Sequential Convolution and Runge-Kutta Residual Architecture for Image Compressed Sensing*

Runkai Zheng[1 [0000−0003−3120−5466]], Yinqi Zhang[1 [0000−0003−4775−5147]], Daolang Huang[1 [0000−0001−6504−8898]], and Qingliang Chen**[1,2,3 [0000−0001−5849−8268]]

[1] Department of Computer Science, Jinan University, Guangzhou 510632, China.
[2] Guangzhou Xuanyuan Research Institute Company, Ltd., Guangzhou 510006, China.
[3] Guangdong E-Tong Software Co., Ltd., Guangzhou 510520, China.

Abstract. In recent years, Deep Neural Networks (DNN) have empowered Compressed Sensing (CS) substantially and have achieved high reconstruction quality and speed far exceeding traditional CS methods. However, there are still lots of issues to be further explored before it can be practical enough. There are mainly two challenging problems in CS, one is to achieve efficient data sampling, and the other is to reconstruct images with high-quality. To address the two challenges, this paper proposes a novel Runge-Kutta Convolutional Compressed Sensing Network (RK-CCSNet). In the sensing stage, RK-CCSNet applies Sequential Convolutional Module (SCM) to gradually compact measurements through a series of convolution filters. In the reconstruction stage, RK-CCSNet establishes a novel Learned Runge-Kutta Block (LRKB) based on the famous Runge-Kutta methods, reformulating the process of image reconstruction as a discrete dynamical system. Finally, the implementation of RK-CCSNet achieves state-of-the-art performance on influential benchmarks with respect to prestigious baselines, and all the codes are available at https://github.com/rkteddy/RK-CCSNet.

Keywords: Compressed Sensing; Convolutional Sensing; Runge-Kutta Methods

1 Introduction

Compressed Sensing (CS) [5] is a prominent technique that combines sensing and compression together at the hardware level, and can ensure high-fidelity

* This research is supported by National Natural Science Foundation of China grant No.61772232.
** The corresponding author: Qingliang Chen (tpchen@jnu.edu.cn).

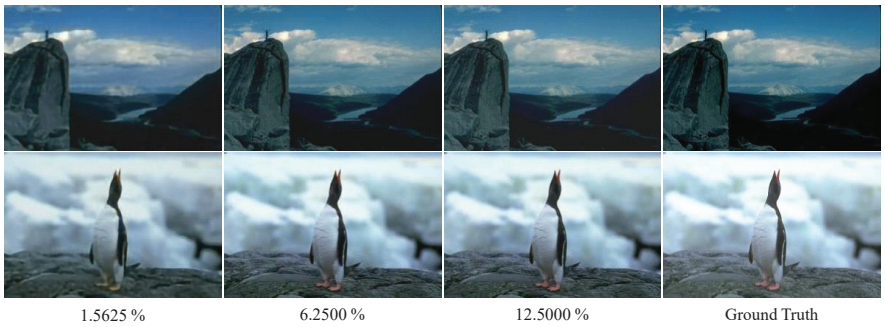

Fig. 1. This figure shows the test results of the proposed RK-CCSNet on BSDS100 [1] in different sampling ratios. Note that almost perfect visual effect is achieved when the sampling ratio is 6.2500%, which implies that our model is capable of reconstructing high-quality images even in low sampling ratios

reconstruction from limited observations received. In the CS framework, signals are acquired by linear projection, which is proved to have the ability to preserve most of the features in a few measurements if the sensing matrices satisfy the *Restricted Isometry Property (RIP)* [3]. Compared with Nyquist's theory, this method uses the sparse nature of the signal to restore the almost perfect original one from a much smaller number of measurements, leading to large reduction in the cost of sensing, storing and transmitting. Several applications such as Single Pixel Camera (SPC) [7], Hyperspectral Compressive Imaging (HCI) [2], Compressive Spectral Imaging System [9], High-Speed Video Camera [13], and CS Magnetic Resonance Imaging (MRI) system [21] have been introduced and implemented. Taking SPC as an example, it uses only a number of single-pixel signals in each shot, and merely a few shots are integrated to reconstruct the original image in the receiving end. Therefore, before decompressing images, the amount of signals needed is much smaller and thus is conducive to long-distance transmission.

Over the years, a great deal of CS algorithms have been proposed such as Orthogonal Matching Pursuit (OMP) [31], Basis Pursuit (BP) [4] and Total Variance minimization by Augmented Lagrangian and ALternating direction ALgorithms (TVAL3) [19]. For instance, Zhang et al. [34] proposed a Group Sparse Representation (GSR) method to enhance both image sparseness and non-local self-similarity. But the common weaknesses of them are that they all demand high computational overhead and perform poorly at low sampling ratios (especially when the measurement is lower than 10%). With the rapid development of deep learning, researchers were inspired to use new end-to-end models to develop algorithms in CS, called Deep Compressed Sensing (DCS). These algorithms do not use the prior knowledge of any signal, but are fed a large number of training data for neural networks instead. The linear sensing module and reconstruction module form an Auto-Encoder structure [25]. Through end-

to-end training, both sensing module and reconstruction module can be jointly optimized. Pure data-driven optimization learns how to make the best of the data structure to speed up the reconstruction process.

There are two challenges in DCS: the linear encoding and the non-linear reconstruction, respectively. For the former one, traditional CS algorithms usually apply hand-designed models according to the nature of the data. However, general DCS models treat the sparse transformation with a fully connected layer, which contains no priors and thus is hard to learn a concise embedding. Since convolution can be an efficient prior that can well describe the structural features of images, and can be easily combined with DCS, we replace the fully connected layer with continuous convolutions (for linear observations, there are no activation after every convolutional layer), named Sequential Convolutional Module (SCM). For the latter one, the main approaches are to develop a powerful reconstruction module with elaborate structures. According to the recent studies on the relationship between ODE [26] and ResNet [11], the conventional residual architecture with simple skip connections can be seen as an approximation of the forward Euler method [26], a simple numerical method. Accordingly, we introduce a novel architecture called Learned Runge-Kutta Block (LRKB) originating from Runge-Kutta methods [26], the higher-order numerical schemes than the forward Euler method.

The main contributions of the paper are summarized as follows:

1. We propose a SCM for image CS, which applies local connectivity priors during the sensing stage. SCM is empirically proved to have the ability to preserve spatial features and thus avoid block artifacts and high frequency noise in the final reconstruction.
2. We further develop a novel LRKB to achieve higher reconstruction quality, by reformulating the process of image reconstruction as a discrete dynamical system. Hence we can adopt highly efficient algorithms from ODE such as Runge-Kutta methods [26], which can offer higher order of accuracy for numerical solutions.
3. An end-to-end Runge-Kutta Convolutional Compressed Sensing Network (RK-CCSNet) is introduced to encapsulate the two modules above, resulting in a novel end-to-end structure. And the implementation of RK-CCSNet are extensively evaluated on influential benchmarks, achieving state-of-the-art performance with respect to prestigious baselines.

The paper is structured as follows. We will present preliminaries in the next section, followed by the section to detail the proposed RK-CCSNet, and then comes the section for empirical and comparative studies on different benchmarks compared with influential models. And we will conclude the paper in the last section.

2 Preliminaries

2.1 Compressed Sensing

CS [5] is a signal acquisition and manipulation paradigm consisting of sensing and compressing simultaneously, which leads to significant reduction in computational cost. Given a high-dimensional signal $\mathbf{x} \in \mathbb{R}^N$, the compressive measurement $\mathbf{y} \in \mathbb{R}^M$ about $\mathbf{x}$ can be obtained by $\mathbf{y} = \mathbf{\Phi x}$, where $\mathbf{\Phi} \in \mathbb{R}^{M \times N}$ $(M \ll N)$ denotes the sensing matrix. The aim of CS is to reconstruct the original signal $\mathbf{x}$ from a much lower dimensional measurement $\mathbf{y}$.

2.2 Data-driven Methods for Image Compressed Sensing

Inspired by the great success of DNN in representation learning, Mousavi et al. [24] designed a new measurement and signal reconstruction framework. Stacked Denoising Autoencoders (SDA) is used as an self-supervised feature learner in the reconstruction network to obtain the statistical correlation between different elements of signals and improve the performance of signal reconstruction. And Kulkarni et al. introduced ReconNet [17], which takes image reconstruction as a task similar to super-resolution, with Convolutional Neural Networks (CNN) to carry out pixel-wise mapping. Later, Mousavi et al. [22] argued that the real-world data is not completely sparse on a fixed basis, and moreover the traditional reconstruction algorithms take a lot of time to converge. And they proposed DeepInverse that utilizes Fully Convolutional Networks (FCN) [27] to recover the original image, which is able to learn a structured representation from training data. And Yao et al. [33] presented DR2Net that applied residual architecture to further improve the reconstruction quality. And Xu et al. [32] used multiple stages of reconstructive adversarial networks through Laplacian pyramid architecture to achieve high-quality image reconstruction. And Shi et al. put forward CSNet [30] and CSNet+ [29] to further improve the reconstruction quality. Most recently, Shi et al. [28] tried to solve the problem that different models should be trained in different sampling ratios by introducing a Scalable Convolutional Neural Network (SCSNet). Parallel convolutions were applied in sensing stage [23] to avoid block-based sensing for better adaptability to different signals like Fourier signals. However, these methods do not make any assumptions about the data (i.e., the natural images), which is very essential to obtain low dimensional embeddings for a specific type of data. And recently it was proposed to use convolution as measurement matrix in [6], in which there is only one convolutional layer, not enough to capture the hierarchical structures.

2.3 Residual Neural Network

The Residual Neural Network (ResNet) was first presented in [11], which introduced the *identity skip connection* that allows data to flow directly to subsequent layers, bypassing residual layers. Generally, a residual block can be written as: $y_{n+1} = y_n + F(y_n)$. Skip connection brings shortcut into neural networks, which

propagates the gradients in a more efficient way, making it possible to build a much deeper neural network without gradient vanishing, and thus can obtain impressive performance in many image tasks. ResNet and its variants [15] have been widely used in different applications besides computer vision.

2.4 ResNet and ODEs

Taking x as the time variable, a first-order dynamical system has the form [18]: $y'(x) = F(x, y(x))$ and $y'(x) = y' = \frac{dy}{dx}$ where y is a dependent variable of the changing system state. This ODE describes the process of a system change, in which the rate of change is a function of current time x and system state y. When the initial value satisfies: $y(x_0) = y_0$, this is called the Initial Value Problem (IVP) [18]. Euler method [18] is a first-order numerical method for IVP, including forward Euler method, backward Euler method and improved Euler method. Forward Euler method approximates the system change by truncating Taylor series and integral as: $y_{n+1} = y_n + hF(x_n, y_n)$ and $h = x_{n+1} - x_n$, which has the similar form to a basic block of ResNet. Over the past few years, this link between residual connection and ODEs has been widely discussed by some literature [8, 20]. It leads to a novel perspective that the neural network can be reformulated as a discrete sequence of a time-dependent dynamical system, providing good theoretical guidance for the design of neural network architectures. And conventional residual architectures have been used in many DCS models and have gained substantial effects [33, 29, 28].

As forward Euler method is just the first-order numerical solution of ODEs, we can naturally think of building a more accurate neural network with higher-order numerical approaches such as Runge-Kutta methods [26]. This motivates us to build a residual architecture with LRKB, to achieve higher precision for image reconstruction.

3 The Proposed Model

3.1 Sequential Convolutional Module

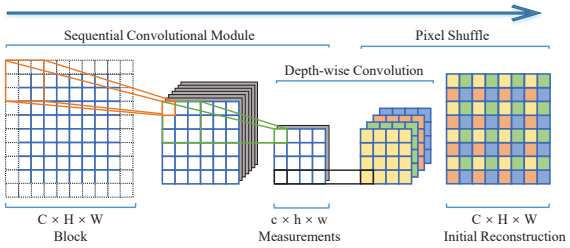

Fig. 2. Sequential Convolutional Module (SCM)

Conventional sensing modules consist of a single fully connected layer to replace the sensing matrix which projects the original image into a measurement of much lower dimension linearly. Here, instead of standard sensing strategies, we propose the SCM, which is also a valid linear operation for CS because convolution can be represented by matrix-matrix multiplication. For a given single channel image $\mathcal{I} \in \mathbb{R}^{1\times H\times W}$, the convolution operations squeeze the image into the shape of $c^2 \times \frac{H}{cr} \times \frac{W}{cr}$, where r^2 is the compression ratio and c^2 is the hyperparameter, both of which depend on the configuration of convolution filters. Then a depth-wise convolution layer expanding the feature channels follows and the shape becomes $c^2r^2 \times \frac{H}{cr} \times \frac{W}{cr}$. Finally, the pixel-shuffle layer will rearrange the elements of $c^2r^2 \times \frac{H}{cr} \times \frac{W}{cr}$ tensor to form a $1 \times H \times W$ tensor, illustrated in Fig. 2.

SCM senses the original image by gradually compacting the image size through a sequence of filters. Compared with conventional sensing strategies, which sense the image block by block through a single shared weight matrix multiplication, our method has the advantage to preserve the spatial features thanks to the sparse local connectivity nature of convolution operations. Moreover, continuous convolution can effectively capture the hierarchical structures in the image. And it can be seen in the following section for experimental studies that SCM is justified to have the ability to eliminate noises introduced by long distance high-frequency component in the block and avoid block artifacts.

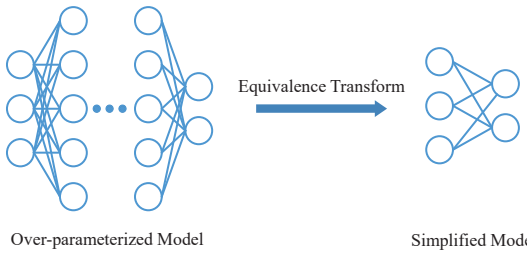

Fig. 3. Simplifying Linear Over-parameterized Model.

The number of feature channels of intermediate layers of SCM during training can be relatively large, as long as the final output shape can meet the required measurements. Since there are no activation functions and no biases, no matter how wide the SCM is, these linear combinations can be finally squeezed into one matrix multiplication as shown in Fig. 3.

To be more specific, we take the feed forward network as an example. Assume that the network input is an n_0-elements vector x, the l^{th} layer contains n_l hidden cells and the i^{th} hidden cell in l^{th} layer is denoted as $h_{l,i}$, the weight in l^{th} layer connecting $h_{l-1,i}$ and $h_{l,j}$ is $w_{l,i,j}$. Then the feed forward network can be modeled by $h_{l,i} = \sum_{j=0}^{n_l} w_{l,i,j} h_{l,j}$.

Every subsequent hidden cell can be represented as a linear combination of x as follows:

$$\begin{aligned} h_{l,i} &= \sum_{j_l=0}^{n_l} w_{l,i,j} \sum_{j_{l-1}=0}^{n_{l-1}} w_{l-1,i,j} \cdots \sum_{j_0=0}^{n_0} w_{0,i,j} x_j \\ &= \sum_{j_l=0}^{n_l} \sum_{j_{l-1}=0}^{n_{l-1}} \cdots \sum_{j_0=0}^{n_0} w_{0,i,j} w_{1,i,j} \dots w_{l,i,j} x_j \\ &= \sum_{j=0}^{n_0} W_{i,j} x_j. \end{aligned}$$

This indicates that the final output of the network y can also be represented as a linear combination of the input x. Thus we can utilize the learning ability of an over-parameterized model to converge to a better optimal point. However, wider model is more likely to cause unstable gradient problems during training. So it is a trade-off to choose a proper width.

Note that SCM still pertains to block-based sensing, but it applies local connectivity priors during training time. It is the same in deployment as block-based methods since all the convolution kernels can be transformed into one matrix during test time. So, SCM just changes the training behavior, leading to better performance on natural images.

3.2 Learned Runge-Kutta Block

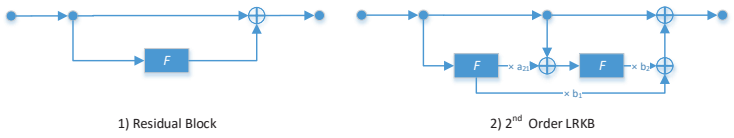

Fig. 4. Comparison of a Residual Block and a Learned Runge-Kutta Block.

The gradual reconstruction of the image can be reformulated as a dynamical system, where the initial condition is the measurements and the ideal termination condition is the original image. In such a dynamical system, each CNN block is a state transition, training data are fed to learn the mapping from low dimension measurements to the original image. Thus, we are able to see the residual block with a single skip connection as a forward Euler method [26], which is just a first-order scheme. So by mimicking higher order numerical methods, we can expect higher accuracy. Hence, we consider Runge-Kutta methods, which is a family of high-precision single step algorithms for numerical solution of ODE, to build a novel residual architecture with better performance.

Specifically, second order Runge-Kutta method takes the following form:

$$y_{n+1} = y_n + b_1 K_1 + b_2 K_2, \tag{1}$$

$$K_1 = hF(x_n, y_n), \tag{2}$$

$$K_2 = hF(x_n + c_2 h, y_n + a_{21} K_1), \tag{3}$$

where a_{21}, b_1, b_2 and c_2 are the coefficients. To specify the exact values of the coefficients, we expand K_2 at (x_n, y_n) according to Taylor's formula:

$$\begin{aligned} & hF(x_n + c_2 h, y_n + a_{21} K_1) \\ = \; & h\left[F(x_n, y_n) + c_2 h {F'}_x + a_{21} K_1 {F'}_y + O(h^2)\right] \\ = \; & h\left[F(x_n, y_n) + c_2 h {F'}_x + a_{21} h F {F'}_y + O(h^3)\right], \end{aligned} \tag{4}$$

where F denotes $F(x_n, y_n)$ and ${F'}_x$, ${F'}_y$ are the partial derivatives of F with respect to x and y, respectively. Then we get:

$$\begin{aligned} y_{n+1} &= y_n + (b_1 K_1 + b_2 K_2) \\ &= y(x_n) + b_1 h F(x_n, y_n) + b_2 h\left[F(x_n, y_n) + c_2 h {F'}_x + a_{21} h F {F'}_y\right] + O(h^3) \\ &= y(x_n) + (b_1 + b_2) h F(x_n, y_n) + c_2 b_2 h^2 {F'}_x + a_{21} b_2 h^2 F {F'}_y + O(h^3). \end{aligned} \tag{5}$$

And we expand $y(x_{n+1})$ at x_n:

$$\begin{aligned} y(x_{n+1}) &= y(x_n) + h y'(x_n) + \frac{h^2}{2!} y''(x_n) + O(h^3) \\ &= y(x_n) + h F(x_n, y_n) + \frac{h^2}{2!}\left[{F'}_x + F {F'}_y\right] + O(h^3). \end{aligned} \tag{6}$$

Let $y(x_{n+1}) = y_{n+1}$, we get:

$$b_2 + b_1 = 1, \; b_2 c_2 = \frac{1}{2}, \; b_2 a_{21} = \frac{1}{2}, \tag{7}$$

which is an under-determined system of equation, all methods satisfying the above forms are collectively referred to as Second-Order Runge-Kutta Method. As we can see, b_2 can be the only free variable, and can be jointly optimized during training time.

Actually the neural network can be trained to predict the auxiliary variable of $\alpha = \log(-\log(b_2))$, to avoid division by zero. Also, when regressing the unconstrained value of α, b_2 is resolved to the value between 0 and 1. Hence we can have: $b_2 = e^{-e^{\alpha}}$, $b_1 = 1 - e^{-e^{\alpha}}$, and $a_{21} = \frac{e^{e^{\alpha}}}{2}$.

Regarding each non-linear state transition function F as an independent CNN block, we build a residual block as shown in Fig. 4, where the state transition functions F is illustrated in Fig. 5. Moreover, we use PReLU [10] as the activation function and adopt pre-activation structure [12], where the two convolution filters share the same weights.

3.3 The Overall Structure

The overall structure of our model is an end-to-end auto-encoder structure as shown in Fig. 6, where the encoder is a sequence of sub-sampling convolutional

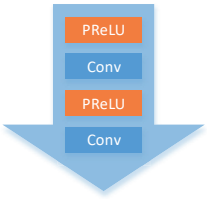

Fig. 5. The State Transition Function F.

sensing filters without activation functions, producing measurements. A followed depth-wise convolution layer expands the feature channels and the resulting feature map is to be rearranged to match the original size by a pixel shuffle layer, whose product is called initial reconstruction. Then the output of encoder is to be fed to the subsequent reconstruction network consisting of a head, body and tail. The head first converts the initial reconstruction to image features by convolution block, followed by a ReLU function. Afterwards the feature maps are further processed by the body consisting of several LRKBs. Then the tail will turn the resulting feature maps back to the final reconstructed image.

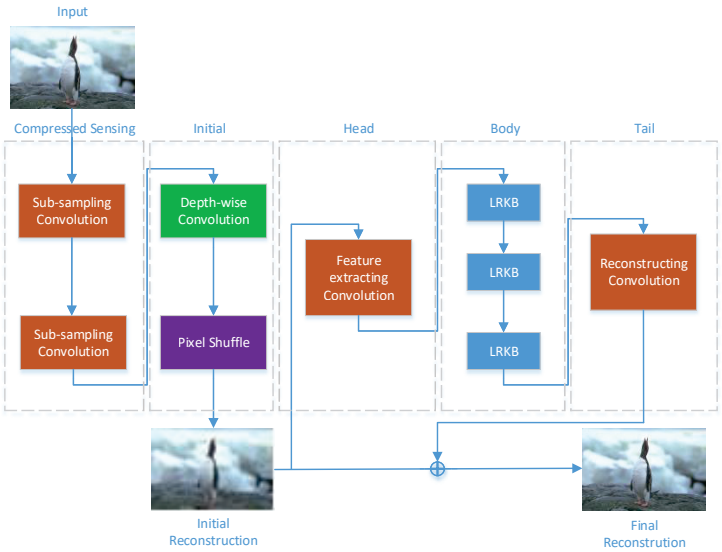

Fig. 6. The Overall Structure of RK-CCSNet.

4 Experimental Studies

4.1 Weights Initialization

Because of the introduction of sequential convolution filters without activation functions, it was observed that the gradient is unstable during training process. By comparative empirical studies, we have identified the source of the problem in weights initialization.

To be more specific, in each convolution step, we define $\mathcal{X} \in \mathbb{R}^{C\times H\times W}$ to be the input matrix convoluted with a filter $\mathcal{F} \in \mathbb{R}^{O\times C\times H_f\times W_f}$, and $\mathcal{Y} \in \mathbb{R}^{O\times H\times W}$ to be the output matrix, then each element $\mathcal{Y}_{ij}$ in the output matrix $\mathcal{Y}$ is defined as:

$$\mathcal{Y}_{k,i,j} = \sum_{n=0}^{C}\sum_{h=0}^{H_f}\sum_{w=0}^{W_f} \mathcal{X}_{n,i+h-\frac{(H_f-1)}{2},j+w-\frac{(W_f-1)}{2}}\mathcal{F}_{k,n,h,w} \tag{8}$$

where we simply assume that stride equals 1 with the same padding strategy, and the height and width of the filter to be odd number, which can be extended to more general situations. As we can see, each output element is the summation of CH_fW_f products of $\mathcal{X}$ and $\mathcal{F}$. We assume that $\mathcal{X}$ is normalized such that $\mathcal{X} \sim N(0,1)$ and we initialize the weight matrix of $\mathcal{F}$ with normal distribution without considering the shape of filter, let's say $\mathcal{F} \sim N(0,1)$, then after convolution we will get $\mathcal{Y} \sim N(0, \sqrt{CH_fW_f})$. If CH_fW_f is greater than 1 (which is surely the case), after a sequence of convolution steps, the elements in the resulting matrix will grow dramatically, leading to gradient explosion. The similar situation will cause gradient vanishing when we initialize the weights with normal distribution of which standard deviation is too small. To address this problem, we simply initialize the weight matrix with scaled normal distribution as:

$$\mathcal{F} \sim N(0, \frac{1}{\sqrt{CH_fW_f}}). \tag{9}$$

To illustrate the effect of our initialization method, we build a toy example which took a tensor of shape $(8, 64, 96, 96)$ as the input, and is sequentially convoluted by 10 filters of shape $(64, 64, 3, 3)$ with stride of 1 and some padding strategy to keep the shape of the input tensor and the output tensor remain unchanged. We initialize the weights with three different distribution: $\mathcal{F}_1 \sim N(0,1)$, $\mathcal{F}_2 \sim N(0, 0.01)$ and $\mathcal{F}_3 \sim N(0, \frac{1}{\sqrt{CH_fW_f}})$. And Fig. 7 shows the changing standard deviation of the output tensor in each convolution stage.

The figure clearly shows that the general weights initialization method is not suitable for continuous convolution operations without activation functions, which will lead to either gradient explosion or gradient vanishing. And this example verifies that the scaled version of $\mathcal{F}$ remains very stable and thus can have better performance and generalization.

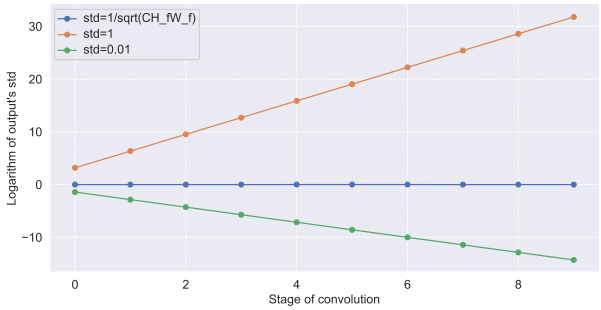

Fig. 7. The comparison of weights initialization with different standard deviations.

4.2 Datasets and Implementation Details

To compare with state-of-the-art deep learning based models, we trained the models on the training set and test set of BSDS500[4], with 400 images for training and 100 images for testing (BSDS100). As the original images are either 321×481 or 481×321, we randomly crop the images into patches of 96×96 and randomly flip horizontally for data augmentation. In addition, we also compare our method with TVAL3 [19] and GSR [34] on Set5 and Set14[5], which contains 5 images and 14 images, respectively. Because those images are not shape consistent, we resize them into $(256, 256)$ for evaluation. All the images are first converted to YCbCr color space and only the Y channel is used as the input of all the models. We use Adam optimizer [16] for training and set the exponential decay rates to 0.9 and 0.999 for the first and second moment estimate. The batch size is set to 4 and both CSNet+ [29] and RK-CCSNet were trained for 200 epochs at all, with the initial learning rate of $1e-3$ and decay of 0.25 at 60, 90, 120, 150 and 180 epochs respectively. The sampling ratio for testing was set from 1/64 to 1/2, i.e., 1.5625%, 3.1250%, 6.2500%, 12.5000%, 25.0000%, and 50.0000%. PSNR (Peak Signal-to-Noise Ratio) and SSIM (Structural SIMilarity) [14] are chosen as the evaluation metrics throughout our experiments.

4.3 Experimental Results

Table 1 presents the test results of CSNet+ [29] and RK-CCSNet on BSDS100 with the corresponding PSNR and SSIM, and the best results are marked in bold font. It can be seen that our model exhibits significantly better performance compared with CSNet+ across all sampling ratios. In average, our model gains 2.14% and 1.72% improvements in PSNR and SSIM, respectively.

Further experimental results of our model on Set5 and Set14 compared with TVAL3, GSR and CSNet+ are provided in Table 2. Our model outperforms all the other ones across all different datasets, exhibiting excellent generalization

[4] https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/resources.html#bsds500
[5] http://vllab.ucmerced.edu/wlai24/LapSRN/

Table 1. Comparisons of CSNet+ and RK-CCSNet on BSDS100

		CSNet+		RK-CCSNet (our)	
Data	ratio	PSNR	SSIM	PSNR	SSIM
BSDS100	1.56%	25.01	0.6904	**25.56**	**0.7055**
	3.12%	26.55	0.7413	**26.99**	**0.7564**
	6.25%	28.14	0.7977	**28.60**	**0.8133**
	12.5%	30.11	0.8602	**30.56**	**0.8759**
	25.0%	32.81	0.9206	**33.43**	**0.9335**
	50.0%	36.62	0.9659	**37.92**	**0.9766**
Average		29.87	0.8294	**30.51**	**0.8437**

Table 2. Comparisons of different CS algorithms on Set5 and Set14

		TVAL3		GSR		CSNet+		RK-CCSNet (our)	
Data	ratio	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Set5	1.5625%	19.00	0.4844	21.39	0.5815	25.02	0.6888	**25.63**	**0.7186**
	3.1250%	19.89	0.5415	23.70	0.6822	27.42	0.7778	**28.03**	**0.8142**
	6.2500%	22.03	0.6175	27.59	0.8163	30.11	0.8605	**30.91**	**0.8867**
	12.5000%	23.75	0.7365	31.61	0.9016	33.57	0.9250	**35.05**	**0.9461**
	25.0000%	27.39	0.8522	36.32	0.9510	37.94	0.9665	**39.29**	**0.9758**
	50.0000%	33.11	0.9430	42.18	0.9908	42.70	0.9856	**44.72**	**0.9913**
Set14	1.5625%	16.79	0.3993	18.93	0.4399	23.13	0.5768	**23.32**	**0.5933**
	3.1250%	18.40	0.4514	20.26	0.5184	25.03	0.6660	**25.42**	**0.6968**
	6.2500%	19.65	0.5287	23.59	0.6526	27.25	0.7651	**27.48**	**0.7897**
	12.5000%	21.03	0.6379	28.08	0.7915	30.16	0.8630	**30.93**	**0.8880**
	25.0000%	22.69	0.7731	31.82	0.8939	33.92	0.9354	**35.03**	**0.9505**
	50.0000%	26.61	0.9004	37.47	0.9619	38.67	0.9756	**40.66**	**0.9848**
Average		22.53	0.6555	28.57	0.7650	31.24	0.8322	**32.21**	**0.8530**

and achieving state-of-the-art results. We selected some representative images to demonstrate visual comparisons of each model, in Fig. 8 and 9. It can be seen that neither TVAL3 nor GSR can reconstruct meaningful features from sample images in extremely low ratios. CSNet+ can roughly restore the original image but results in serious blocking artifacts, while RK-CCSNet produces much smoother boundary between blocks. Moreover, RK-CCSNet has less luminance loss compared with CSNet+. When the sampling ratio comes to 12.5%, these models all perform well. However, the one reconstructed by TVAL3 has a lot of noises. GSR does a bit better in visual but brings about distortions. CSNet's reconstruction performs poorly in details of the image. We also found that in the case of block by block sensing, if the block contains high-frequency components, the noise will be distributed across all parts of the reconstructed block, causing the whole reconstructed block less smooth. And this difference between blocks exacerbates blocking artifacts. However, RK-CCSNet with SCM as the sensing module, will not lead to this phenomenon, which is thus significantly better in the reconstruction of high frequency details of the image. All in all, RK-CCSNet has the highest reconstruction quality among all the models.

4.4 Ablation Studies

Ablation studies are further carried out to justify the efficacy of the two modules proposed in our model. In general, we divide a CS model into two submodules: sensing module and reconstruction module. We replace different mod-

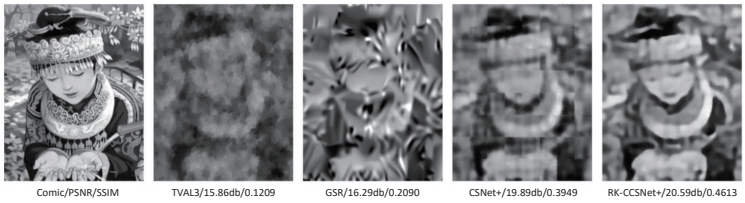

Fig. 8. Visual comparisons of the reconstructed image in sampling ratio of 1.5625%

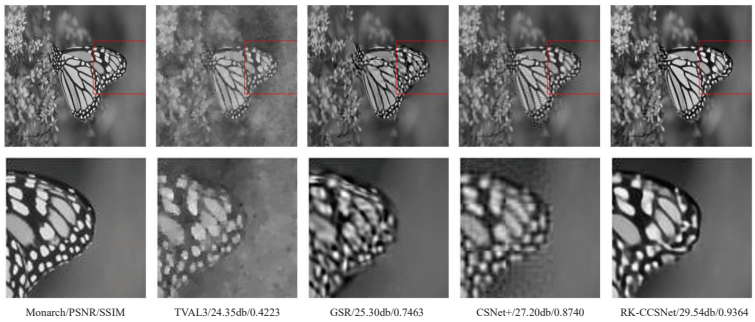

Fig. 9. Visual comparisons of the reconstructed image in sampling ratio of 12.5000%

ules of CSNet+ to form different models. To be more specific, the models compared are listed as follows: Baseline (CSNet+), Baseline with SCM, Baseline with LRKB, and RK-CCSNet. The experimental results are shown in Table 3. It can be seen that both proposed modules can lead to appreciable improvements over the baseline model. LRKB has more non-linear reconstruction strength for the global structure of the image when the observation rate is limited, since LRKB is from ODE theory with higher order of accuracy for numerical analysis than the one in the baseline with a conventional residual architecture. SCM can restore more details and eliminate most noises when the observation rate is sufficient. And SCM's power of preserving spatial features grows with the increasing of sampling ratios, because the larger spatial shape of the measurement will contain more spatial information, while the standard fully connected layer cannot capture spatial features well. Moreover, SCM's local sensing strategy can also avoid introducing noises.

Table 3. Ablation results on BSDS100

		1.5625%		3.1250%		6.2500%		12.5000%		25.0000%		50.0000%	
SCM	LRKB	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
		25.01	0.6904	26.55	0.7413	28.14	0.7977	30.11	0.8602	32.81	0.9206	36.62	0.9659
✓		25.31	0.7014	26.64	0.7488	28.25	0.8075	30.26	0.8710	33.05	0.9294	37.59	0.9753
	✓	25.49	0.7010	26.91	0.7499	28.48	0.8055	30.11	0.8584	32.43	0.9138	36.91	0.9670
✓	✓	**25.56**	**0.7055**	**26.99**	**0.7564**	**28.60**	**0.8133**	**30.56**	**0.8759**	**33.43**	**0.9335**	**37.92**	**0.9766**

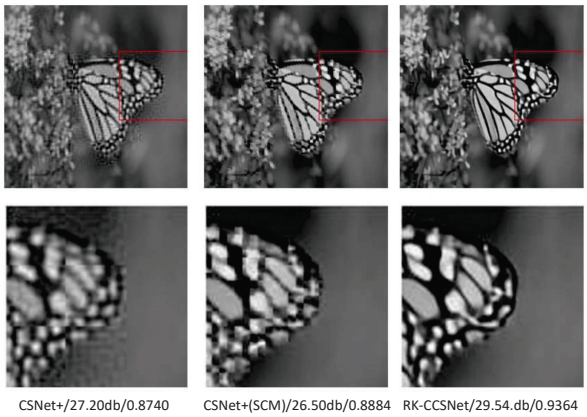

Fig. 10. Visual comparisons of reconstructed image in the sampling ratio of 12.5%

To further present the effect of SCM and LRKB, we compare the visual quality of the reconstructed image by three different models in Fig. 10. It can be seen that our proposed SCM can eliminate most noises inside the block caused by high frequency components as mentioned above, and alleviate luminance loss. Combined with LRKB's powerful reconstruction strength, our model can restore images to a higher level.

5 Conclusion

In this paper, we have proposed a sensing module and a reconstruction module respectively to enhance DCS frameworks. In the sensing stage, the proposed SCM applies continuous convolution operations to replace the conventional single matrix multiplication to preserve spatial features. In the reconstruction module of the proposed LRKB, we reformulate the forward process of ResNet as a discrete dynamical system and introduce a novel residual architecture inspired by Runge-Kutta methods, which can lead to much more precise reconstructions. Furthermore, we have introduced an end-to-end RK-CCSNet to encapsulate the two modules above. The implementation of RK-CCSNet has outperformed other prestigious baselines when extensively evaluated on influential benchmarks. In addition, ablation studies are also carried out that have justified the efficacy of the two modules individually.

References

1. Arbelaez, P., Maire, M., Fowlkes, C.C., Malik, J.: Contour detection and hierarchical image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence **33**(5), 898–916 (2011)

2. August, Y., Vachman, C., Rivenson, Y., Stern, A.: Compressive hyperspectral imaging by random separable projections in both the spatial and the spectral domains. Applied optics **52**(10), D46–D54 (2013)
3. Candès, E.J.: The restricted isometry property and its implications for compressed sensing. Comptes rendus mathematique **346**(9-10), 589–592 (2008)
4. Chen, S.S., Donoho, D.L., Saunders, M.A.: Atomic decomposition by basis pursuit. SIAM review **43**(1), 129–159 (2001)
5. Donoho, D.L., et al.: Compressed sensing. IEEE Transactions on Information Theory **52**(4), 1289–1306 (2006)
6. Du, J., Xie, X., Wang, C., Shi, G., Xu, X., Wang, Y.: Fully convolutional measurement network for compressive sensing image reconstruction. Neurocomputing **328**, 105–112 (2019)
7. Duarte, M.F., Davenport, M.A., Takhar, D., Laska, J.N., Sun, T., Kelly, K.F., Baraniuk, R.G.: Single-pixel imaging via compressive sampling. IEEE Signal Processing Magazine **25**(2), 83–91 (2008)
8. E, W.: A proposal on machine learning via dynamical systems. Communications in Mathematics and Statistics **5**, 1–11 (02 2017). https://doi.org/10.1007/s40304-017-0103-z
9. Gehm, M., John, R., Brady, D., Willett, R., Schulz, T.: Single-shot compressive spectral imaging with a dual-disperser architecture. Optics express **15**(21), 14013–14027 (2007)
10. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: 2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015. pp. 1026–1034. IEEE Computer Society (2015)
11. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. pp. 770–778. IEEE Computer Society (2016)
12. He, K., Zhang, X., Ren, S., Sun, J.: Identity mappings in deep residual networks. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part IV. Lecture Notes in Computer Science, vol. 9908, pp. 630–645. Springer (2016)
13. Hitomi, Y., Gu, J., Gupta, M., Mitsunaga, T., Nayar, S.K.: Video from a single coded exposure photograph using a learned over-complete dictionary. In: Metaxas, D.N., Quan, L., Sanfeliu, A., Gool, L.V. (eds.) IEEE International Conference on Computer Vision, ICCV 2011, Barcelona, Spain, November 6-13, 2011. pp. 287–294. IEEE Computer Society (2011)
14. Horé, A., Ziou, D.: Image quality metrics: PSNR vs. SSIM. In: 20th International Conference on Pattern Recognition, ICPR 2010, Istanbul, Turkey, 23-26 August 2010. pp. 2366–2369 (2010)
15. Huang, G., Liu, Z., van der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. pp. 2261–2269. IEEE Computer Society (2017)
16. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings (2015)

17. Kulkarni, K., Lohit, S., Turaga, P.K., Kerviche, R., Ashok, A.: Reconnet: Non-iterative reconstruction of images from compressively sensed measurements. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. pp. 449–458. IEEE Computer Society (2016)
18. Lambert, J.D.: Numerical Methods for Ordinary Differential Systems: The Initial Value Problem. John Wiley & Sons, Inc., USA (1991)
19. Li, C.: An efficient algorithm for total variation regularization with applications to the single pixel camera and compressive sensing. Master's thesis, Rice University (2010)
20. Lu, Y., Zhong, A., Li, Q., Dong, B.: Beyond finite layer neural networks: Bridging deep architectures and numerical differential equations. In: Dy, J.G., Krause, A. (eds.) Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018. Proceedings of Machine Learning Research, vol. 80, pp. 3282–3291. PMLR (2018)
21. Lustig, M., Donoho, D.L., Santos, J.M., Pauly, J.M.: Compressed sensing MRI. IEEE Signal Processing Magazine **25**(2), 72 (2008)
22. Mousavi, A., Baraniuk, R.G.: Learning to invert: Signal recovery via deep convolutional networks. In: 2017 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2017, New Orleans, LA, USA, March 5-9, 2017. pp. 2272–2276. IEEE (2017)
23. Mousavi, A., Dasarathy, G., Baraniuk, R.G.: A data-driven and distributed approach to sparse signal representation and recovery. In: International Conference on Learning Representations (2018)
24. Mousavi, A., Patel, A.B., Baraniuk, R.G.: A deep learning approach to structured signal recovery. In: 53rd Annual Allerton Conference on Communication, Control, and Computing, Allerton 2015, Allerton Park & Retreat Center, Monticello, IL, USA, September 29 - October 2, 2015. pp. 1336–1343. IEEE (2015)
25. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning Internal Representations by Error Propagation, p. 318–362. MIT Press, Cambridge, MA, USA (1986)
26. Sauer, T.: Numerical Analysis. Addison-Wesley Publishing Company, USA, 2nd edn. (2011)
27. Shelhamer, E., Long, J., Darrell, T.: Fully convolutional networks for semantic segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **39**(4), 640–651 (2017)
28. Shi, W., Jiang, F., Liu, S., Zhao, D.: Scalable convolutional neural network for image compressed sensing. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019. pp. 12290–12299. Computer Vision Foundation / IEEE (2019)
29. Shi, W., Jiang, F., Liu, S., Zhao, D.: Image compressed sensing using convolutional neural network. IEEE Transactions on Image Processing **29**, 375–388 (2020)
30. Shi, W., Jiang, F., Zhang, S., Zhao, D.: Deep networks for compressed image sensing. In: 2017 IEEE International Conference on Multimedia and Expo, ICME 2017, Hong Kong, China, July 10-14, 2017. pp. 877–882. IEEE Computer Society (2017)
31. Tropp, J.A., Gilbert, A.C.: Signal recovery from random measurements via orthogonal matching pursuit. IEEE Transactions on Information Theory **53**(12), 4655–4666 (2007)
32. Xu, K., Zhang, Z., Ren, F.: LAPRAN: A scalable laplacian pyramid reconstructive adversarial network for flexible compressive sensing reconstruction. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) Computer Vision - ECCV 2018 -

15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part X. Lecture Notes in Computer Science, vol. 11214, pp. 491–507. Springer (2018)
33. Yao, H., Dai, F., Zhang, D., Ma, Y., Zhang, S., Zhang, Y.: Dr2-net: Deep residual reconstruction network for image compressive sensing. Neurocomputing **359**, 483–493 (2017)
34. Zhang, J., Zhao, D., Jiang, F., Gao, W.: Structural group sparse representation for image compressive sensing recovery. In: Bilgin, A., Marcellin, M.W., Serra-Sagristà, J., Storer, J.A. (eds.) 2013 Data Compression Conference, DCC 2013, Snowbird, UT, USA, March 20-22, 2013. pp. 331–340. IEEE (2013)