterative algorithm. Iterative neural network combines a forward neural network and an iterative algorithm, leading to good generalization capability over an iterative algorithm. For example, an iterative threshold shrink algorithm (ISTA) is a simple algorithm for the sparse representation problem. The learning iterative threshold shrink algorithm (LISTA) [15] is an iterative neural network that can be trained through supervised learning. Moreover, a convolutional iterative threshold shrink algorithm is a simple algorithm for the CSC problem. The learning convolutional iterative threshold shrink algorithm, named learning convolutional sparse coding, is an corresponding iterative neural network [24]. Similarly, a deep coupled ISTA network is proposed for multi-modal image super-resolution problem [9]. A deep convolutional sparse coding network is proposed for jpeg artifacts reduction [14]. A deep coupled convolutional sparse coding network is proposed for pan-sharpening CSC problem [32]. Inspired by multilayer neural network, a multilayer ISTA is proposed and a multilayer ISTA network is proposed for classification [26].

Recently, a supervised convolution sparse coding network (CSCNet) is proposed for image denoising problem [22]. CSCNet is based on the ISTA algorithm and LISTA network [15, 24]. It is trained via stochastic gradient-descent using self-supervised form. Thus, naturally CSCNet can learn the convolutional dictionary over very large datasets. Although it first answers a problem why the CSC model denoise natural images poorly, there are some problems in CSCNet. The ISTA iterative algorithm tends to jump into a local minima, leading to CSCNet producing the worse results for the application in the noise measurements.

In order to jump out of a local minimal of the ISTA algorithm, an iterative shrinkage-thresholding with dry friction algorithms (ISTDFA) is proposed. It is an improved proximal gradient algorithm with forward-backward splitting based on the a heavy ball system with dry friction [1]. The nature of dry friction can make the system reach a stable state in a finite time. Dry friction enters into the algorithm based on the proximal mapping. The proposed ISTDFA algorithm can effectively reduce the value of objective function of the ISTA algorithm. However, dry friction has a certain effect on the update of the convolution dictionary. The back propagation algorithms commonly used in deep learning do not take into account the effect of dry friction. The ISTDFA algorithm solves the problem throughout the recently proposed variance regularization [12]. It makes the update of the convolution dictionary independent of dry friction. Moreover, ISTDFA is extended to a learned iterative shrinkage-thresholding with dry friction algorithms (LISTDFA) based on iterative neural network. The novel iterative neural network, named CSCNet-DF, is proposed. CSCNet-DF includes an encoder and a decoder. The encoder uses LISTDFA to produce the coding from input images. The decoder is used to reconstruct the images. The minimum mean square error between the input images and output images is used to update the convolution dictionary by the back propagation algorithms.

Finally, the experimental results show that the proposed CSCNet-DF network is superior to CSCNet.

2 Related Work

2.1 Convolutional Sparse Coding Model

This sparse coding model assumes that a signal $\mathbf{y} = \mathbf{D\Gamma}$ is a linear combination of atoms, where $\mathbf{D} \in \mathbb{R}^{N \times M}$ is a dictionary and $\mathbf{\Gamma} \in \mathbb{R}^{M}$ is a sparse vector. For a given signal $\mathbf{y}$ and dictionary $\mathbf{D}$, the sparse representation of $\mathbf{y}$ is solved by the following optimization problem:

$$\min_{\mathbf{\Gamma}} F(\mathbf{\Gamma}) = \min_{\mathbf{\Gamma}} \frac{1}{2} \|\mathbf{y} - \mathbf{D\Gamma}\|_2^2 + \lambda \|\mathbf{\Gamma}\|_1 \ . \tag{1}$$

The solution of this sparse coding problem (1) is unique and can be obtained by many classical algorithms, such as the Orthogonal Matching Pursuit (OMP) [5], and Basis Pursuit (BP) [6]. The corresponding classical dictionary learning methods include MOD [11], trainlets [25], online dictionary learning [19], K-SVD [2], and so on. If D is a shift-invariant convolutional dictionary, this problem (1) is changed as the convolutional sparse coding problem. The solution is obtained by the Fourier-based fast methods [4, 21]. But the Fourier-based signal representation loses signal localization.

Assume that a signal $\mathbf{y}$ can be expressed as $\mathbf{y} = \mathbf{D\Gamma} = \sum_{i=1}^{N} \mathbf{P}_i^T \mathbf{D}_L \boldsymbol{\alpha}_i$. All shifted version of the local dictionary $\mathbf{D}_L \in \mathbb{R}^{n \times m}$ compose the convolutional dictionary $\mathbf{D}$. And all sparse vectors $\boldsymbol{\alpha}_i$ compose the sparse vector $\mathbf{\Gamma}$. The slice $\mathbf{s}_i$ is defined as $\mathbf{s}_i = \mathbf{D}_L \boldsymbol{\alpha}_i$, represents the i-th slice. The $\mathbf{P}_i^T$ represents the operation that put $\mathbf{D}_L \boldsymbol{\alpha}_i$ in the i-th position of the signal. The slice-based convolution sparse coding is proposed [20], which can be reformulated as

$$\min_{\boldsymbol{\alpha}_i} \frac{1}{2} \|\mathbf{y} - \sum_{i=1}^{n} \mathbf{P}_i^T \mathbf{s}_i\|_2^2 + \lambda \sum_{i=1}^{n} \|\boldsymbol{\alpha}_i\|_1 \ . \tag{2}$$

As far as we know, K-SVD [2] is one of the method to resolve the CSC. Both SBDL [20] and LoBCoD [37] are effective algorithms to solve the CSC problem (2). All of them have great dependence on alternating optimization algorithm, which makes the algorithms have more iterations.

2.2 Convolutional Sparse Coding Network

We believe that the CSC model is suitable for processing texture signals, leading to Gabor-like non-smooth filters. It is used in some problems, such as cartoon texture separation, image fusion or single image super-resolution to model the texture part of an image. However, the CSC model cannot cope with image denoising or other inverse problems involving noisy signals. In other words, the CSC model can only be used to model noise-free images. Simon et al. [23] argue that a convolutional dictionary with high coherence over-fits the noisy signal, so the CSC model is not suitable for the natural image denoising problem. This is because the filters in the CSC model cannot simultaneously meet the conditions of low global cross-correlation and low auto-correlation. To solve this problem, a new idea has been proposed [23]. To suppress the dictionary coherence and obtain the optimal solution, a larger stride size in the convolutional dictionary should be chosen. Unlike the basic CSC model with $q = 1$, the filters can be partially overlap when the stride q is chosen in the range of $1 \leq q \leq n$ and q is large enough. In this case, the convolutional dictionary is guaranteed to be mutually consistent even for the smooth filters. A CSCNet network is proposed for the image denoising problem [22]. CSCNet is an iterate neural network that unfolds the ISAT algorithm [24], which is a supervised denoising model. CSCNet first replicates the input noisy images q^2 times to obtain the possible q^2 shifted versions. Then the sparse coefficients is computed by CSCNet. Finally, the denoised reconstructed image is obtained by calculating the average image of all the shifted reconstructed images. The convolutional dictionary and network parameters are updated by back-propagation algorithm based on the minimization of the mean square error between the clean and denoised images. Experimental results show that CSCNet not only obtains similar denoising performance to SOTA supervised methods, but also uses only fewer neural network parameters.

2.3 Variance Regularization

The goal of convolution sparse coding is to find the corresponding sparse vector $\mathbf{\Gamma}$ and convolution dictionary $\mathbf{D}$ on the premise of a given signal $\mathbf{y}$. For any positive constant c, we can obtain the same reconstruction from the re-scaled convolutional dictionary $D' = c\mathbf{D}$ and sparse vector $\mathbf{\Gamma}' = \mathbf{\Gamma}/c$. If $c > 1$ holds, then we can get the same reconstruction from a smaller ℓ_1 norm of $\mathbf{\Gamma}'$ than $\mathbf{\Gamma}$. If there is no upper limit on the values of the convolution dictionary $\mathbf{D}$, which means the values of the needles can be arbitrarily small, leading to the collapse of the ℓ_1 norm of the sparse vector. In order to avoid the collapse of ℓ_1 regularization of sparse vector, it is often necessary to restrict convolution dictionary $\mathbf{D}$ in optimization problems. In practice, the column of the convolution dictionary $\mathbf{D}$ is often limited to a constant ℓ_2 norm. However, this is a very challenging thing to expand to a network.

In recent work, the variance principle [3] is presented, which uses a regularization term on the variance of the embeddings on each dimension individually to avoid the collapse of the sparse vector. Similar to this method, Yann et al.

[12] add a regularization term to the minimized objection function to neutralize the effect of $\mathbf{D}$'s weights, which ensures the latent sparse vector components have greater variances than a fixed threshold over the sparse representations for given inputs. This strategy also plays a role in one iterative neural network because the variance regularization term is independent from the dictionary $\mathbf{D}$'s architecture.

3 Proposed CSCNet-DF Network

3.1 Iterative Shrinkage-Thresholding with Dry Friction Algorithm

Throughout this section $\mathbb{H}$ is a real Hilbert space, and the associated norm $\|\cdot\|$. The minimized objective function has the following form $f + g$, where the non-convex smooth function $f : \mathbb{H} \to \mathbb{R}$ represents a L-Lipschitz continuous function, and the convex non-smooth function $g : \mathbb{H} \to \mathbb{R} \cup \{+\infty\}$ denotes a proper lower semi-continuous function.

For the non-smooth non-convex potential function $f + g$, we can get the corresponding heavy ball with dry friction system [1]

$$\mathbf{x}''(t) + \gamma \mathbf{x}'(t) + \partial\phi\left(\mathbf{x}'(t)\right) + \bigtriangledown f\left(\mathbf{x}(t)\right) + \partial g\left(\mathbf{x}(t)\right) \ni 0 \ . \tag{3}$$

The system contains two different kinds of damping:

(1) Viscous damping: $\gamma \mathbf{x}'(t)$ is viscous damping, and γ is a viscous damping coefficient satisfying $\gamma > 0$.

(2) Dry friction damping: $\partial\phi\left(\mathbf{x}'(t)\right)$ represents dry friction. ϕ is the dry friction potential function, which has the following properties: ϕ is a convex and lower semi-continuous function, and ϕ has a sharp minimum at the origin, that is, $\min_{\xi \in H} \phi(\xi) = \phi(0) = 0$.

According to the properties of dry friction ϕ, we assume that $\phi(\mathbf{x}) = r\|\mathbf{x}\|_1$, and r is the dry friction coefficient satisfying $r > 0$. From the definition of the function ϕ, the dry friction properties are satisfied with $B(0, r) \subset \partial\phi(0)$.

The time discretization of Equation (3) with a fixed time step size $h > 0$ is given as follows

$$\frac{\left(\mathbf{x}^{k+1}-\mathbf{x}^{k}\right)-\left(\mathbf{x}^{k}-\mathbf{x}^{k-1}\right)}{h^2} + \frac{\gamma\left(\mathbf{x}^{k+1}-\mathbf{x}^{k}\right)}{h} + \partial\phi\left(\frac{\mathbf{x}^{k+1}-\mathbf{x}^{k}}{h}\right) + \bigtriangledown f\left(\mathbf{x}^{k}\right) + \partial g\left(\mathbf{x}^{k+1}\right) \ni 0 \ . \tag{4}$$

Equation (4) relates the classical dynamics method to the proximal gradient methods: the smooth function f is implicit in Equation (4), corresponding to the gradient step of the proximal gradient method; similarly, non-smooth functions g and ϕ are explict in Equation (4), corresponding to the proximal step of the proximal gradient method.

For each $k \in \mathbb{N}$, let's introduce the auxiliary convex function defined by

$$\phi_k\left(\mathbf{x}^k + hx\right) := h\phi(x) \ . \tag{5}$$

Then, we have $\partial\phi_k\left(\mathbf{x}^{k+1}\right)=\partial\phi_k\left(\mathbf{x}^k+h\frac{\mathbf{x}^{k+1}-\mathbf{x}^k}{h}\right)=\partial\phi\left(\frac{\mathbf{x}^{k+1}-\mathbf{x}^k}{h}\right)$. And then we can induce that from (4)

$$\begin{array}{l}\frac{1}{h^2}\left(\mathbf{x}^{k+1}-2\mathbf{x}^k+\mathbf{x}^{k-1}\right)+\frac{\gamma}{h}\left(\mathbf{x}^{k+1}-\mathbf{x}^k\right)\\+\partial\phi_k\left(\mathbf{x}^{k+1}\right)+\bigtriangledown f\left(\mathbf{x}^k\right)+\partial g\left(\mathbf{x}^{k+1}\right)\ni 0\end{array}. \tag{6}$$

Since ϕ is continuous, we have $\partial\phi_k+\partial g=\partial\left(\phi_k+g\right)$, which implies

$$\tfrac{1+\gamma h}{h^2}\mathbf{x}^{k+1}+\partial\left(g+\phi_k\right)\left(\mathbf{x}^{k+1}\right)\ni\tfrac{2+\gamma h}{h^2}\mathbf{x}^k-\tfrac{1}{h^2}\mathbf{x}^{k-1}-\bigtriangledown f\left(\mathbf{x}^k\right)\ . \tag{7}$$

We finally get the iterative shrinkage-thresholding with dry friction algorithm (ISTDFA),

$$\mathbf{x}^{k+1}=\operatorname{prox}_{\frac{h^2}{1+\gamma h}\left(g+\phi_k\right)}\left(\mathbf{x}^{k+1/2}-\frac{h^2}{1+\gamma h}\bigtriangledown f\left(\mathbf{x}^k\right)\right)\ , \tag{8}$$

where $\mathbf{x}^{k+1/2}=\mathbf{x}^k+\frac{1}{1+\gamma h}\left(\mathbf{x}^k-\mathbf{x}^{k-1}\right)$, and the proximal map is defined as follows:

$$\operatorname{prox}_{\eta p}\left(\mathbf{x}\right):=\arg\min_{\boldsymbol{\xi}\in\mathbb{H}}\left\{\eta p\left(\boldsymbol{\xi}\right)+\frac{1}{2}\left\|\mathbf{x}-\boldsymbol{\xi}\right\|^2\right\}\ . \tag{9}$$

Theorem 1. *Suppose that the parameters $h>0$, $\gamma>0$ satisfy $h<\frac{2\gamma}{L}$. Then for the sequence $\left\{\mathbf{x}^k\right\}$ genereted by the algorithm (ISTDFA) we have:*
(1) $\sum_k\left\|\boldsymbol{x}^{k+1}-\boldsymbol{x}^k\right\|<+\infty$, and therefore $\lim_k\boldsymbol{x}^k=\boldsymbol{x}^*$ exists for the strong topology of $\mathbb{H}$.
(2) The vector $\boldsymbol{x}^*$ satisfies $0\in\partial\phi\left(0\right)+\bigtriangledown f\left(\boldsymbol{x}^*\right)+\partial g\left(\boldsymbol{x}^*\right)$.
(3) Suppose that $\mathbb{H}$ is finite dimensional, and suppose that $-\left(\bigtriangledown f\left(\boldsymbol{x}^*\right)+\partial g\left(\boldsymbol{x}^*\right)\right)\in int\left(\partial\left(0\right)\right)$. Then, the sequence $\left\{\mathbf{x}^k\right\}$ is finitely convergent.

3.2 Application ISTDFA to CSC Model

Consider the novel minimization problem for CSC model via local processing and variance regularization as follows:

$$\arg\min_{\left\{\boldsymbol{\alpha}_{l,i}\right\}_{i=1}^N}\frac{1}{2}\sum_{l=1}^{I}\left\|\sum_{i=1}^{N}\mathbf{P}_i^T\mathbf{D}_L\boldsymbol{\alpha}_{l,i}-\mathbf{y}_l\right\|_2^2+\lambda\sum_{l=1}^{I}\sum_{i=1}^{N}\left\|\boldsymbol{\alpha}_{l,i}\right\|_1+\beta\sum_{i=1}^{N}\left[\left(T-\sqrt{\mathrm{Var}(\boldsymbol{\alpha}_{\cdot i})}\right)_+\right]^2\ , \tag{10}$$

where $\mathbf{D}_L$ is the local convolutional dictionary which has n rows and m columns; $\boldsymbol{\alpha}_{l,i}$ is the sparse coding of each component i of each sample l; $\mathbf{P}_i^T$ which has N rows and n columns is the operator that puts $\mathbf{D}_L\boldsymbol{\alpha}_{l,i}$ in the i-th position and pads the rest of the entries with zero; $\mathbf{y}_l$ is the nosiy signal; $\left\|\cdot\right\|_2$ stands for vector of ℓ_2 norm or Frobenius norm of matrix; λ is the super parameter. When a noisy signal $\mathbf{y}_l=\mathbf{x}_l+\mathbf{v}_l\in\mathbb{R}^N$ is at hand, seeking for its sparse representation $\hat{\boldsymbol{\alpha}}_{l,i}$, leads to an estimation of the original signal via $\hat{\mathbf{x}}_l=\mathbf{P}_i^T\mathbf{D}_L\hat{\boldsymbol{\alpha}}_{l,i}$.

In order to keep the variance of each potential code component remains above a preset threshold, a regularization term is added in (10). For the non-smooth convex optimization model, we define f and g as follows:

$$f\left(\boldsymbol{\alpha}_{l,i}\right)=\frac{1}{2}\sum_{l=1}^{I}\left\|\sum_{i=1}^{N}\mathbf{P}_i^T\mathbf{D}_L\boldsymbol{\alpha}_{l,i}-\mathbf{y}_l\right\|_2^2+\beta\sum_{i=1}^{N}\left[\left(T-\sqrt{\operatorname{Var}\left(\boldsymbol{\alpha}_{\cdot i}\right)}\right)_+\right]^2 \ , \quad (11)$$

$$g\left(\boldsymbol{\alpha}_{l,i}\right)=\lambda\sum_{l=1}^{I}\sum_{i=1}^{N}\left\|\boldsymbol{\alpha}_{l,i}\right\|_1 \ . \quad (12)$$

The first item in (11) is the reconstruction term. The second item in (11) is over squared hinge terms involving the variance of each latent component $\boldsymbol{\alpha}_{\cdot i}\in\mathbb{R}^n$ across the batch where $\operatorname{Var}\left(\boldsymbol{\alpha}_{\cdot i}\right)=\frac{1}{I-1}\sum_{l=1}^{I}\left(\boldsymbol{\alpha}_{l,i}-\mu_i\right)^2$ and μ_i is the mean across the i-th latent component, namely $\mu_i=\frac{1}{I}\sum_{l=1}^{I}\boldsymbol{\alpha}_{l,i}$. The hinge terms are non-zero for any latent dimension whose variance is below the fixed threshold of $\sqrt{T}$.

For solving the CSC problem, we extended the LISTA to the learning ISTDFA. We proposed convolutional learning ISTDFA algorithm to approximate the convolutional sparse coding model, which is presented in Algorithm 1. The input of the proposed convolutional learning ISTDFA algorithm are the noise signal $\mathbf{y}_l$, the dictionary $\mathbf{D}_L$. Some parameters are firstly initialized. Going into the algorithm, we first compute the gradient of f and the local Lipschitz constant respectively, then $\boldsymbol{\alpha}_{l,i}^{k+1}$ can be updated by the proximal mapping.

Next we can induce the gradient of f,

$$\bigtriangledown f=\begin{cases}\mathbf{D}_L^T\mathbf{P}_b\left(\sum\limits_{i=1}^{N}\mathbf{P}_i^T\mathbf{D}_L\boldsymbol{\alpha}_{a,i}-\mathbf{y}_a\right)-\frac{2\beta}{I-1}\frac{\left(T-\sqrt{\operatorname{Var}(\boldsymbol{\alpha}_{\cdot b})}\right)}{\sqrt{\operatorname{Var}(\boldsymbol{\alpha}_{\cdot b})}}(\boldsymbol{\alpha}_{a,b}-\mu_b), & \sqrt{\operatorname{Var}(\boldsymbol{\alpha}_{\cdot b})}<T\\ \mathbf{D}_L^T\mathbf{P}_b\left(\sum\limits_{i=1}^{N}\mathbf{P}_i^T\mathbf{D}_L\boldsymbol{\alpha}_{a,i}-\mathbf{y}_a\right), & otherwise\end{cases} . \quad (13)$$

Now, let us analyze the computation of proximal function. From the definition of ϕ and the relationship of ϕ and ϕ_k, we induce that $\phi_k\left(\mathbf{x}\right)=r\left\|\mathbf{x}-\mathbf{a}\right\|_1$, where $\mathbf{a}=\mathbf{x}_k$.

Setting $\lambda=\frac{h^2}{1+\gamma h}$, from the definition of proximal mapping, we can induce that

$$\operatorname{prox}_{\lambda(\phi_k+g)}\left(\mathbf{x}\right)=\arg\min_{\mathbf{u}}\frac{1}{2}\left\|\mathbf{x}-\mathbf{u}\right\|^2+\lambda\left\|\mathbf{u}\right\|_1+\lambda r\left\|\mathbf{u}-\mathbf{a}\right\|_1 \ . \quad (14)$$

By noticing that $\operatorname{prox}_{\lambda(\phi_k+g)}$ is a separable optimization problem, it can be reduced to the computation component-wise of a one dimensional optimization problem. For each $a\in\mathbb{R}$, set

$$T_a\left(x\right)=\arg\min_{u}\frac{1}{2}\left(u-x\right)^2+\lambda r\left|u-a\right|_1+\lambda\left|u\right|_1 \ . \quad (15)$$

Observe that $T_a\left(x\right)=-T_{-a}\left(-x\right)$. So, we just need to consider the case $a\geq 0$.

Using the discontinuous but differentiable of ℓ_1 regularization, we can get that $\lambda r \partial \left|u-a\right|_1 + \lambda \partial \left|u\right|_1 \ni x - u$.

(1) When $u > a$, $\partial \left|u-a\right|_1 = 1$, $\partial \left|u\right|_1 = 1$, then $u = \mathrm{prox}_{\lambda(\phi_k+g)}(x) = x - \lambda(1+r) > a$;

(2) When $a < u < 0$, $\partial \left|u-a\right|_1 = -1$, $\partial \left|u\right|_1 = 1$, then $a < u = \mathrm{prox}_{\lambda(\phi_k+g)}(x) = x - \lambda(1-r) < 0$;

(3) When $u < 0$, $\partial \left|u-a\right|_1 = -1$, $\partial \left|u\right|_1 = -1$, then $u = \mathrm{prox}_{\lambda(\phi_k+g)}(x) = x + \lambda(1+r) < 0$;

(4) When $u = a$, $\partial \left|u-a\right|_1 = \{-1,1\}$, $\partial \left|u\right|_1 = 1$, then $u = \mathrm{prox}_{\lambda(\phi_k+g)}(x) = x - \lambda(1 + r\{-1,1\}) = a$;

(5) When $u = 0$, $\partial \left|u-a\right|_1 = 1$, $\partial \left|u\right|_1 = \{-1,1\}$, then $u = \mathrm{prox}_{\lambda(\phi_k+g)}(x) = x - \lambda(\{-1,1\} + r) = 0$;

Through the above analysis, we can get that the unique solution of $T_a(x)$ is

$$T_a(x) = \begin{cases} x - \lambda(1+r) & x > \lambda(1+r) + a \\ a & \lambda(1-r) + a < x < \lambda(1+r) + a \\ x - \lambda(1-r) & \lambda(1-r) < x < \lambda(1-r) + a \\ 0 & -\lambda(1+r) < x < \lambda(1-r) \\ x + \lambda(1+r) & x < -\lambda(1+r) \end{cases}. \tag{16}$$

This is a threshold operator with two critical values a and 0. Consequently, for each $i = 1, 2, ..., n$, we have $\left(\mathrm{prox}_{\lambda(\phi_k+g)}(x)\right)_i$ is

$$\left(\mathrm{prox}_{\lambda(\phi_k+g)}(x)\right)_i = \begin{cases} T_{a_i}(x_i) & a_i \geq 0 \\ -T_{-a_i}(-x_i) & a_i \leq 0 \end{cases}, \tag{17}$$

with a_i the i-th component of the vector $\mathbf{a} = \mathbf{x}^k$ and $\lambda = \frac{h^2}{1+\gamma h}$.

The iterative shrinkage-thresholding with dry friction algorithm for the CSC model is proposed in Algorithm 1.

Algorithm 1 Iterative shrinkage-thresholding with dry friction algorithm for CSC.

Input: noised signal $\mathbf{y}_l$, local convolutional dictionary $\mathbf{D}_L$.

Output: Estimated coding $\boldsymbol{\alpha}_{l,i}^k$.

Initialization: $\boldsymbol{\alpha}_{l,i}^0 = \boldsymbol{\alpha}_{l,i}^1, \gamma > 0, h < \frac{2\gamma}{L}$

For iteration k= 0:$K-1$

compute $\bigtriangledown f\left(\boldsymbol{\alpha}_{l,i}^k\right)$ using 13,

$\boldsymbol{\alpha}_{l,i}^{k+1/2} = \boldsymbol{\alpha}_{l,i}^k + \frac{1}{1+\gamma h}\left(\boldsymbol{\alpha}_{l,i}^k - \boldsymbol{\alpha}_{l,i}^{k-1}\right)$,

$\boldsymbol{\alpha}_{l,i}^{k+1} = \mathrm{prox}_{\frac{h^2}{1+\gamma h}(\phi_k+g)}\left(\boldsymbol{\alpha}_{l,i}^{k+1/2} - \frac{h^2}{1+\gamma h}\bigtriangledown f\left(\boldsymbol{\alpha}_{l,i}^k\right)\right)$,

end for

3.3 Convolutional Sparse Coding Network with Dry Friction

Through Algorithm 1, the update formula of needles can be written as:

$$\boldsymbol{\alpha}_{l,i}^{k+1} = \operatorname{prox}_{\frac{h^2}{1+\gamma h}(\phi_k+g)} \left(\boldsymbol{\alpha}_{l,i}^{k} + \frac{1}{1+\gamma h} \left(\boldsymbol{\alpha}_{l,i}^{k} - \boldsymbol{\alpha}_{l,i}^{k-1} \right) - \frac{h^2}{1+\gamma h} \bigtriangledown f\left(\boldsymbol{\alpha}_{l,i}^{k}\right) \right), \tag{18}$$

where $\frac{1}{1+\gamma h}\left(\boldsymbol{\alpha}_{l,i}^{k} - \boldsymbol{\alpha}_{l,i}^{k-1}\right)$ represents the inertia item.

The CSCNet network with dry friction (CSCNet-DF) is proposed based on (18) in the Algorithm 1. The network diagram of CSCNet-DF is presented in Fig. 1. Generally speaking, convergence requires a lot of times to achieve, which will bring great computational burden. In order to overcome this burden, we adopt the calculation idea of LISTA algorithm by learning the parameters $\mathbf{A}$ and $\mathbf{B}$ of the nonlinear recursive encoder strictly following (18), where $\mathbf{A}$ is a convolution operator and $\mathbf{B}$ is a transposed-convolution operation. Once the needles are at hand, the estimated clean image is then obtained by a linear transposed-convolutional decoder, $\hat{\mathbf{x}} = \mathbf{C}\boldsymbol{\Gamma}$, where $\boldsymbol{\Gamma}$ is a matrix composed of needles $\boldsymbol{\alpha}_{l,i}$. In this paper, matrices $\mathbf{A}, \mathbf{B}$ and $\mathbf{C}$ are constructed as a set of bounded shift invariant filters, and self-supervised learning is carried out together with thresholds $\frac{h^2}{1+\gamma h}$.

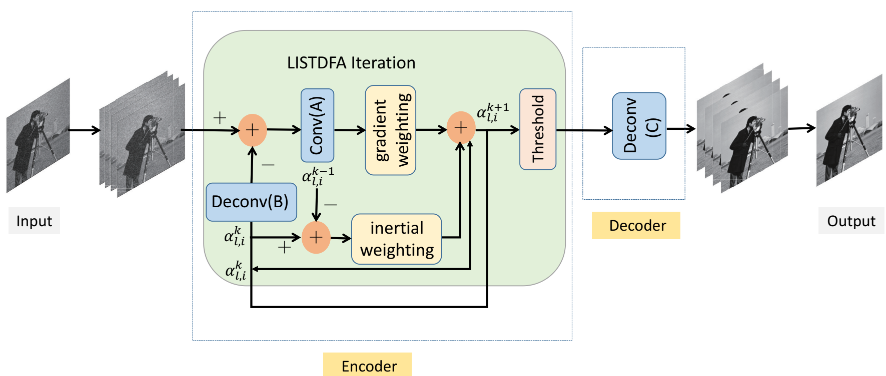


Fig. 1. The CSCNet-DF network. The encoder is LISTDFA Iteration. The decoder is Deconv(C) module.

Our proposed network structure is based on the CSCNet structure. The input is the noised image, which is duplicated many times using a shifted version. The output is denoised image, which is a simple average of the estimates of all the shifts. The main body of the network structure is the encoder and decoder. The encoder is the iterative network unfolded by the LISTDFA algorithm.

The decoder is the reconstruction process of an image. Compared with CSCNet, CSCNet-DF has the same decoder, while they have the different encoder unfolded by the LISTA and LISTDFA, respectively. In detail, our proposed network has some different unit, such as, gradient weighting, inertia weighting and threshold modules. Compared with CSCNet, our proposed network adds a gradient weight module and an inertia weight module. In a way, CSCNet is a special case of our proposed network without dry friction.

In order to improve the efficiency of the network, we use the back propagation algorithm to update the convolution dictionary by minimizing the loss function. The loss function is the Mean Square Error (MSE) between the reconstructed signal $\hat{\mathbf{y}}_l^k = \mathbf{P}_i^T \mathbf{D}_L \hat{\boldsymbol{\alpha}}_{l,i}^K$ and the original signal $\mathbf{y}_l$:

$$L_{MSE} = \frac{1}{I} \sum_{l=1}^{I} \left\| \mathbf{y}_l - \hat{\mathbf{y}}_l^K \right\|_2^2 \ , \tag{19}$$

where $\hat{\mathbf{y}}_l^k$ represents the reconstruction result of k-th iteration, and $\mathbf{y}_l$ represents the l-th of the original signal $\mathbf{y}$.

4 Experiments and Results

This section provides numerical results for image denoising on the Set12, BSD68 and color-BSD68 datasets to analyse the performances of the proposed CSCNet-DF based on the ISTDFA algorithm. All the experiments were run using Python and Pytorch in Linux system. The experimental settings are the same as the literature [22]. The experimental set of CSCNet-DF is $\gamma = 50$, $\beta = 1$, $T = 0.5$, $r = 0.1$, $h = 0.1$, which are selected by the grid search. In order to illustrate that ISTDFA have faster convergence than ISTA, we first compared ISTA algorithm and ISTA-DF algorithm on the reconstruction experiments on City dataset [20]. The experimental results are shown in Fig.2 in terms of objective function values changing at each iteration. In the case of the same objective function value, ISTDFA algorithm needs fewer iterations than ISTA algorithm, which indicates that ISTDFA algorithm has faster convergence speed.

The above image reconstruction experiments have demonstrated the convergence of the ISTDFA algorithm. In the following, we will mainly verify the performance of the proposed CSCNet-DF network according to the ISTDFA algorithm in image denoising experiments. Tab.1 presents the denoising performance (PSNR) results of the CSCNet [22] and CSCNet-DF networks on the Set12 datasets. Due to limited space, we do not show the learned filters. We observe Tab.1 and know that the proposed CSCNet-DF network outperforms CSCNet. To further validate the proposed iterative neural networks, Tab.2 presents the PSNR results of CSCNet, CSCNet-DF, the well-known BM3D [8] and DnCNN [34] models on the BSD68 dataset. Tab.3 presents the PSNR results of two CSCNet networks and the well-known BM3D [8], DnCNN [34], AdmFM-Net [18], SwinIR [17] models on color-BSD68 dataset. Experimental results on BSD68 and color-BSD68 datasets show that our proposed CSCNet-DF network

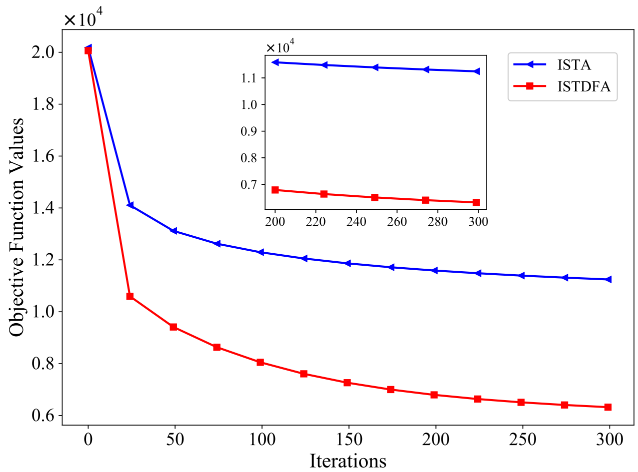


Fig. 2. Performance comparison of ISTDFA and ISTA in terms of objective function values changing at each iterations.

outperforms the BM3D, CSCNet and DnCNN. Experimental results on color-BSD68 dataset show that our proposed CSCNet-DF network outperforms the AdaFM-Net. The performance of our proposed network is slightly worse than that of SwinIR using Swin transformer, but relatively speaking, our network is simpler, with less parameters and less computation. Through the comparative analysis of the PSNR of the denoised images using different models, it can be found that the performance of CSCNet-DF is better than BM3D, CSCNet and DnCNN in practice. In addition, in order to make the experiment more convincing, we conducted a denoising experiment on the Set12 dataset, and the images are shown in Fig.3. It can be seen that compared with CSCNet, our proposed network texture is clearer and achieves better denoising performance.

Table 1. Denoising performance (PSNR) on the Set12 dataset.

σ	CSCNet	ours
15	31.23	32.66
25	28.73	30.24
50	25.32	27.14
75	23.28	25.35
time	1.67s	0.79s

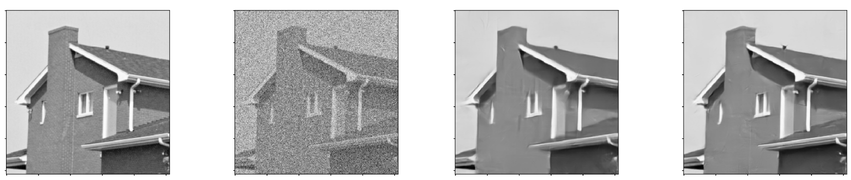
Finally, we design an ablation experiment to illustrate the role of dry friction in the network. We compare the image denoising performance with different dry

Table 2. PSNR performance on the BSD68 dataset.

σ	BM3D	CSCNet	DnCNN	ours
15	31.07	31.57	31.72	32.15
25	28.57	29.11	29.22	29.39
50	25.62	26.24	26.23	26.41
75	24.21	24.77	24.64	25.76

Table 3. PSNR performance on the color-BSD68 dataset.

σ	BM3D	CSCNet	DnCNN	AdaFM-Net	SwinIR	ours
15	33.52	31.81	33.89	34.10	34.42	34.14
25	30.71	29.30	31.23	31.35	31.78	31.39
50	27.38	26.37	27.92	27.95	28.56	28.08
75	25.74	24.87	24.47	26.35	26.45	26.38



(a) original image (b) noised image (c) CSCNet (d) CSCNet-DF

Fig. 3. Illustration of images denosing by CSCNet and CSCNet-DF. (a) The original image, (b) the noising image, (c) the denoising image using CSCNet, (d) the denoising image using CSCNet-DF.

friction coefficients on color datasets which is shown in Tab.4. The results show that dry friction can improve the network performance to a certain extent.

5 Conclusion

In this paper the iterative neural network based on the convolutional sparse coding model using the learned ISTDFA algorithm is proposed. Introduce of the dry friction, achieves fast convergence and low values of objection function for image denoising problem. The forward process of the proposed CSCNet-DF network includes encoder and decoder, giving the sparse coding and reconstructed signal. The backward process uses back propagation algorithms to update the convolutional dictionary. The experimental results on three dataset show that the CSCNet-DF network is superior to the CSCNet. In the future, the Nesterov accelerated method with dry friction will be studied.

Table 4. Comparison of PSNR performance under different dry friction coefficients on the color-BSD68 dataset ($\sigma = 15$).

r	0	0.01	0.05	0.1
PSNR(dB)	31.81	32.85	33.45	34.14

Acknowledgements We thank referee for comments that greatly improved the manuscript. Thanks to Jin Li and Ze Li of Yanshan University for their help in programming and experiment.

References

1. Adly, S., Attouch, H.: First-order inertial algorithms involving dry friction damping. Mathematical Programming **193**(1), 405–445 (2022)
2. Aharon, M., Elad, M., Bruckstein, A.: K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. IEEE Transactions on Signal Processing **54**(11), 4311–4322 (2006). https://doi.org/10.1109/TSP.2006.881199
3. Bardes, A., Ponce, J., LeCun, Y.: VICReg: Variance-invariance-covariance regularization for self-supervised learning. In: International Conference on Learning Representations (2022), https://openreview.net/forum?id=xm6YD62D1Ub
4. Chalasani, R., Principe, J.C., Ramakrishnan, N.: A fast proximal method for convolutional sparse coding. In: 2013 The International Joint Conference on Neural Networks (IJCNN). pp. 1–5 (2013). https://doi.org/10.1109/IJCNN.2013.6706854
5. Chen, S., Billings, S.A., Luo, W.: Orthogonal least squares methods and their application to non-linear system identification. International Journal of Control **50**(5), 1873–1896 (1989)
6. Chen, S.S., Saunders, D.M.A.: Atomic decomposition by basis pursuit. Siam Review **43**(1), 129–159 (2001)
7. Chun, I.Y., Huang, Z., Lim, H., Fessler, J.: Momentum-net: Fast and convergent iterative neural network for inverse problems. IEEE Transactions on Pattern Analysis and Machine Intelligence pp. 1–1 (2020). https://doi.org/10.1109/TPAMI.2020.3012955
8. Dabov, K., Foi, A., Katkovnik, V., Egiazarian, K.: Image denoising by sparse 3-d transform-domain collaborative filtering. IEEE Transactions on Image Processing **16**(8), 2080–2095 (2007). https://doi.org/10.1109/TIP.2007.901238
9. Deng, X., Dragotti, P.L.: Deep coupled ista network for multi-modal image super-resolution. IEEE Transactions on Image Processing **29**, 1683–1698 (2019)
10. Dong, C., Deng, Y., Loy, C.C., Tang, X.: Compression artifacts reduction by a deep convolutional network. In: 2015 IEEE International Conference on Computer Vision (ICCV). pp. 576–584 (2015). https://doi.org/10.1109/ICCV.2015.73
11. Engan, K., Aase, S., Hakon Husoy, J.: Method of optimal directions for frame design. In: 1999 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP). vol. 5, pp. 2443–2446 (1999). https://doi.org/10.1109/ICASSP.1999.760624
12. Evtimova, K., LeCun, Y.: Sparse coding with multi-layer decoders using variance regularization. Transactions on Machine Learning Research (2022), https://openreview.net/forum?id=4GuIi1jJ74

13. Fan, L., Zhang, F., Fan, H., Zhang, C.: Brief review of image denoising techniques. Visual Computing for Industry, Biomedicine, and Art **2**(1), 1–12 (2019)
14. Fu, X., Zha, Z.J., Wu, F., Ding, X., Paisley, J.: Jpeg artifacts reduction via deep convolutional sparse coding. In: 2019 IEEE/CVF International Conference on Computer Vision (CVPR). pp. 2501–2510 (2019)
15. Gregor, K., LeCun, Y.: Learning fast approximations of sparse coding. In: the 27th international conference on international conference on machine learning (ICML). pp. 399–406 (2010)
16. Guillemot, C., Le Meur, O.: Image inpainting : Overview and recent advances. IEEE signal processing magazine **31**(1), 127–144 (2014). https://doi.org/10.1109/MSP.2013.2273004, doi: 10.1109/MSP.2013.2273004
17. He, J., Dong, C., Qiao, Y.: Modulating image restoration with continual levels via adaptive feature modification layers. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 11048–11056 (2019). https://doi.org/10.1109/CVPR.2019.01131
18. Liang, J., Cao, J., Sun, G., Zhang, K., Van Gool, L., Timofte, R.: Swinir: Image restoration using swin transformer. In: 2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW). pp. 1833–1844 (2021). https://doi.org/10.1109/ICCVW54120.2021.00210
19. Mairal, J., Bach, F., Ponce, J., Sapiro, G.: Online dictionary learning for sparse coding. In: the 26th Annual International Conference on Machine Learning (ICML). p. 689–696 (2009). https://doi.org/10.1145/1553374.1553463
20. Papyan, V., Romano, Y., Elad, M., Sulam, J.: Convolutional dictionary learning via local processing. In: 2017 IEEE International Conference on Computer Vision (ICCV). pp. 5306–5314 (2017). https://doi.org/10.1109/ICCV.2017.566
21. Papyan, V., Sulam, J., Elad, M.: Working locally thinking globally: Theoretical guarantees for convolutional sparse coding. IEEE Transactions on Signal Processing **65**(21), 5687–5701 (2017). https://doi.org/10.1109/TSP.2017.2733447
22. Simon, D., Elad, M.: Rethinking the csc model for natural images. In: 2019 Neural Information Processing Systems (NeurIPS). p. 2274–2284 (2019). https://doi.org/10.5555/3454287.3454491
23. Simon, D., Sulam, J., Romano, Y., Lu, Y.M., Elad, M.: Mmse approximation for sparse coding algorithms using stochastic resonance. IEEE Transactions on Signal Processing **67**(17), 4597–4610 (2019). https://doi.org/10.1109/TSP.2019.2929464
24. Sreter, H., Giryes, R.: Learned convolutional sparse coding. In: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 2191–2195 (2018). https://doi.org/10.1109/ICASSP.2018.8462313
25. Sulam, J., Ophir, B., Zibulevsky, M., Elad, M.: Trainlets: Dictionary learning in high dimensions. IEEE Transactions on Signal Processing **64**(12), 3180–3193 (2016)
26. Sulam, J., Aberdam, A., Beck, A., Elad, M.: On multi-layer basis pursuit, efficient algorithms and convolutional neural networks. IEEE transactions on pattern analysis and machine intelligence **42**(8), 1968–1980 (2019)
27. Tian, C., Fei, L., Zheng, W., Xu, Y., Zuo, W., Lin, C.W.: Deep learning on image denoising: An overview. Neural Networks **131**, 251–275 (2020)
28. Wang, W., Xia, X.G., He, C., Ren, Z., Wang, T., Lei, B.: A noise-robust online convolutional coding model and its applications to poisson denoising and image fusion. Applied Mathematical Modelling **95**(1), 644–666 (2021). https://doi.org/10.1016/j.apm.2021.02.023

29. Wohlberg, B.: Efficient algorithms for convolutional sparse representations. IEEE Transactions on Image Processing **25**(1), 301–315 (2016). https://doi.org/10.1109/TIP.2015.2495260
30. Xu, J., Deng, X., Xu, M.: Revisiting convolutional sparse coding for image denoising: From a multi-scale perspective. IEEE Signal Processing Letters **29**, 1202–1206 (2022). https://doi.org/10.1109/LSP.2022.3175096
31. Yang, L., Li, C., Han, J., Chen, C., Ye, Q., Zhang, B., Cao, X., Liu, W.: Image reconstruction via manifold constrained convolutional sparse coding for image sets. IEEE Journal of Selected Topics in Signal Processing **11**(7), 1072–1081 (2017). https://doi.org/10.1109/JSTSP.2017.2743683
32. Yin, H.: Pscsc-net: A deep coupled convolutional sparse coding network for pansharpening. IEEE Transactions on Geoscience and Remote Sensing **60**, 1–16 (2022). https://doi.org/10.1109/TGRS.2021.3088313
33. Zhang, K., Li, Y., Zuo, W., Zhang, L., Van Gool, L., Timofte, R.: Plug-and-play image restoration with deep denoiser prior. IEEE Transactions on Pattern Analysis and Machine Intelligence **44**(10), 6360–6376 (2022). https://doi.org/10.1109/TPAMI.2021.3088914
34. Zhang, K., Zuo, W., Chen, Y., Meng, D., Zhang, L.: Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. IEEE Transactions on Image Processing **26**(7), 3142–3155 (2017). https://doi.org/10.1109/TIP.2017.2662206
35. Zhang, K., Zuo, W., Zhang, L.: Ffdnet: Toward a fast and flexible solution for cnn-based image denoising. IEEE Transactions on Image Processing **27**(9), 4608–4622 (2018). https://doi.org/10.1109/TIP.2018.2839891
36. Zheng, J., Zhang, X., Wang, W., Jiang, X.: Handling slice permutations variability in tensor recovery. In: 2022 the AAAI Conference on Artificial Intelligence. pp. 3499–3507 (2022). https://doi.org/10.1609/aaai.v36i3.20261
37. Zisselman, E., Sulam, J., Elad, M.: A local block coordinate descent algorithm for the csc model. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 8200–8209 (2019). https://doi.org/10.1109/CVPR.2019.00840