Single MR Image Super-Resolution via Channel Splitting and Serial Fusion Network

Xiaole Zhao, Hangfei Liu, Yulun Zhang, Yun Qin, Tao Zhang, Xueming Zou

***Abstract*—Spatial resolution is a critical imaging parameter in magnetic resonance imaging (MRI). Acquiring high-resolution MRI data usually takes long scanning time and would subject to motion artifacts due to hardware, physical, and physiological limitations. Single image super-resolution (SISR), especially that based on deep learning techniques, is an effective and promising alternative technique to improve the native spatial resolution of magnetic resonance (MR) images. However, the deeper network is more difficult to be effectively trained because the information is gradually weakened as the network deepens. This problem becomes more serious for medical images due to the degradation of training examples. In this paper, we present a novel channel splitting and serial fusion network (CSSFN) for single MR image super-resolution. Specifically, the proposed CSSFN network splits the hierarchical features into a series of subfeatures, which are then integrated together in a serial manner. Thus, the network becomes deeper and can deal with the subfeatures on different channels discriminatively. Moreover, a dense global feature fusion (DGFF) is adopted to integrate the intermediate features, which further promotes the information flow in the network. Extensive experiments on several typical MR images show the superiority of our CSSFN models over other advanced SISR methods.**

***Index Terms*—Convolutional neural network, channel splitting, magnetic resonance imaging, super-resolution, serial fusion.**

I. INTRODUCTION

MAGNETIC resonance imaging (MRI) is an important and widely used tool for diagnosis and image-guided therapeutics. High-resolution (HR) magnetic resonance (MR) images are usually preferred in clinical practice due to more clear image structure and texture details, as well as the benefits to subsequent analysis and processing [1], [2]. However, the acquisition of HR images is constrained by hardware, physical and physiological factors, and increasing the spatial resolution of MR images typically reduces the signal noise ratio (SNR) and/or increases imaging time [3], which further increases the risk of MR images affected by motion artifacts.

X. Zhao is with the School of Life Science and Technology, University of Electronic Science and Technology of China, Chengdu, Sichuan 611731, China (e-mail: zxlation@foxmail.com).

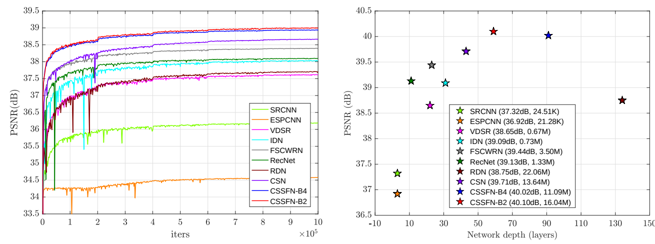
Y. Zhang is now with the Department of Electrical and Computer Engineering, Northeastern University, Boston, MA 02115, USA (e-mail: yulun100@gmail.com).

H. Liu and Y. Qin are with the School of Life Science and Technology, University of Electronic Science and Technology of China, Chengdu, 611731, China (e-mail: lhfuestc@163.com; qyuner@163.com).

T. Zhang is with High Field Magnetic Resonance Brain Imaging Laboratory of Sichuan and Key Laboratory for NeuroInformation of Ministry of Education, Chengdu, Sichuan 611731, China; He is also with the School of Life Science and Technology, University of Electronic Science and Technology of China, Chengdu, Sichuan 611731, China (e-mail: taozhangjin@gmail.com).

X. Zou is with the School of Life Science and Technology, University of Electronic Science and Technology of China, Chengdu, Sichuan 611731, China (mark.zou@alltechmed.com).

Manuscript received XXXX XX, 2019; revised XXXX XX, 2019.



(a) Validation curves (b) Testing results

Fig. 1. Model performance and scale (depth and parameters) comparison between several deep models. The results are evaluated on T2 MR images of IXI dataset for SR×2. "B2" and "B4" imply the number of subfeatures when performing channel splitting. The proposed CSSFN models have better tradeoffs between performance and model scale.

Image super-resolution (SR) is an effective and cost efficient alternative technique to increase the spatial resolution of MR images, which aims at inferring a HR image from one or more low-resolution (LR) images. Up to now, many SR algorithms have been investigated and proposed for both natural images and medical images, e.g., interpolation-based and edge-guided methods [4], [5], [6], [7], modeling and reconstruction based methods [8], [9], [14], [15], example learning based methods [10], [11], and dictionary learning and sparse representation methods [12], [13], [16], [18] etc. However, the performance of these conventional methods is essentially limited because they apply inadequate additional information and models with limited representational capacity to solve the ill-posed inverse problem of image SR tasks [19], [20].

In recent years, deep learning [21] based single image super resolution (SISR) methods have demonstrated great superiority over conventional SR methods. A pioneering work that uses convolutional neural networks (CNNs) [66] to deal with SISR is the super-resolution convolutional neural network (SRCNN) [22], [23]. It implicitly learns an end-to-end mapping function between LR and HR images by utilizing a fully-convolutional network. Subsequently, many more advanced SISR techniques based on deep CNNs were proposed. Some typical examples are DRCN [24], DRRN [25], VDSR [26], MemNet [27], ESPCNN [28], SRResNet [29], EDSR/MDSR [30], RDN [31], CMSCN [34] and RCAN [35] etc. These methods have overwhelming advantages over traditional methods and greatly promote the best state of SISR performance. However, they are mainly aimed at the SISR tasks of natural images, instead of medical images (or more specifically, MR images). Thus, they may be unsuitable for solving medical image SR tasks due to the degradation of training examples [19], although they have excellent performance on natural images.

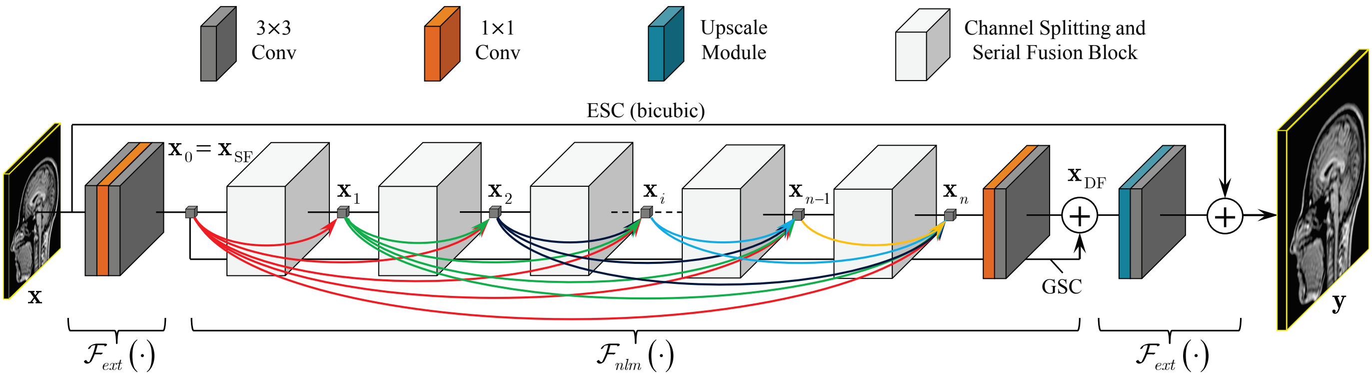


Fig. 2. The overall architecture of the proposed channel splitting and serial fusion network (CSSFN). The symbol "+" indicates element-wise summation between two tensors with the same shape. GSC and ESC denote global skip connection and external skip connection, respectively. The hierarchical features are integrated together in a dense learning manner [55], [56]. We term this as a dense global feature fusion (DGFF).

Some deep learning based methods specializing in the SISR tasks of medical images have also emerged due to the tremendous success of deep learning techniques in computer vision and pattern recognition [36], [37], [38], [39]. These methods utilize relatively shallow network structures to process medical images, e.g., Pham *et al*. [36] presented an algorithm for brain MR images SR according to SRCNN [22]. Despite their extension to 3D cases (named SRCNN3D), the entire network is very shallow and the representational ability of the model is relatively limited, resulting in unsatisfactory SR performance.

The depth of deep CNNs is of crucial importance for the task of image SR [35], and usually defined as the longest path from the input to the output [19], [25], [34]. However, the deeper networks are more difficult to be effectively and fully trained, especially with medical images due to the degeneracy of training samples [19]. Actually, it is verified that the original EDSR model [30] with about 43M model parameters and 70 layers of depth is difficult to be well-trained with 2D proton density (PD) images [19]. Although Zhao *et al*. [40] trained the EDSR model [30] using T1-weighted magnetization-prepared rapid gradient echo (MP-RAGE) images, the reported results are not satisfactory. This is probably because that the EDSR model, which has enormous parameters and very deep network structure, is not adequately well-trained with "good" training samples. In this regard, *whether deeper networks are capable of further contributing to improve the performance of medical image SR and how to construct trainable networks with much deeper structure for medical images remain to be explored.*

A recent work [19] has alleviated the dilemma between the trainability and the performance of deep CNN models for MR image SR to some extent. It presented an effective manner to deepen the network but without significant increase in the number of model parameters, i.e., channel splitting. The model proposed by [19], however, is a kind of multistream structure and the multiple information branches are formed by channel splitting instead of the reuse of preceding features [34], [41]. This multistream structure implies that the information flow in the network is *locally parallel*. In this work, we present a serial information fusion mechanism for channel splitting. The proposed model, which we term as channel splitting and serial fusion network (CSSFN), first splits the hierarchical features into a series of subfeatures and then integrates them together in a serial manner. Different from the CSN [19], the proposed CSSFN network is a single-branch structure. Thus, it can reach a deeper structure (up to 90 layers) than the CSN model with fewer model parameters (Fig.1). On the other hand, channel splitting also allows the proposed CSSFN model to deal with features on different channels discriminatorily. But unlike the CSN model, each subfeature in our framework has different depth in the nonlinear mapping process.

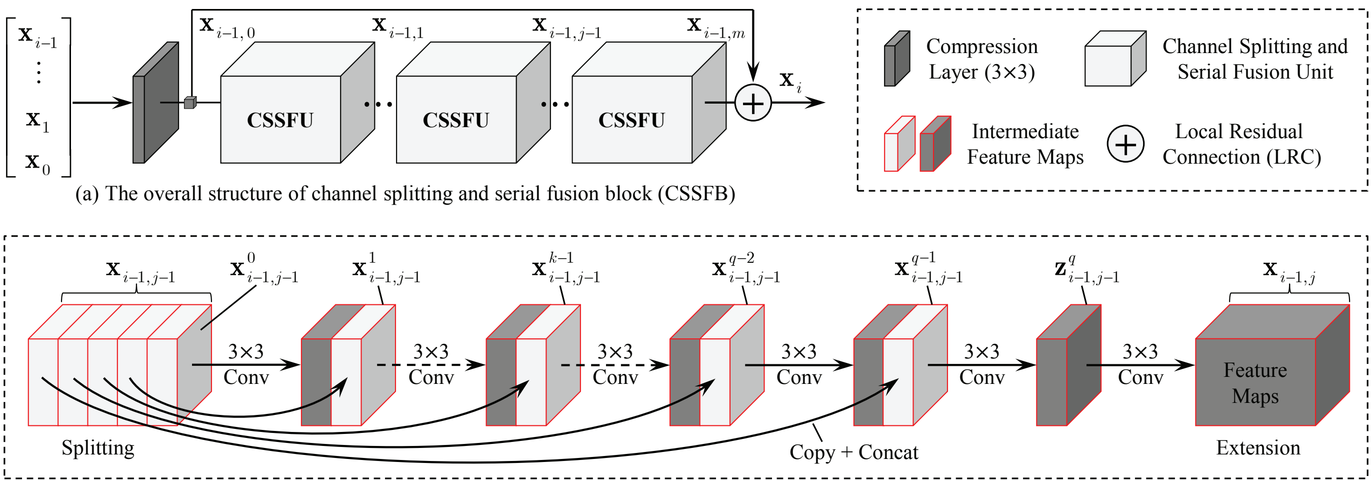
To alleviate the instability of model training caused by the single-branch structure and the increase in model depth, we use a dense global feature fusion (DGFF) strategy to enhance the information flow. The proposed CSSFN model consists of a series of building blocks, each of which contains a battery of channel splitting and serial fusion units (CSSFU) and has one or more inter-block connections to all subsequent building blocks, thus propagating its local features to all successors. Although the DGFF increases the model parameters to some extent, our models still have moderate parameters compared with EDSR [30], RDN [31] and even CSN [19].

In summary, our contributions are as follows: (1) we further improve the tradeoff between the performance and model scale of MR image SR models by introducing a serial local feature fusion (SLFF) approach for channel splitting; (2) we propose a dense global feature fusion (DGFF) and experimentally verify that the combination of DGFF and SLFF is helpful to further improve the performance of the model; (3) we experimentally illustrate that channel splitting should not aggressively increase the number of subfeatures unless there exists a more effective information fusion mechanism. The remainder of this work is organized as follows. We first present some previous work related to the proposed model in section II. Then, the proposed method is illustrated in detail in section III and the experimental results are presented in section IV. Finally, we discuss some related topics in section V and conclude the paper in section VI.

II. Related Work

A. MR Image Super-Resolution

The purpose of MR image SR methods is to overcome the hardware limitations and meet the clinical needs of imaging



(a) The overall structure of channel splitting and serial fusion block (CSSFB)

(b) The structure of the channel splitting and serial fusion unit (CSSFU)

Fig. 3. The overall structure of the basic building block, which consists of m stacked channel splitting and serial fusion units (CSSFU). (a) Each CSSFB also has a short skip connection (SSC) to form local residual learning. (b) The input feature of each CSSFU is split into q subfeatures. Cuboids in light gray imply subfeatures from channel splitting of the input feature, and those in dark gray denote the subfeatures produced by the 3×3 conv layer.

TABLE I
THE STATISTICS OF NETWORK DEPTH AND MODEL PARAMETERS WHEN $c = 256$ AND $n = m = 4$. HERE ic DENOTES THE NUMBER OF INPUT CHANNELS.

Execution	Pure 2D ($ic = 1$)						Pseudo 3D ($ic = 96$)					
Subfeature	$q = 2$			$q = 4$			$q = 2$			$q = 4$		
r	×2	×3	×4	×2	×3	×4	×2	×3	×4	×2	×3	×4
D	59	59	60	91	91	92	59	59	60	91	91	92
P	16.40M	19.35M	18.76M	11.09M	14.04M	13.45M	16.84M	19.79M	19.20M	11.53M	14.48M	13.89M

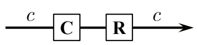
procedures by reconstructing HR images from LR acquisitions using post-processing methods. These SR methods could have strong impacts on structural MRI when focusing on cortical surface or fine-scale structure analysis [36]. The application of SR methods to MR images initially focuses on multiple image super resolution (MISR), e.g., [14], [15], [42]. However, MISR methods usually need calibration and fusion between multiple LR images, which in itself is a very challenging problem that is difficult to achieve with high precision [19].

SISR can avoid the difficulty of calibration and fusion faced by MISR, where only one LR image is required to predicate its HR counterpart, e.g., [16], [18], [43], [44]. A major problem with SISR methods is that there is limited extra information available for HR image reconstruction. Subsequently, some SR methods based on traditional machine learning, e.g., sparse representation [16], [18], example learning [45], [46], as well as compressive sensing [47] etc., have emerged. However, the limited representational capability of these SR methods makes them unable to accurately reflect the highly nonlinear mapping between LR and HR images. Recently, more advanced SISR methods based on deep learning [21] have also been applied to MR image SR tasks [2], [17], [19], [36], [37], [38], [40], which have greatly promoted the performance of SR technologies for medical images or, more specifically, MR images.

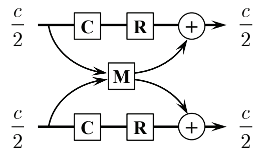
B. Channel Discrimination

The feature maps on different channels of deep CNN models have different types of information and different impacts on the performance of deep models [35], [49], and it is reasonable to deal with the feature mappings discriminatorily. One typical way of channel discrimination is attention mechanism, which is broadly viewed as a tool to bias the allocation of available processing resources towards the most informative components of the input signal [48]. In recent years, it has been introduced to deep neural networks (DNNs) to boost the performance of deep models, such as image generation [50], image captioning [51], [52], image classification [48], [53] and image restoration [35], [54], [49], [57]. These methods have further improved the best state of related fields. For instance, the residual channel attention network (RCAN) [35] pushed the state-of-the-art SR performance forward on natural images, with an extremely deep network structure (over 400 layers).

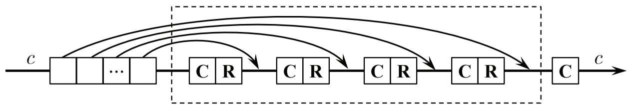
However, few works have been conducted to investigate the effect of channel discrimination for low-level computer vision tasks in medical image processing community (e.g., MR image SR). In this respect, a representative work for single MR image SR is the CSN network [19], where channel discrimination is achieved by channel splitting and the merge-and-run mapping between different branches [41]. This model adopted a *parallel* two-way channel splitting strategy to handle the hierarchical features on different channels, which limited the network depth to some extent. Inspired by channel discrimination mechanism and increasing the network depth, we integrate the subfeatures into a single branch in a serial manner (Fig.3(b)). Thus, the network becomes deeper and thinner (if we constrain the width of subfeatures), which is analogous to stretching a rubber band. Despite the single branch, the serial fusion retains the channel discrimination ability of the network, because the subfeatures have different depths in the processing.



(a) Basic Conv + ReLU

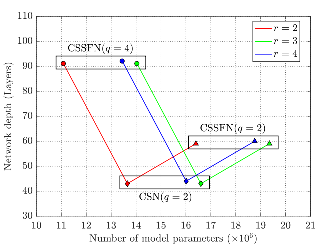


(b) The merge-and-run (MAR)

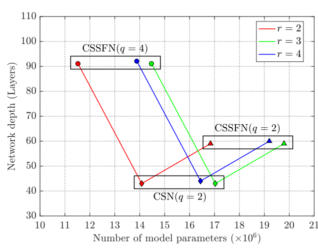


(c) Serial fusion structure

Fig. 4. Three stage mapping structures for comparing branch information fusion (BIF). "C", "R" and "+" represent Conv, ReLU, and skip connection respectively. (a) The basic Conv + ReLU structure. (b) The merge-and-run structure [41] with channel splitting [19]. The number of branches is set to 2 for display purposes, but in the experiments, it is set to 4 for fair comparison. (c) The proposed serial fusion structure. The number of channels in the dashed box can be adjusted accordingly.



(a) Pure 2D execution ($ic = 1$)



(b) Pseudo 3D execution ($ic = 96$)

Fig. 5. Comparison of network depth and model parameters between the CSN [19] and the proposed CSSFN. For all compared models, we set $m = n = 4$ and $c = 256$. The symbols $\triangle$ and $\circ$ represent CSSFN with $q = 2$ and $q = 4$ respectively, and $\diamond$ denotes the CSN model with 2 branches.

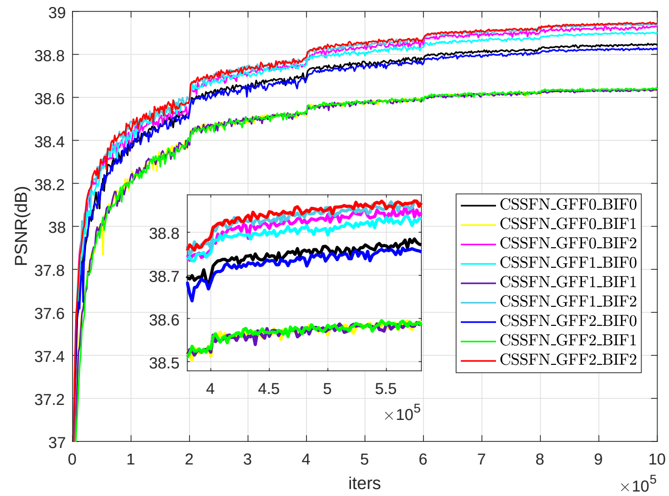


Fig. 6. Comparison of validation performance between different combinations of GFF and BIF. The PSNR curves are evaluated on $\mathcal{V}$(T2, BD) with $r = 2$ and correspond to the testing results in Table II.

C. Hierarchical Feature Fusion

The notorious problem of gradient vanishing and weakened information flow becomes more obvious as the network depth increases [35], [49], which hinders the training of deep models seriously. Unfortunately, the degradation of training samples will further aggravate the difficulty of training deep models in the context of medical images [19]. In order to promote the information flow in the network and improve the trainability of the model, many recent works have been devoted to resolving these problems. A popular method is to fuse the hierarchical features through skip connections, e.g., DenseNet [55] helps to explore new feature maps, and ResNet [58], [59] contributes to the reuse of the preceding features. The basic idea of fusing hierarchical features by residual learning and dense learning is also widely applied to many CNN-based methods, e.g., [17], [26], [29], [30], [31], [33], [35], [49], [56], [60], to build very wide and deep networks for performance improvement.

Since most recent CNN-based SR models are modular, the hierarchical feature fusion can be divided into local feature fusion (LFF) and global feature fusion (GFF), which integrate intra-block and inter-block features, respectively. The LFF is conducive to learning more effective hierarchical features and stabilizing model training [31], while the GFF enables short paths to be built from high-level features to low-level features directly and further alleviate the problem of gradient vanishing for training very deep networks [56]. In the proposed CSSFN model, local features are fused together in a serial manner, as shown in Fig.3(b) and Fig.4(c). It can be viewed as a manner of partially dense learning where the subfeatures are "densely" connected to subsequent layers. Then, a short skip connection (SSC) [31], [35], which is also called shortcut connection [58], [59] (Fig.3(a)), is used to conduct local residual learning. In terms of GFF, we present a dense global feature fusion (DGFF) for effective feature exploitation and important information preservation (Fig.2). Besides, it helps to alleviate the instability of model training caused by the increase in network depth and the decrease in network width.

III. Proposed Method

A. Network Architecture

In this paper, we focus on the task of single 2D MR image super-resolution. Given a LR image $\mathbf{x} \in \mathbb{R}^{h \times w}$, the target here is to recover a HR image $\mathbf{y} \in \mathbb{R}^{(r \cdot h) \times (r \cdot w)}$ that corresponds to the LR input $\mathbf{x}$, where r is the scaling factor. The overall architecture of the proposed CSSFN model is outlined in Fig.2 and Fig.3, which consists of three typical parts, i.e., shallow feature extraction, nonlinear mapping from shallow features to deep features and HR image recovery. As investigated in [19], we extract the shallow features by two 3×3 conv layers with a 1×1 conv layer in the middle. Denote $\mathcal{F}_{ext}(\cdot)$ as the corresponding mapping function of the entire shallow feature extraction stage, then the extracted shallowed features $\mathbf{x}_{\text{SF}}$ can be represented as:

$$\mathbf{x}_{\text{SF}} = \mathcal{F}_{ext}(\mathbf{x}), \tag{1}$$

where $\mathbf{x}$ denotes the original LR input. Next, $\mathbf{x}_{\text{SF}}$ is fed into the nonlinear mapping, which contains a series of stacked building blocks. The entire nonlinear mapping process can be expressed as follows:

$$\mathbf{x}_{\text{DF}} = \mathcal{F}_{nlm}(\mathbf{x}_0), \tag{2}$$

where $\mathbf{x}_0 = \mathbf{x}_{\text{SF}}$ is the extracted shallow features and $\mathcal{F}_{nlm}(\cdot)$ is the function corresponding to the entire nonlinear mapping process. To make more full use of the hierarchical features

TABLE II

ABLATION INVESTIGATION OF DIFFERENT GLOBAL FEATURE FUSION (GFF) AND BRANCH INFORMATION FUSION (BIF) METHODS. ALL THE MODELS ARE TRAINED ON $\mathcal{D}$(T2, BD) FOR ONE MILLION ITERATIONS AND TESTED ON $\mathcal{T}$(T2, BD). SF DENOTES SERIAL FUSION ($c = 256, m = n = q = 4$).

GFF	CGFF (GFF1)	×	√	×	×	×	√	√	×	×
	DGFF (GFF2)	×	×	√	×	×	×	×	√	√
BIF	MAR (BIF1)	×	×	×	√	×	√	×	√	×
	SF (BIF2)	×	×	×	×	√	×	√	×	√
$r = 2$	PSNR (dB)	39.90	39.95	39.88	39.66	40.01	39.66	40.03	39.68	40.05
	SSIM	0.9867	0.9868	0.9866	0.9862	0.9869	0.9861	0.9869	0.9863	0.9870

TABLE III

THE IMPACT OF THE OUTPUT WIDTH OF SERIAL FUSION ON THE PERFORMANCE OF THE MODEL. ALL THE MODELS ARE TRAINED ON $\mathcal{D}$(PD, BD) FOR ONE MILLION ITERATIONS AND TESTED ON $\mathcal{T}$(PD, BD). THE BASIC CONFIGURATION IS $c = 256$ AND $m = n = 4$ (PSNR (DB) | SSIM | P | D).

Width	r	$q=2$				$q=4$				$q=8$				$q=16$			
$c_o = \frac{c}{q}$	×2	41.45	0.9898	16.40M	59	41.30	0.9896	11.09M	91	41.20	0.9893	7.990M	155	40.99	0.9889	6.341M	283
	×3	36.15	0.9711	19.35M	59	35.99	0.9702	14.04M	91	35.83	0.9690	10.95M	155	35.56	0.9673	9.294M	283
	×4	33.71	0.9520	18.76M	60	33.60	0.9509	13.45M	92	33.38	0.9484	10.35M	156	33.07	0.9447	8.703M	284
$c_o = 64$	×2	41.32	0.9895	9.911M	59	41.33	0.9896	11.09M	91	41.35	0.9896	13.46M	155	41.28	0.9895	18.18M	283
	×3	36.06	0.9706	12.86M	59	36.01	0.9703	14.04M	91	36.02	0.9704	16.41M	155	35.99	0.9701	21.13M	283
	×4	33.57	0.9506	12.27M	60	33.59	0.9509	13.45M	92	33.59	0.9506	15.82M	156	33.59	0.9508	20.54M	284

and further stabilize model training, we also utilize the global feature fusion (GFF) [19], [31] to integrate these intermediate features. Unlike [19] and [31], we fuse the hierarchical features in a dense learning manner [56], instead of concatenating all the inter-block features together and then fuse them through a 1×1 conv layer. Thus, the input to the i-th building block is the concatenation of the output feature maps of all preceding blocks, i.e., $[\mathbf{x}_{i-1}, \ldots, \mathbf{x}_1, \mathbf{x}_0]$, where $[\ldots]$ denotes the concat operation along channel direction. Assuming that the mapping function of the i-th building block is $\mathcal{F}_b^i(\cdot)$, then we have:

$$\mathbf{x}_i = \mathcal{F}_b^i([\mathbf{x}_{i-1}, \ldots, \mathbf{x}_1, \mathbf{x}_0]), \quad i = 1, 2, \ldots, n, \tag{3}$$

where n is the number of building blocks in the network. Each block is connected to all preceding and subsequent blocks, and therefore facilitates the information propagation of the entire network. Iteratively, we can obtain the final output of all these stacked blocks:

$$\mathbf{x}_n = \mathcal{F}_b^n([\mathbf{x}_{n-1}, \ldots, \mathbf{x}_1, \mathbf{x}_0]). \tag{4}$$

Then, $[\mathbf{x}_n, \ldots, \mathbf{x}_0]$ is further integrated as the deep feature $\mathbf{x}_{\text{DF}}$ through a 1×1 conv layer and a 3×3 conv layer, followed by a global skip connection (GSC) [19], [26], [30], [31]:

$$\mathbf{x}_{\text{DF}} = \mathbf{x}_0 + \mathcal{F}_c([\mathbf{x}_n, \ldots, \mathbf{x}_0]), \tag{5}$$

where $\mathcal{F}_c(\cdot)$ corresponds to the mapping function of the two conv layers, as shown in Fig.2. Subsequently, the deep feature $\mathbf{x}_{\text{DF}}$ is used to recover the HR image $\mathbf{y}$ by the reconstruction sub network:

$$\mathbf{y} = \mathcal{F}_{rec}(\mathbf{x}_{\text{DF}}) = \mathcal{F}_{up}(\mathbf{x}_{\text{DF}}) + \hat{\mathbf{x}}, \tag{6}$$

where $\mathcal{F}_{up}(\cdot)$ represents the mapping function of the upscale module followed by a 3×3 conv layer, and $\hat{\mathbf{x}}$ is the (bicubic) interpolated version of $\mathbf{x}$. This is termed as an external skip connection (ESC) in [19], which approximates the residual between the original input and the final output of the network by interpolation [26], and further contributes to stabilizing the training process.

We adopt L_1 loss as the training objective. Given a training dataset $\mathcal{D} = \{\mathbf{x}^{(i)}, \mathbf{y}^{(i)}\}_{i=1}^{|\mathcal{D}|}$, where $|\mathcal{D}|$ denotes the number of training examples in $\mathcal{D}$. The loss function is expressed as:

$$L(\boldsymbol{\theta}) = \frac{1}{|\mathcal{D}|} \sum_{i=1}^{|\mathcal{D}|} ||\mathbf{y}^{(i)} - \mathcal{F}_{net}(\mathbf{x}^{(i)}; \boldsymbol{\theta})||_1. \tag{7}$$

Here, $\mathcal{F}_{net}(\cdot)$ denotes the function corresponding to the entire CSSFN network, and $\boldsymbol{\theta}$ is the set of model parameters.

B. Channel Splitting and Serial Fusion Block (CSSFB)

The structure of the proposed channel splitting and serial fusion block (CSSFB) is outlined in Fig.3(a). At the beginning of each CSSFB, there is a 3×3 channel compression layer, which is used to reduce the feature dimension of the input tensor to a predefined value c. According to (3), we have:

$$\mathbf{x}_{i-1,0} = \mathcal{H}_1^c([\mathbf{x}_{i-1}, \ldots, \mathbf{x}_1, \mathbf{x}_0]), \quad i = 1, 2, \ldots, n, \tag{8}$$

where $\mathcal{H}_1^c(\cdot)$ represents a convolution layer with 1×1 kernel size and c output channels. This implies that the feature map $\mathbf{x}_{i-1,0}$ has c channels.

Subsequently, a series of stacked channel splitting and serial fusion units (CSSFUs) form the main part of the CSSFB, as shown in Fig.3(a). Let's denote the function of the j-th CSSFU as $\mathcal{F}_u^j(\cdot)$, which we will describe in III-C in detail. Then we have the following equation for this CSSFU:

$$\mathbf{x}_{i-1,j} = \mathcal{F}_u^j(\mathbf{x}_{i-1,j-1}), \quad j = 1, 2, \ldots, m, \tag{9}$$

where m is the number of CSSFUs in each CSSFB. We can also get the output of the last CSSFU $\mathbf{x}_{i-1,m}$ iteratively:

$$\begin{aligned} \mathbf{x}_{i-1,m} &= \mathcal{F}_u^m(\mathbf{x}_{i-1,m-1}) \\ &= \mathcal{F}_u^m(\mathcal{F}_u^{m-1}(\cdots \mathcal{F}_u^1(\mathbf{x}_{i-1,0}) \cdots)). \end{aligned} \tag{10}$$

Local residual learning (LRL) [19], [30], [31], [35], [49] is another manner to stabilize model training. We also introduce LRL into the proposed CSSFB modules, so the final output of the i-th CSSFB can be expressed as:

$$\mathbf{x}_i = \mathbf{x}_{i-1,0} + \mathbf{x}_{i-1,m}. \tag{11}$$

TABLE IV
QUANTITATIVE COMPARISON BETWEEN DIFFERENT METHODS ON 6 TEST DATASETS (2 IMAGE DEGRADATIONS AND 3 MR IMAGE TYPES). THE MAXIMAL PSNR (DB) AND SSIM VALUES OF EACH COMPARISON GROUP ARE MARKED IN RED, AND THE SECOND ONES ARE MARKED IN BLUE (PSNR / SSIM).

Method	r	Bicubic downsampling $\mathcal{T}(:, \text{BD})$			k-space truncation $\mathcal{T}(:, \text{TD})$		
		PD	T1	T2	PD	T1	T2
Bicubic	×2	35.04 / 0.9664	33.80 / 0.9525	33.44 / 0.9589	34.65 / 0.9625	33.38 / 0.9460	33.06 / 0.9541
NLM [65]	×2	37.26 / 0.9773	35.80 / 0.9685	35.58 / 0.9722	36.18 / 0.9707	34.71 / 0.9581	34.56 / 0.9641
SRCNN [22]	×2	38.96 / 0.9836	37.12 / 0.9761	37.32 / 0.9796	38.23 / 0.9802	36.52 / 0.9705	37.04 / 0.9773
ESPCNN [28]	×2	38.27 / 0.9814	36.91 / 0.9747	36.92 / 0.9773	37.88 / 0.9792	36.35 / 0.9693	36.79 / 0.9754
VDSR [26]	×2	39.97 / 0.9861	37.67 / 0.9783	38.65 / 0.9836	39.89 / 0.9850	37.58 / 0.9760	38.74 / 0.9823
IDN [32]	×2	40.27 / 0.9869	37.79 / 0.9787	39.09 / 0.9846	40.43 / 0.9862	37.79 / 0.9765	39.48 / 0.9842
RDN [31]	×2	40.31 / 0.9870	37.95 / 0.9795	38.75 / 0.9838	40.39 / 0.9862	38.08 / 0.9784	40.02 / 0.9826
RecNet [33]	×2	40.43 / 0.9873	37.86 / 0.9792	39.13 / 0.9848	40.10 / 0.9857	37.54 / 0.9764	39.03 / 0.9832
FSCWRN [17]	×2	40.72 / 0.9880	37.98 / 0.9797	39.44 / 0.9855	40.91 / 0.9876	38.04 / 0.9786	39.82 / 0.9851
CSN [19]	×2	41.28 / 0.9895	38.27 / 0.9810	39.71 / 0.9863	41.77 / 0.9897	38.62 / 0.9813	40.47 / 0.9868
CSSFN-B4 [Ours]	×2	41.30 / 0.9896	38.33 / 0.9812	40.05 / 0.9870	41.91 / 0.9900	38.67 / 0.9815	40.64 / 0.9872
CSSFN-B2 [Ours]	×2	41.45 / 0.9898	38.36 / 0.9813	40.10 / 0.9871	41.97 / 0.9902	38.76 / 0.9818	40.73 / 0.9874
Bicubic	×3	31.20 / 0.9230	30.15 / 0.8900	29.80 / 0.9093	30.88 / 0.9167	29.79 / 0.8793	29.50 / 0.9016
NLM [65]	×3	32.81 / 0.9436	31.74 / 0.9216	31.28 / 0.9330	32.02 / 0.9324	30.83 / 0.9027	30.57 / 0.9197
SRCNN [22]	×3	33.60 / 0.9516	32.17 / 0.9276	32.20 / 0.9440	32.90 / 0.9432	31.72 / 0.9187	31.80 / 0.9381
ESPCNN [28]	×3	33.52 / 0.9505	32.10 / 0.9242	32.13 / 0.9421	32.54 / 0.9417	31.52 / 0.9140	31.64 / 0.9353
VDSR [26]	×3	34.66 / 0.9599	32.91 / 0.9378	33.47 / 0.9559	34.27 / 0.9555	32.57 / 0.9304	33.23 / 0.9515
IDN [32]	×3	34.96 / 0.9619	33.06 / 0.9394	33.92 / 0.9591	34.88 / 0.9598	32.86 / 0.9348	33.95 / 0.9569
RDN [31]	×3	35.08 / 0.9628	33.31 / 0.9430	33.91 / 0.9591	35.00 / 0.9609	33.33 / 0.9416	33.99 / 0.9576
RecNet [33]	×3	34.96 / 0.9623	33.05 / 0.9399	33.85 / 0.9588	34.67 / 0.9590	32.80 / 0.9347	33.69 / 0.9554
FSCWRN [17]	×3	35.37 / 0.9653	33.24 / 0.9423	34.27 / 0.9618	35.30 / 0.9636	33.09 / 0.9390	34.34 / 0.9603
CSN [19]	×3	35.87 / 0.9693	33.53 / 0.9464	34.64 / 0.9647	36.09 / 0.9697	33.68 / 0.9464	34.95 / 0.9653
CSSFN-B4 [Ours]	×3	35.99 / 0.9702	33.56 / 0.9468	34.84 / 0.9661	36.23 / 0.9706	33.73 / 0.9469	35.12 / 0.9663
CSSFN-B2 [Ours]	×3	36.15 / 0.9711	33.59 / 0.9471	34.96 / 0.9668	36.32 / 0.9713	33.75 / 0.9472	35.23 / 0.9671
Bicubic	×4	29.13 / 0.8799	28.28 / 0.8312	27.86 / 0.8611	28.82 / 0.8713	27.96 / 0.8182	27.60 / 0.8511
NLM [65]	×4	30.27 / 0.9044	29.31 / 0.8655	28.85 / 0.8875	29.27 / 0.8906	28.68 / 0.8439	28.37 / 0.8718
SRCNN [22]	×4	31.10 / 0.9181	29.90 / 0.8796	29.69 / 0.9052	30.52 / 0.9078	29.31 / 0.8616	29.32 / 0.8960
ESPCNN [28]	×4	31.02 / 0.9169	29.77 / 0.8781	29.32 / 0.9022	30.22 / 0.9034	29.29 / 0.8618	29.28 / 0.8954
VDSR [26]	×4	32.09 / 0.9311	30.57 / 0.8932	30.79 / 0.9240	31.69 / 0.9244	30.14 / 0.8818	30.51 / 0.9162
IDN [32]	×4	32.47 / 0.9354	30.74 / 0.8966	31.37 / 0.9312	32.33 / 0.9318	30.40 / 0.8889	31.31 / 0.9270
RDN [31]	×4	32.73 / 0.9387	31.05 / 0.9042	31.45 / 0.9324	32.64 / 0.9362	31.00 / 0.9018	31.49 / 0.9301
RecNet [33]	×4	32.58 / 0.9378	30.86 / 0.9005	31.30 / 0.9310	32.16 / 0.9310	30.46 / 0.8900	31.03 / 0.9243
FSCWRN [17]	×4	32.91 / 0.9415	30.96 / 0.9022	31.71 / 0.9359	32.78 / 0.9387	30.79 / 0.8973	31.71 / 0.9334
CSN [19]	×4	33.40 / 0.9486	31.23 / 0.9093	32.05 / 0.9413	33.51 / 0.9489	31.27 / 0.9092	32.28 / 0.9421
CSSFN-B4 [Ours]	×4	33.60 / 0.9509	31.34 / 0.9102	32.27 / 0.9441	33.64 / 0.9501	31.35 / 0.9095	32.46 / 0.9440
CSSFN-B2 [Ours]	×4	33.71 / 0.9520	31.37 / 0.9104	32.38 / 0.9453	33.75 / 0.9514	31.39 / 0.9098	32.57 / 0.9453

C. Channel Splitting and Serial Fusion Unit (CSSFU)

In the CSN network [19], the feature map transmitted to a channel splitting block (CSB) is first split into two branches with different network structures, which are then integrated together with the merge-and-run (MAR) mapping [34], [41]. In our CSSFN, the hierarchical feature is also split into several subfeatures with fewer channels. However, we do not transmit the information in a multi-branch way. Instead, the subfeatures are reintegrated into a single branch step-by-step through conv and concatenation operations, which can be viewed as partially dense learning with channel splitting.

The input map of the j-th CSSFU in the i-th CSSFB is first split into q subfeatures equally, i.e., $\{\mathbf{x}^0_{i-1,j-1}, \ldots, \mathbf{x}^{q-1}_{i-1,j-1}\}$. Denote $\mathbf{z}^k_{i-1,j-1}$ as the output of the k-th 3×3 conv layer in Fig.3(b) (cubes in dark gray), which is followed by a ReLU layer [61]. Then we have:

$$\mathbf{z}^k_{i-1,j-1} = \max\left(0, \mathcal{H}^{c/q}_3([\mathbf{x}^{k-1}_{i-1,j-1}, \mathbf{z}^{k-1}_{i-1,j-1}])\right), \quad (12)$$

where $k = 1, 2, \ldots, q$ and $\mathbf{z}^0_{i-1,j-1} = 0$. Therefore, all these subfeatures are reintegrated together and the network is in a single branch. Finally, we extend the channel of the last output feature, $\mathbf{z}^q_{i-1,j-1}$, by a 3×3 channel extension layer at the end of the CSSFU:

$$\mathbf{x}_{i-1,j} = \mathcal{H}^c_3(\mathbf{z}^q_{i-1,j-1}), \quad (13)$$

where $\mathbf{x}_{i-1,j}$ is the output feature map of the j-th CSSFU in the i-th CSSFB. It is worth pointing out that the purpose of channel splitting in this paper is not to form a multi-branch structure, but to be a preprocessing of the subsequent serial fusion. The single-branch structure makes the network deeper and narrower, which makes model training more unstable. This is part of the reason why we adopt DGFF to fuse inter-block features. Therefore, channel splitting and serial fusion can be regarded as "stretching" a relatively shallow but wider network into a deeper but narrower network.

D. Network Depth and Model Parameters

The network depth is usually defined as the length of the longest path from the input to the output [19], [34]. According to the entire structure of the proposed model, the depth of our CSSFN network is given by:

$$D = n[1 + m \times (q + 1)] + s + 6, \quad (14)$$

where s denotes the depth of the upscale modula and depends on the specific value of the scaling factor r. Specifically, $s = 1$ for $r = 2$ or $r = 3$, and $s = 2$ for $r = 4$. The first "1" in

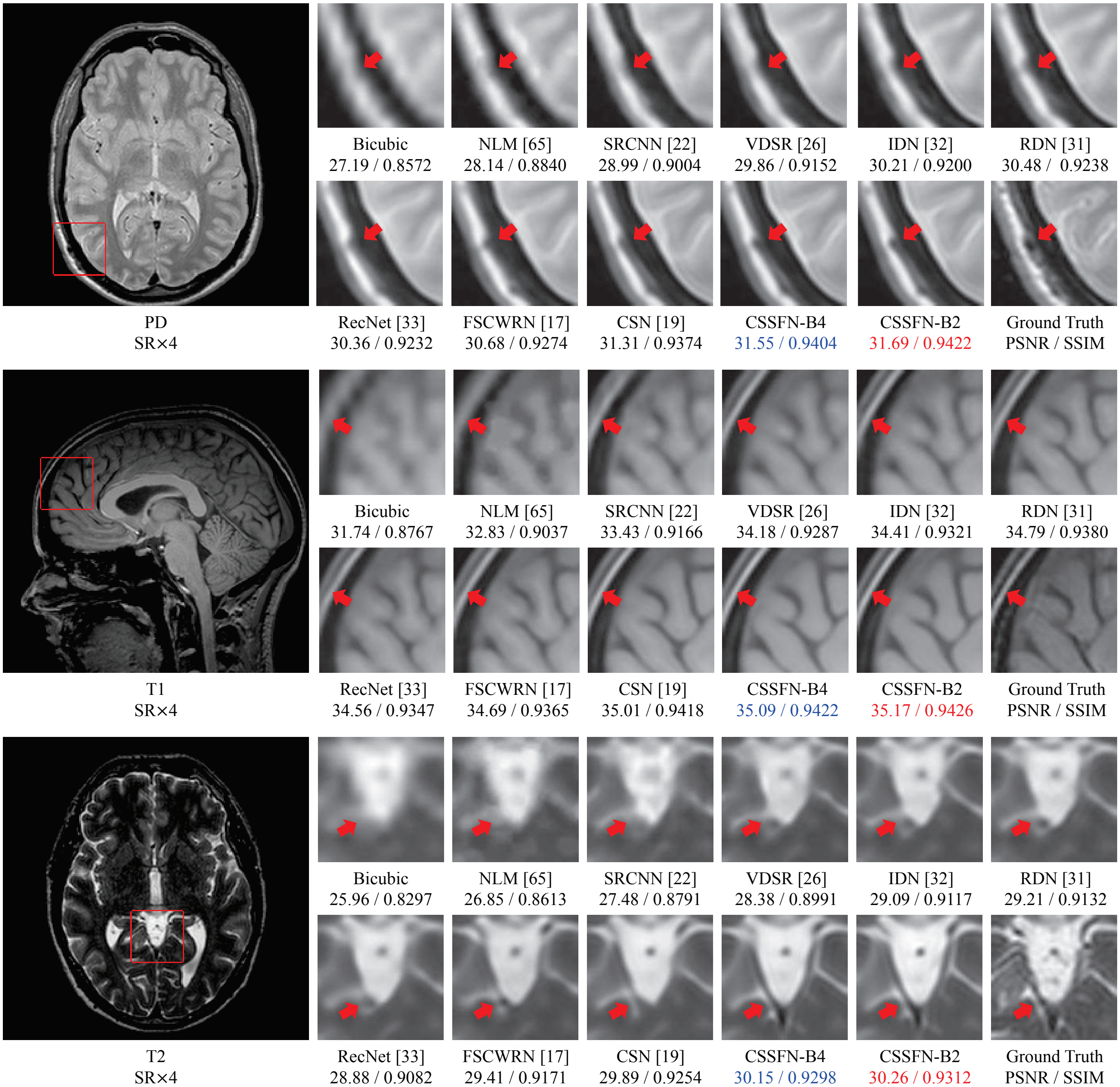


Fig. 7. The visual comparison between several advanced SISR methods on a PD (top), T1 (middle) and T2 (bottom) image with $r = 4$. The image degradation is **bicubic degradation**. The maximal PSNR (dB) and SSIM for each group of comparison are in red and the second ones are in blue.

(14) corresponds to the compression layer at the beginning of each CSSFB, and the second one denotes the extension layer at the end of each CSSFU.

Table I collects the network depth (D) and the number of model parameters (P) of the proposed CSSFN model under several configuration conditions, where pseudo 3D execution implies that the model regards 96 slices of a 3D MR volume as 96 channels of a 2D image. As can be seen, all models need to determine about 10M $\sim$ 20M model parameters. The most similar model to our CSSFN is the CSN [19], so we display the comparison of network configuration between the CSN and our CSSFN in Fig.5. We can observe that the proposed CSSFN model increases in network depth for both $q = 4$ and $q = 2$. However, it has fewer model parameters when $q = 4$ and more model parameters when $q = 2$.

IV. Experimental Results

In this section, we first briefly introduce the datasets used in this work and the details of model implementation. Then we investigate and analyze the structure of our network, including the influence of the way of global feature fusion (GFF) and branch information fusion (BIF), and the number of branches (q) on the model performance. Finally, the proposed method is compared with other advanced methods quantitatively and

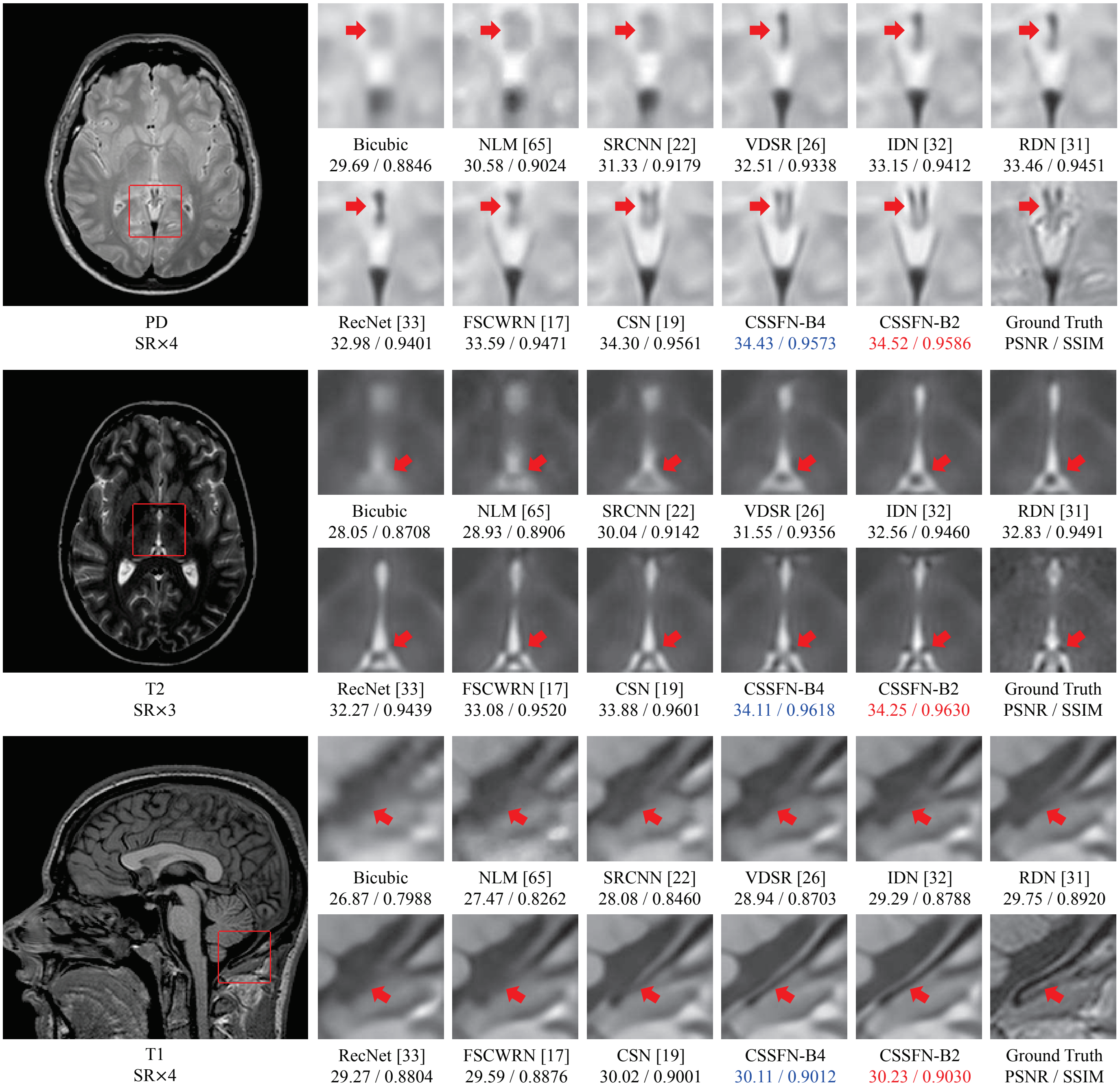


Fig. 8. The visual comparison between several state-of-the-art SISR methods on a PD (top), T2 (middle) and T1 (bottom) image with $r = 4$, $r = 3$ and $r = 4$, respectively. The image degradation is **truncation degradation**. The maximal PSNR (dB) and SSIM for each group of comparison are in red and the second ones are in blue.

qualitatively. The frequently used peaks signal to noise ratio (PSNR) and structural similarity index metric (SSIM) [62] are taken as the metrics of quantitative evaluation.

A. Datasets and Implementation Details

We utilize the same MR datasets as [19] to perform relevant experiments. They are derived from the IXI[1] dataset and have three types of 3D MR volumes: T1-weighted, T2-weighted and PD-weighted images, each of which contains 500, 70 and 6 volumes for training, testing and quick validation, respectively. Two LR image degradations, i.e., bicubic downsampling (BD) and k-space truncation (TD), are implemented to simulate LR images. For convenience, we follow the convention of [19] to indicate a particular subset of data, i.e., the sub dataset with a specific type of MR images and degradation is denoted as dataset type (MR type, degradation). The size of each 3D MR volume is cut to 240×240×96, where 96 is the number of slices of a 3D volume. If the model takes a single slice of a 3D volume as input, we call it *pure 2D execution*; if the model regards 96 slices as 96 channels of a 2D input, we term it as

[1] http://brain-development.org/ixi-dataset/

TABLE V

QUANTITATIVE COMPARISON BETWEEN SEVERAL METHODS IN CASE OF PSEUDO 3D EXECUTION. THE MAXIMAL PSNR (DB) AND SSIM OF EACH COMPARATIVE GROUP ($\mathcal{T}$(:, BD) AND $\mathcal{T}$(:, TD)) ARE MARKED IN RED, AND THE SECOND ONES ARE MARKED IN BLUE.

	r	Bicubic downsampling $\mathcal{T}$(:, BD))				k-space truncation $\mathcal{T}$(:, TD))			
		EDSR [30]	CSN [19]	CSSFN ($q=4$)	CSSFN ($q=2$)	EDSR [30]	CSN [19]	CSSFN ($q=4$)	CSSFN ($q=2$)
PD	×2	39.87/0.9857	40.15/0.9865	40.28/0.9869	40.34/0.9871	39.47/0.9837	39.50/0.9839	39.80/0.9849	39.91/0.9853
	×3	34.39/0.9578	34.68/0.9598	34.78/0.9609	34.76/0.9611	33.97/0.9531	34.12/0.9540	34.24/0.9554	34.15/0.9550
	×4	31.80/0.9284	32.19/0.9325	32.21/0.9332	32.11/0.9329	31.44/0.9219	31.72/0.9246	31.78/0.9252	31.68/0.9257
T1	×2	37.56/0.9774	37.60/0.9778	37.74/0.9786	37.81/0.9789	37.09/0.9741	36.99/0.9737	37.18/0.9748	37.25/0.9754
	×3	32.76/0.9347	32.83/0.9360	32.86/0.9362	32.85/0.9366	32.27/0.9274	32.25/0.9266	32.34/0.9276	32.32/0.9275
	×4	30.46/0.8902	30.53/0.8915	30.58/0.8919	30.61/0.8923	30.04/0.8803	30.07/0.8794	30.09/0.8795	30.14/0.8812
T2	×2	38.28/0.9824	38.53/0.9831	38.79/0.9836	38.92/0.9842	38.11/0.9803	38.20/0.9807	38.54/0.9817	38.92/0.9842
	×3	33.15/0.9528	33.36/0.9547	33.46/0.9556	33.50/0.9559	32.89/0.9482	33.00/0.9490	33.21/0.9512	33.26/0.9518
	×4	30.52/0.9198	30.81/0.9231	30.93/0.9242	30.89/0.9241	30.31/0.9137	30.54/0.9163	30.62/0.9182	30.58/0.9178

pseudo 3D execution. The model parameters vary slightly with the number of input channels, as shown in Table I.

The overall configuration of the proposed network is shown Fig.2 and Fig.3 with $c = 256$ and $m = n = q = 4$. The size of minibatch is set to 16. The kernel size of each convolutional layer is marked in Fig.2 and Fig.3. For each convolution layer in CSSFU, we keep the channel size of the output feature is the same as that of subfeature maps, i.e., c/q, except that the last channel extension layer has c output channels. For fair comparison, we also train the model with LR image patches of size 24×24 with the corresponding HR patches. These training patches are further augmented by random horizontal flips and 90° rotations, just like [19], [30], [31], [35]. All models are implemented in TensorFlow 1.11.0 and trained on a NVIDIA GeForce GTX 1080 Ti GPU for one million iterations. We apply Xavier's method [63] to initialize model parameters. The optimizer to minimize the L_1 loss is the Adam algorithm [64] with $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-8}$. Learning rate is initialized as 0.0001 for all layers and decays in a piecewise constant manner, i.e., halved at every 200K iterations.

B. Feature Fusion

In this section, we investigate the effects of GFF and BIF on the performance of the model. To this end, we designed several structures for ablation investigation. For GFF, we compare the DGFF (Fig.2) and the method adopted by [19], [31], which we term as concat global feature fusion (CGFF). On the other hand, we compare the proposed serial fusion (SF, Fig.3(b) and Fig.4(c)) and the MAR mapping [41] (Fig.4(b)) for BIF. Note that the latter can be regarded as a way of local parallel fusion for multi-branch structure. In addition, we also constructed a benchmark structure without either GFF or BIF, where the GFF is removed from the entire network and the part in the dotted box in Fig.4(c) is replaced with the basic Conv + ReLU structure in Fig.4(a). Table II shows the results of comparison evaluated on $\mathcal{T}$(T2, BD), for SR×2. As can be seen, the benchmark structure without channel splitting (the 3rd column) achieves PSNR value of 39.90dB, a relatively good result. This is probably because that the stage mapping in the benchmark structure (refer to Fig.4(a) and Fig.4(c)) evolves into the residual block structure of EDSR[2] model [30].

[2] Nevertheless, the model is still trainable here because $m = n = 4$ means that the overall parameters and network depth of the model are much smaller than those of the original EDSR model.

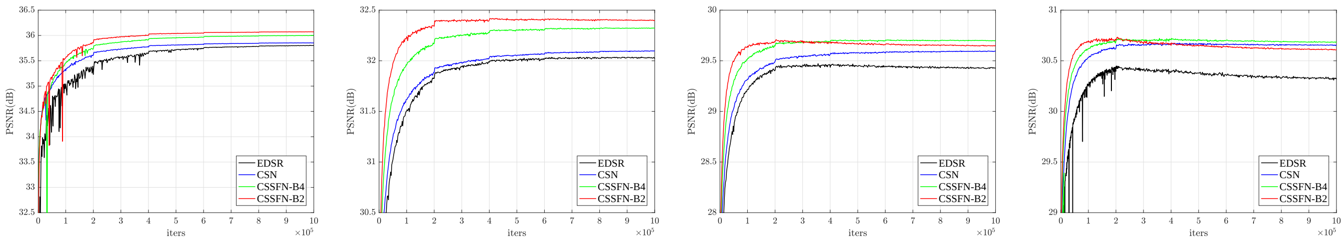
For convenience, we use two numbers to represent different GFF and BIF methods, 0 for the benchmark structure, GFF1 for CGFF, GFF2 for DGFF, BIF1 for MAR (Fig.4(b)), and BIF2 for SF (Fig.4(c)). According to the 6th, 8th and 10th columns, we can observe that the MAR mapping degrades the model performance seriously when $q = 4$, which implies that one can not improve the SR performance by simply increasing the branch number of the CSN [19]. On the contrary, our SF strategy can boost the performance of the model (the 7th, 9th and 11th columns), compared with the benchmark structure (the 3rd column). Another interesting thing is that CGFF performs significantly better than DGFF if without channel splitting (the 4th column vs. the 5th column). However, the situation is reversed if with channel splitting (the 8th column vs. the 10th column and the 9th column vs. the 11th column). This shows that the combination of DGFF and SF can better promote the flow of information in the network and improve the model performance.

We also visualize the convergence process of these models in Fig.6. These validation curves are almost consistent with the results displayed in table II and the above analysis, and the comparison is more obvious. Both the quantitative results in Table II and the visualization of the validation process demonstrate the effectiveness and benefits of the proposed SF and the combination with DGFF.

C. Channel Splitting

The output width of serial fusion, i.e., c/q, will be changed according to the number of subfeatures in previous settings. In this case, the number of model parameters, P, decreases as q increases. However, if we set the output channel of serial fusion to a fixed value, then P will increase with the increase of q. Denote the output channel of serial fusion as c_o, we study the effects of the number of subfeatures and c_o on the model performance. To this end, we train the CSSFN model with different configurations with $\mathcal{D}$(PD, BD) and collect the results in Table III.

1) Unfixed Output Width: In this case, $c_o = c/q$. According to rows 2, 3 and 4 of Table III, we can see that the model performance degrades with the increase of q. This is easy to understand, because the increase of q reduces the parameters of the model sharply, although it superficially increases the network depth.



(a) $\mathcal{V}$(T1, BD) with $r = 2$ (b) $\mathcal{V}$(T2, TD) with $r = 3$ (c) $\mathcal{V}$(T2, TD) with $r = 4$ (d) $\mathcal{V}$(PD, BD) with $r = 4$

Fig. 9. The validation performance comparison of the compared methods on several randomly selected sub datasets, in case of pseudo 3D execution. It can be observed that the severity of model performance degradation due to over-fitting/under-fitting: (a) < (b) < (c) < (d).

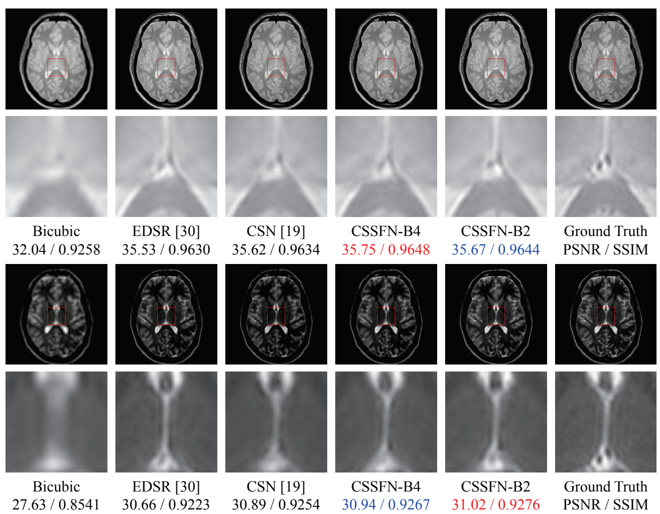


Fig. 10. The visual comparison of pseudo 3D execution on a PD (top) and a T2 (bottom) image with $r = 3$ and $r = 4$, respectively. The image degradation is **truncation degradation**.

2) Fixed Output Width: We set $c_o = 64$ for comparison in this case. As shown in rows 5, 6 and 7 of Table III, we can not obtain significant performance gains by increasing the number of subfeatures, q. This result is strange, because the parameters of the model (P) and the depth of the network (D) increase with the increase of q, but the model performance does not improve and even seems to decline (e.g., SR×3). Increasing q actually makes it more difficult to integrate these subfeatures in an effective way. Therefore, channel discrimination should *not* aggressively increase the number of subfeatures unless there exists a more effective information fusion mechanism.

From Table III, we can generally draw the conclusion that the performance of the model is mainly related to c_o when the overall framework remains unchanged, regardless of how the model parameters and the network depth change with q.

D. Comparison with Other Methods

To illustrate the effectiveness and superiority of the CSSFN models, we compare them with several typical SISR methods in this section, including:

* a classic traditional method for MR image upsampling: NLM [65];
* two light-weight and representative models: SRCNN [22] and ESPCNN [28];
* two moderate-scale models: VDSR [26] and IDN [32];
* two large-scale CNN models: RDN [31] and EDSR [30];
* two CNN-based models specifically for single MR image SR: FSCWRN [17] and CSN [19];
* a UNet-based model specifically for general MR image reconstruction: RecNet [33];

Some results are directly cited from [19] because of the same datasets and image degradations, while others are obtained by retraining the models with the datasets described in IV-A.

1) Bicubic Degradation (BD): As one of the most common image degradation models for simulating LR images, bicubic degradation simply utilizes the bicubic kernel to reduce image size in image space. The Table IV exhibits the quantitative results of the compared methods over the testing datasets of PD, T1 and T2 MR images for all SR scales ($\mathcal{T}$(:, BD)). It can be seen that our CSSFN model surpasses the CSN model [19] and achieves the best SR performance on all MR image types and all scaling factors. In particular, the CSSFN-B4 model has relatively few model parameters but also gives excellent performance on all MR image types and SR scales.

Fig.7 shows the visual comparison between these SR methods on a PD, T1 and T2 image with SR×4. We can see that most CNN-based methods (e.g., VDSR [26] and RDN [31]) can surpass the traditional methods (bicubic and NLM [65]) by a large margin, achieving very good approximations to the ground truth. Although it is not easy to observe the difference between our methods and other CNN-based methods on the PD and T1 images, but the proposed CSSFN models are the most accurate in quantitative evaluation. Besides, we can see from the position indicated by the red arrow on the T1 image that the results of our CSSFN models show sharper edges, and the dark trench between two bright curves is clearer than that of other methods. A more significant visual difference can be observed on the T2 image. The gap between the white and gray areas is more obvious in the results of our models, while it can hardly be observed in the results of other methods. In addition, the black spot in the white area can be more clearly observed in the results of our models.

2) Truncation Degradation (TD): The k-space truncation of the HR image is a process that simulates the real MR image acquisition process where a LR image is scanned by reducing acquisition lines in phase and slice encoding directions [19]. When the scaling factor r is the same, the k-space truncation often reduces image quality more seriously than the bicubic downsampling due to the "steep" loss of the k-space data. This

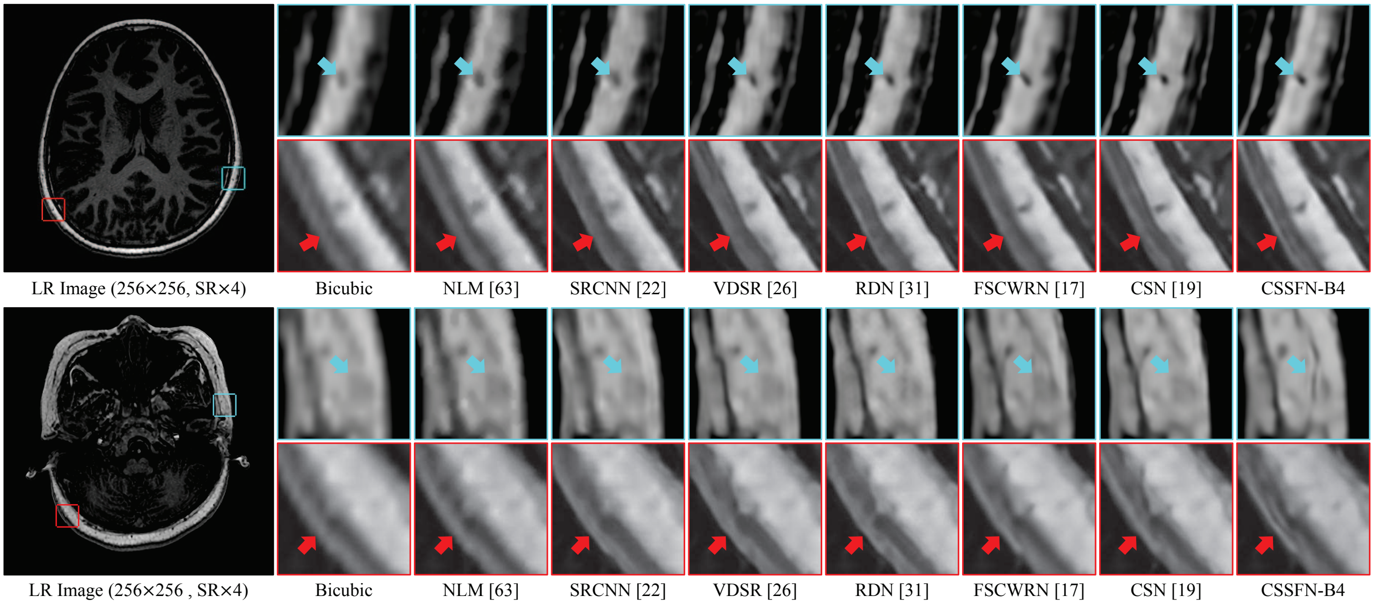


Fig. 11. The visual comparison between several state-of-the-art SISR methods on in-vivo T1 MR images with $r = 4$. In this case, the ground truth HR images are not available.

can be verified by the fact that bicubic interpolation performs better in bicubic downsampling than in k-space truncation (see the 3rd and 6th columns of Table IV). Table IV also presents the quantitative results of the compared methods in terms of the truncation degradation. Again, our CSSFN models exhibit better SR performance on all MR image types and all scaling factors. But more importantly, the CSSFN models (and the CSN model [19]) perform better than in the case of the bicubic downsampling, which may imply that the proposed models are more suitable for MR image super-resolution tasks.

Visually, we can more easily observe the advantages of the proposed method over other methods. Fig.8 shows the visual effect of the compared methods on a PD, T2 and T1 image with $r = 4$, $r = 3$ and $r = 4$, respectively. The proposed method recovers images with clearer and sharper edges, thus making them more faithful to the ground truth HR images. In this case, the visual advantage of our CSSFN models is more observable. For instance, in PD and T2 images, the proposed models show clearer contours and details at the locations indicated by the red arrows. In particular, CSSFN-B2 has more obvious advantages and is closer to the ground truth HR images. In the T1 image, the superiority of the proposed models is more obvious. The dark trench that can be clearly identified in the results of our models can hardly be marked in the results of other methods.

3) Pseudo 3D Execution: One of the major problems of training large-scale models with MR images is the degradation of training samples. This problem can be alleviated by pseudo 3D execution at the cost of accuracy reduction [19], due to overfitting or underfitting. To compare the methods more comprehensively, we also conduct the comparative experiments of pseudo 3D execution. Note that we do not include other models here because the reduction of training samples makes the training of most other algorithms fail.

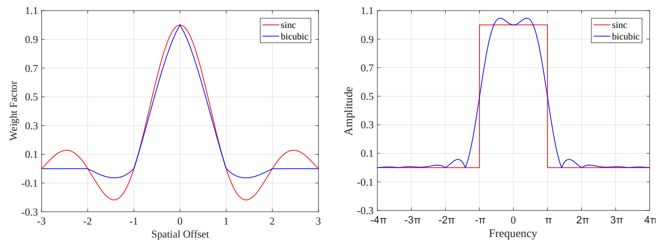
Table V shows the quantitative comparison between EDSR [30], CSN [19], CSSFN-B4 and CSSFN-B2 in this case, with both image degradations. The performance improvement of the proposed CSSFN models is obvious. However, the advantage of CSSFN-B2 over CSSFN-B4 seems to be diminished when comparing with pure 2D execution, such as $\mathcal{T}$(PD, BD) with $r = 4$ and $\mathcal{T}$(T1, TD) with $r = 3$. In some cases, the PSNR performance of CSSFN-B2 is even worse than that of CSN [19], e.g., $\mathcal{T}$(PD, TD) with $r = 4$. Actually, this is primarily due to the underfitting or overfitting caused by training sample reduction, as shown in Fig.9. It can be seen that CSSFN-B2 essentially performs better than CSSFN-B4 in that it converges faster and has higher PSNR maximums. Besides, only CSN [19] did not show obvious overfitting/underfitting in all cases. Fig.10 shows the visual comparison between these methods in terms of pseudo 3D execution, from which we can also observe that the proposed methods provide a clearer indication of the underlying structure, compared with other methods. Despite the overfitting/underfitting, we still train the models for one million iterations to keep the training iterations consistent with those of other comparative methods.

4) Performance on in-vivo Images: We also execute SR experiments on two in-vivo T1 images collected from Alltech Medical Systems Co., LTD. In this case, the ground truth HR images are not available and the image degradation is unknown either. We compare the proposed CSSFN-B4 model with NLM [65], SRCNN [22], VDSR [26], RDN [31], FSCWRN [17] and CSN [19]. As shown in Fig.11, our CSSFN-B4 model recovers sharper edges and finer details than other state-of-the-art methods, as indicated by the arrows.

V. Discussion

A. Channel Discrimination Ability

In the CSN model [19], the channel discrimination ability of the model is achieved by different branch structures. The



(a) (b)

Fig. 12. Graphical comparison between bicubic downsampling and k-space truncation in spatial domain (a) and frequency domain (b). The truncation in frequency domain corresponds to a sinc function in spatial domain (1D).

hierarchical features is divided into 2 parts by channel splitting and fused together by the marge-and-run mapping [41]. But the propagation paths of each subfeature have different branch structures. In the proposed CSSFN model, the subfeatures are also processed discriminatorily though they are placed in a single branch, in that the concat connections locate them at different depths of the network. This can be regarded as the *fine-grained hierarchy* of the intermediate features, thus realizing partial continuous memory mechanism [31], which is believed to be beneficial to the feedback propagation of the network [59].

From Table III, we can observe that the model performance degrades gradually with the increase of subfeatures (q) when $c_o = c/q$. However, the increase of model parameters leads no obvious performance gains when $c_o = 64$. Thus, we speculate that the degradation of model performance is mainly because that increasing subfeatures will also increase the difficulty of information fusion. Further more, if more effective information fusion mechanisms are explored, the trade-off between model performance and model scale can be further improved.

B. Image Degradation Model

As in [19], we also investigate two image degradation model in this work, i.e., the bicubic downsampling and the k-space truncation. The truncation degradation can be considered as more aggressive because the information outside the sampling range is "steeply" cut off without any cushion (Fig.12). As mentioned above, it can be verified by the fact that the bicubic interpolation performs better in bicubic degradation than in truncation degradation. However, the performance of the last few models (e.g., CSN and CSSFN) in Table IV is contrary to that of bicubic interpolation. On the other hand, the truncation degradation simulates the real MRI acquisition that operates in k-space and truncates the frequency spectrum of the object. This indicates that these models may be more suitable for the scenarios of MR image super-resolution due to stronger representational capacity.

VI. Conclusion

Channel discrimination is an effective way to improve the SR performance of deep models in the context of degradation of training samples, but how to fuse the channel information remains to be explored. We presented a serial fusion strategy for channel discrimination in this work. The hierarchical features are first divided into several "small" subfeatures along channel direction, which are then integrated into a single branch in a serial manner. The proposed CSSFN model has the channel discrimination ability in that each subfeature is processed in different network depths. To further improve the information flow of the network, we combine serial fusion with a DGFF to integrate the intermediate features. Extensive quantitative and qualitative experiments demonstrate that our CSSFN model achieves superiority over other state-of-the-art SISR methods. Besides, the serial fusion mode might shed some light on the development of other information fusion methods and channel discrimination methods.

Acknowledgment

The work is supported in part by the National Key Research and Development Program of China (No. 2016YFC0100800 and 2016YFC0100802).

References

[1] E. Carmi, S. Liu, N. Alona, A. Fiat, D. Fiat, "Resolution enhancement in MRI," *Magn. Reson. Imag.*, vol. 24, no. 2, pp. 133-154, Feb. 2006.
[2] O. Oktay, *et al.*, "Multi-input cardiac image super-resolution using convolutional neural networks," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Intervent*, Oct. 2016, pp. 246-254.
[3] E. Plenge, D. H. J. Poot, M. Bernsen, *et al*, "Super-resolution methods in MRI: can they improve the trade-off between resolution, signal-to-noise ratio, and acquisition time?," *Magn. Reson. Med.*, vol. 68, no. 6, pp. 1983-1993, Feb. 2012.
[4] R. G. Keys. "Cubic convolution interpolation for digital image processing," *IEEE Transactions on Acoustics Speech and Signal Processing*, vol. 29, no. 6, pp. 1153-1160, Dec. 1981.
[5] D. Gottlieb and C. W. Shu, "On the gibbs phenomenon and its resolution," *Siam Review*, vol. 39, no. 4, pp. 644-668, Dec. 1997.
[6] X. Li, Orchard, M. T. "New edge-directed interpolation," *IEEE Trans. Image Process.*, vol. 10, no. 10, pp. 1521-1527, Oct. 2001.
[7] J. Sun, Z. Xu, H. Y. Shum. "Image super-resolution using gradient profile prior," *IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1-8.
[8] M. Irani and S. Peleg, "Improving resolution by image registration," *CVGIP: Graph. Model. Image Process.*, vol. 53, no. 3, pp. 231-239, May. 1991.
[9] H. Stark and P. Oskoui, "High resolution image recovery from image plane arrays using convex projections," *J. Opt. Soc. Am.*, vol. 6, no. 11, pp. 1715C1726, 1989.
[10] W. T. Freeman, T. R. Jones, E.C. Pasztor, "Example-based super-resolution," *IEEE Comput. Graph. Appl.*, vol. 22, no. 2, pp. 56-65, Aug. 2002.
[11] C. Kim, K. Choi, J. B. Ra, "Example-based super-resolution via structure analysis of patches," *IEEE Signal. Process. Lett.*, vol. 20, no. 4, pp. 407-410, Mar. 2013.
[12] J. Yang, J. Wright, T. Huang, Y. Ma, "Image super-resolution as sparse representation of raw image patches," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp.1-8.
[13] J. Yang, Z. Wang, Z. Lin, *et al.*, "Coupled Dictionary Training for Image Super-Resolution," *IEEE Trans. Image Process.*, vol. 21, no. 8, pp. 3467-3478, Apr. 2012.
[14] S. Peled and Y. Yeshurun, "Superresolution in MRI: application to human white matter fiber tract visualization by diffusion tensor imaging," *Mag. Reson. Med.*, vol. 45, no. 1, pp. 29-35, Jan. 2001.
[15] H. Greenspan, G. Oz, N. Kiryati, and S. Peled, "MRI inter-slice reconstruction using super-resolution," *Magn. Reson. Imag.*, vol. 20, no. 5, pp. 437-446, Jun. 2002.
[16] A. Rueda, N. Malpica, E. Romero, "Single-image super-resolution of brain MR images using over complete dictionaries," *Med. Image Anal.*, vol. 17, no. 1, pp. 113-132, Jan. 2013.
[17] J. Shi, Z. Li, S. Ying, C. Wang, Q. Liu, Q. Zhang, and P. Yan, "MR image super-resolution via wide residual networks with fixed skip connection," IEEE Journal of Biomedical and Health Informatics, 2018.
[18] Y. H. Wang, J. Qiao, J. B. Li, P. Fu, S. C. Chu, and J. F. Roddick, "Sparse representation-based MRI super-resolution reconstruction," *Measurement*, vol. 47, no. 1, pp. 946-953, Jan. 2014.

[19] X. Zhao, Y. Zhang, T. Zhang, X. Zou. (2018). "Channel splitting network for single MR image super-resolution." [Online]. Available: https://arxiv.org/abs/1810.06453

[20] W. Yang, X. Zhang, Y. Tian, W. Wang, Xue, J. H. (2018). "Deep learning for single image super-resolution: a brief review," [Online]. Available: https://arxiv.org/abs/1808.03344

[21] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436-444, May. 2015.

[22] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Trans. Pattern Analy. Mach. Intell.*, vol. 38, no. 2, pp. 295-307, Feb. 2015.

[23] C. Dong, C. C. Loy, and X. Tang, "Accelerating the super-resolution convolutional neural network," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Oct. 2016, pp.391-407.

[24] J. Kim, J. K. Lee, K. M. Lee, "Deeply-recursive convolutional network for image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp.1637-1645.

[25] Y. Tai, J. Yang, and X. Liu, "Image super-resolution via deep recursive residual network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2790-2798.

[26] J. Kim, J. K. Lee, and K. M. Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1646C1654.

[27] Y. Tai, J. Yang, X. Liu, and C. Xu, "Memnet: a persistent memory network for image restoration," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Oct. 2017, pp. 4549-4557.

[28] W. Shi, J. Caballero, H. Ferenc, J. Totz, A. P. Aitken, *et al*. "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1874-1883.

[29] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, *et al*. (2016). "Photo-realistic single image super-resolution using a generative adversarial network," [Online]. Available: https://arxiv.org/abs/1609.04802

[30] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "Enhanced deep residual networks for single image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops. (CVPRW)*, Jul. 2017, pp.1132-1140.

[31] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu. (2018). "Residual dense network for image super-resolution." [Online]. Available: https://arxiv.org/abs/1802.08797

[32] Z. Hui, X. Wang, X. Gao. (2018). "Fast and accurate single image super-resolution via information distillation network." [Online]. Available: https://arxiv.org/abs/1803.09454

[33] C. M. Hyun, H. P. Kim, S. M. Lee, S. Lee, J. K. Seo. (2018). "Deep learning for undersampled MRI reconstruction." [Online]. Available: https://arxiv.org/abs/1709.02576

[34] Y. Hu, X. Gao, J. Li, Y. Huang, and H. Wang. (2018). "Single image super-resolution via cascaded multi-scale cross network." [Online]. Available: https://arxiv.org/abs/1802.08808

[35] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, Y. Fu. (2018). "Image super-resolution using very deep residual channel attention networks," [Online]. Available: https://arxiv.org/abs/1807.02758

[36] C. H. Pham, A. Ducournau, R. Fablet, and F. Rousseau, "Brain MRI super-resolution using deep 3D convolutional networks," in *Proc. IEEE Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2017, pp. 197-200.

[37] Y. Chen, Y. Xie, Z. Zhou, F. Shi, A. G. Christodoulou, and D. Li. (2018). "Brain MRI super resolution using 3D deep densely connected neural networks." [Online] Available: http://cn.arxiv.org/abs/1801.02728

[38] Y. Chen *et al*. (2018). "Efficient and accurate MRI super-resolution using a generative adversarial network and 3D multi-level densely connected network." [Online]. Available: https://arxiv.org/abs/1803.01417

[39] C. You, G. Li, Y. Zhang, X. Zhang, H. Shan, S. Ju, *et al*. (2018). "CT super-resolution GAN constrained by the identical, residual, and cycle learning ensemble(gan-circle)," [Online]. Available: https://arxiv.org/abs/1808.04256

[40] C. Zhao, A. Carass, B. E. Dewey, and J. L. Prince. (2018). "Self super-resolution for magnetic resonance images using deep networks." [Online]. Available: https://arxiv.org/abs/1802.09431

[41] L. Zhao, J. Wang, X. Li, Z. Tu, and W. Zeng. (2017). "Deep convolutional neural networks with merge-and-run mappings." [Online]. Available: https://arxiv.org/abs/1611.07718

[42] R. Z. Shilling *et al.*, "A super-resolution framework for 3D high-resolution and high-contrast imaging using 2-D multislice MRI," *IEEE Trans. Med. Imag.*, vol. 28, no. 5, pp. 633-644, May. 2009.

[43] J. V. Manjón, P. Coupé, A. Buades, D. L. Collins, and M. Robles, "MRI super-resolution using self-similarity and image priors," *Int. J. Biomed. Imaging.*, vol. 2010, Article ID: 425891 (11 pages), Dec. 2010.

[44] F. Rousseau, "Brain hallucination," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Oct. 2008, pp. 497C508.

[45] D. H. Trinh, M. Luong, F. Dibos, J. M. Rocchisani, C. D. Pham, T. Q. Nguyen, "Novel example-based method for super-resolution and denoising of medical images," *IEEE Trans. Image Process.*, vol. 23, no. 4, pp. 1882-1895, Apr. 2014.

[46] J. Hu, X. Wu, J. Zhou, "Single image super resolution of 3D MRI using local regression and intermodality priors," in *Proc. Int. Conf. Digit. Image Process.*, vol.10033, 2016. pp.100334C.

[47] S. Roohi, J. Zamani, M. Noorhosseini, and M. Rahmati, "Super-resolution MRI images using Compressive Sensing," in *Iranian Conference on Electrical Engineering (ICEE2012)*, May. 2012 pp. 1618-1622.

[48] J. Hu, L. Shen, S. Albanie, G. Sun. (2017). "Squeeze-and-excitation networks." [Online]. Available: https://arxiv.org/abs/1709.01507

[49] Y. Hu, J. Li, Y. Huang, X. Gao. (2018). "Channel-wise and spatial feature modulation network for single image super-resolution." Available: https://arxiv.org/abs/1809.11130v1

[50] E. Mansimov, E. Parisotto, J. L. Ba, and R. Salakhutdinov, "Generating images from captions with attention," in *Int. Conf. Learn. Rep. (ICLR)*, May. 2016, pp. 1C4.

[51] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, R. S. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *Int. Conf. Mach. Learn. (ICML)*, Jul. 2015, pp. 2048C2057.

[52] L. Chen, H. Zhang, J. Xiao, L. Nie, J. Shao, W. Liu, and T. S. Chua, "Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5659C5667.

[53] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang, "Residual attention network for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3156C3164.

[54] X. Wang, K. Yu, C. Dong, and C. C. Loy, "Recovering realistic texture in image super-resolution by deep spatial feature transform," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 606C615.

[55] G. Huang, Z. Liu, L. V. D. Maaten, and K. Q. Weinberger. (2016). "Densely connected convolutional networks." [Online]. Available: https://arxiv.org/abs/1608.06993

[56] T. Tong, G. Li, X. Liu, and Q. Gao. Image Super-Resolution Using Dense Skip Connections. In: *IEEE Int. Con. Comput. Vis. (ICCV)*. IEEE Computer Society, Oct. 2017, pp. 4799-4807.

[57] X. Cheng, X. Li, Y. Tai, J. Yang. (2018). "Sesr: single image super resolution with recursive squeeze and excitation networks." [Online]. Available: https://arxiv.org/abs/1801.10319

[58] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp.770-778.

[59] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Oct. 2016, pp.630-645.

[60] Y. Wang, J. Shen, J. Zhang. (2018). "Deep bi-dense networks for image super-resolution," [Online]. Available: https://arxiv.org/abs/1810.04873

[61] V. Nair, G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *ICML*, Jun. 2010, pp. 807C814.

[62] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Trans. Image Proces.*, vol. 13, no. 4, pp. 600-612, Apr. 2004.

[63] X. Glorot, Y. Bengio. "Understanding the difficulty of training deep feedforward neural networks." in *Proc. AISTATS10*, May 2010, vol. 9, pp. 249C256.

[64] D. P. Kingma and J. L. Ba. (2014). "Adam: a method for stochastic optimization." [Online]. Available: https://arxiv.org/abs/1412.6980v9

[65] J. V. Manjon, P. A. Coupe, V. Fonov, C. D. Louis, and M. Robles, "Non-local MRI upsampling," *Med. Image Anal.*, vol. 14, no. 6, pp. 784-792, Dec. 2010.

[66] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278-2324, Nov. 1998.