

Enhanced generative adversarial network for 3D brain MRI super-resolution

Jiancong Wang^{*1}, Yuhua Chen^{*3}, Yifan Wu¹, Jianbo Shi², and James Gee¹

¹*Penn Image Computing and Science Laboratory, University of Pennsylvania, Philadelphia, PA 19104, USA*

²*Department of Computer and Information Science, University of Pennsylvania, Philadelphia, PA 19104, USA*

{*jiancong.wang@pennmedicine., yfwu@seas., jshi@seas., gee@*}*upenn.edu*

³*Department of Bioengineering, University of California, Los Angeles, CA 90095, USA*
chyuhua@ucla.edu

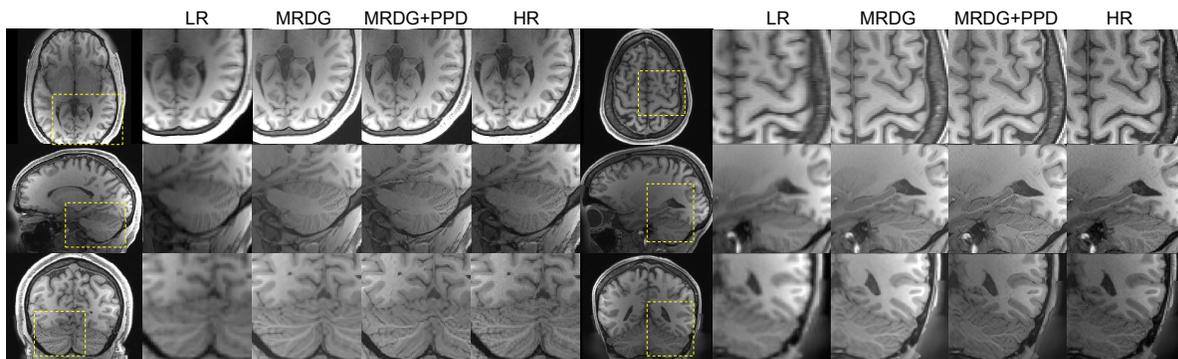


Figure 1: Left to right, LR, MRDG, MRDG with PPD, HR

Abstract

Single image super-resolution (SISR) reconstruction for magnetic resonance imaging (MRI) has generated significant interest because of its potential to not only speed up imaging but to improve quantitative processing and analysis of available image data. Generative Adversarial Networks (GAN) have proven to perform well in image recovery tasks. In this work, we followed the GAN framework and developed a generator coupled with discriminator to tackle the task of 3D SISR on T1 brain MRI images. We developed a novel 3D memory-efficient residual-dense block generator (MRDG) that achieves state-of-the-art performance in terms of SSIM (Structural Similarity), PSNR (Peak Signal to Noise Ratio) and NRMSE (Normalized Root Mean Squared Error) metrics. We also designed a pyramid pooling discriminator (PPD) to recover details on different size scales simultaneously. Finally, we introduced model blending, a simple and computational efficient method to balance between image and texture

quality in the final output, to the task of SISR on 3D images.

1. Introduction

High spatial resolution (HR) structural MRI provides fine-grain anatomical information and makes accurate quantitative image analysis feasible. However, it requires long scan time. Long scan time introduces acquisition challenges [26] like motion artifacts and coil interference. It also potentially limits clinical accessibility in situations where only short scans are feasible. For example, patients in critical condition relying on external life maintenance machines could not stay for a prolong period in the MRI machine. Single image super-resolution (SISR) reconstruction for MRI has generated interest because superresolution images (SR) resembling the actual HR images may potentially be derived from k-space sub-sampling low resolution images (LR), which requires only a fraction of the scan time of HR.

Deep convolutional neural (CNN) network has achieved

*equal contribution.

great success in image superresolution task on 2D natural images [7, 8, 19, 29, 32–34]. The CNN SISR pipeline usually consists of a single generator network that takes in the degraded/down-sampled LR images as input and directly outputs the SR images. A photo-metric loss is calculated between the SR images and the ground truth HR images and drives the network to recover image details.

[16] has shown that deeper generator networks achieve better superresolution. However, CT and MRI, two typical medical image modalities, come in 3D volume. Given memory limit on modern GPU, it is usually infeasible to feed the full 3D volume in full resolution into the generator but to feed patch by patch. Hence one major challenge in medical image SISR is to design a memory efficient generator that can take into a patch including enough spatial context while maintaining reasonable depth. Another challenge that few articles has explored, is the design of the discriminator. We have observed that the patch-GAN discriminator [15] commonly used in image generation/ image style transfer tasks has limitations on recovering details on different scales. Last but not least, unlike 2D natural images which is usually consumed only by human readers, medical images are read by clinical practitioners but also may need to go through downstream automatic pipeline for quantitative measurement, which may prefer different tuning than human reader e.g. high PSNR versus realistic looking texture. Being able to trade-off freely between high PSNR and texture-realism with a single model is of great practicality.

In this work, we follow the GAN framework for 3D image superresolution. We developed and tested our algorithm on k-space sub-sampled low resolution T1 MRI images of the brain, as in [3], and the same network structure is applicable to other 3D image modalities. Our major contributions are follows: 1) we developed a new 3D generator based on memory efficient implementation of residual-dense connections, termed memory-efficient residual-dense generator (MRDG), that significantly outperforms previous state-of-the-art in terms of both traditional metrics and a clinical-related measurement; 2) we developed a fully-convolutional pyramid pooling discriminator (PPD) that is capable of capturing details on various scales simultaneously and outperforms individual patch-GAN discriminators; 3) we first introduced a computationally efficient blending approach to trade off between PSNR and GAN oriented models for 3D medical images.

2. Related work

2.1. 2D natural image SISR by CNN approach

In CNN approaches for SISR, the network architecture of generator has been an active area of research. As a pioneer work, Dong et al. propose SRCNN [6, 7] and variant FSRCNN [8] to learn the mapping from LR to HR images

in an end-to-end fashion, proving that a multi-layer convolutional neural network (CNN) is capable of achieving superior performance against previous works. Various network architectures for SISR have been proposed. Ledig et al. [19] introduced a residual network [12] for SISR and Zhang et al. [38] extends the idea with the residual in residual connection. Kim et al. [16] proposed a very deep network with residual learning. Lai et al. [18] proposed a Laplacian pyramid structure. [17, 22, 29] separately introduced recursive structure. [30] adapted the densely connected network. First and second order channel wise re-scaling [5, 37], deep back projection [11] and residual dense network [38] have also been explored. Recently, combining the residual-in-residual connection and dense connection, Wang et al. [34] proposed a hybrid of residual and dense connections, termed residual-in-residual dense block, to replace the basic residual block in [19]. Anwar et al. [1] proposed a dense residual Laplacian module and achieved state-of-the-art performance on multiple 2D natural image data sets. However, the networks in [34] and [1] are memory intensive and not suitable for 3D medical image SISR. We designed and implemented a memory efficient 3D generator following the inspiration from [34] and [38]. Our proposed model outperformed 3D variants of [37] and [1].

Parallel to the active development of generators, efforts have also been put into adversarial training. [9] invented generative adversarial network (GAN) and shown that fine grain texture and structure realism can be generated/recovered with supervision from a discriminator. [19] introduced GAN in SISR task and GAN training has become popular in SISR task [33, 34, 34, 37, 38]. Since there is no convergence guarantee for GAN, stability of GAN has been an issue. Arjovsky et al. [2] pointed out the popular KL divergence and cross entropy loss are not suitable for GAN training and proposed a training procedure to approximate the Wasserstein distance. The training procedure was further improved by Gulrajani et al. [10] and variant has been proposed by [35]. Meanwhile, Isola et al. [15] proposed patch-GAN, a fully convolutional discriminator. In this work, our discriminator adopts a fully convolutional design and we used the WGAN-GP [10] training procedure to stabilize GAN training.

2.2. 3D medical image SISR

Many solutions using 3D CNN have been proposed for the medical imaging SISR problem [3, 4, 24, 39]. Sanchez et al. [27] adapted the standard super-resolution GAN (SR-GAN) [19] framework for brain image super-resolution. Zhao et al. [39] developed an 3D super-resolution residual network (EDSR) [23] for axial slice super-resolution on T2 brain image. Chen et al. [31] used a reversible GAN for chest-CT superresolution. Chen et al. [3] proposed a multi-layer DenseNet [13] based network for fast and ef-

efficient inference and WGAN-GP training [10] for realistic texture recovery and achieved state-of-the-art result on T1 MRI brain superresolution. We showed that our proposed model outperformed [3,4].

3. Methods

The overall pipeline is illustrated in Fig.2. The generator is trained with L_1 loss to obtain our PSNR oriented model, and is fine-tuned using a pyramid pooling discriminator to be our GAN oriented model. The former model is optimized with respect to conventional similarity metrics but ignores textural fidelity, while the latter GAN model recovers the realistic texture details but at the expense of potentially introducing artifact. A model blending parameter α permits free trade off between the two models in the final generated image.

We do not add the GAN loss right at the beginning of training because it produce images with unwanted artifacts and worse image quality for our use case. We have conducted both types of training using the same discriminator and same GAN loss weight, and show a comparison in the Fig.2 in supplemental material and provide an intuitive explanation within the figure caption.

3.1. Memory efficient residual dense generator (MRDG)

For the generator, inspired by [34] and [38], we developed a 3D memory-efficient residual-in-residual-dense generator(MRDG). MRDG consists of 3D convolutions and receives 3D patches as input and directly output 3D patches. The overall architecture design are shown in Fig. 3.a), resembling a 3D SRResNet [19]. Instead of residual blocks, the MRDG features a memory-efficient residual-in-residual-dense blocks (MRDB). The proposed MRDG block is shown in Figs. 3.b). It adopts a residual-in-residual structure, where residual learning is used in two levels, similar to [36]. In the low level residual connection, residual scaling [23, 28] is applied with scaling factor β between 0 and 1. Similar to [3, 34, 38], we used 3D dense block within the bottom level residual connection. Dense blocks [13] effectively increases the network capacity while being computationally efficient by reusing previous features.

The MRDB is memory-efficient in two aspects. First, we implemented the dense accumulation within each MRDB with gradient check-pointing [25] for memory efficiency (shown by the "Check" block in Figs. 3.b). Namely, for computationally cheap operations like concatenation, instead of storing the intermediate results, the intermediates are discarded during forward propagation and re-calculated during back-propagation when needed. During our experiment, we found that gradient check-pointing reduces memory consumption by roughly 30% with negligible increase in training time. Second, within each MRDB the feature

Models	n_f	k	n_c
MRDG16	16	12	4
MRDG32	32	12	4
MRDG48	48	12	4
MRDG64	64	12	6

Table 1: Structures of the 4 MRDG models.

is residual summed; therefore, the number of features is unchanged throughout all blocks. Compared to mDCSRN proposed in [3,4], which densely accumulates features globally and grows wider as the network gets deeper, the MRDG remains narrow through out the whole depth. We could therefore feed patches to large enough to ensure spatial context while keeping the network deep. To be specific, our largest and best-performing model MRDG64 could fit into a GTX 1080 Ti during training only with the aforementioned memory optimization. Therefore memory-efficiency indeed translates to better performance in the 3D case.

The MRDG is completely determined by the number of residual features n_f , the dense block growth rate k , and the number of MRDB blocks n_c . We have experimented with 4 model configurations: MRDG 16/32/48/64 and the exact configuration n_f, k, n_c of the model are shown in Table.1. Across all 4 configurations, we did not use batch norm, as we found that the removal of the batch norm layer, similar to the practice in [34], leads to improvement of generated image quality in terms of PSNR, SSIM and NRMSE by roughly 0.5 %. We used a scaling factor $\beta = 0.2$ and activation as leaky rectified linear unit.

For completeness of discussion, we have also experimented further architectures changes and training techniques, which do not further improve performance. A summarizing table of those attempts can be found in the supplemental material.

3.2. Pyramid pooling discriminator

During our experiment, we tested the popular patch GAN discriminator [15] and we observed that the scale of recovered details is related to the depth of patch GAN discriminator. A shallower discriminator with less receptive field, smaller spatial context and higher spatial resolution is better in recovering local texture (1-3 voxels in size), e.g. texture pattern in white matter, whereas a deeper discriminator with larger receptive field, bigger spatial context and lower spatial resolution is better in recovering large edges (4-10 voxels in size), e.g. edges between cerebral spinal fluid (CSF) and gray matter (the dark groove on the image). Such observation is shown in Fig. 7.

To allow the discriminator to capture features at different scales, we designed a pyramid pooling discriminator (PPD) for GAN training. As shown in Fig. 4, the discriminator

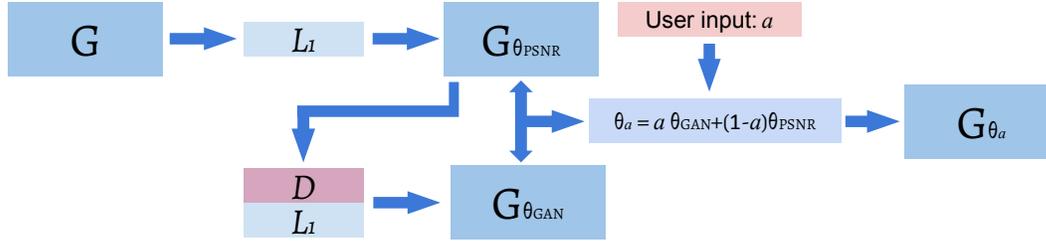


Figure 2: Model training and blending pipeline. α is a user input linear blending weight from 0 to 1. θ refers to the collection of parameter in the generator.

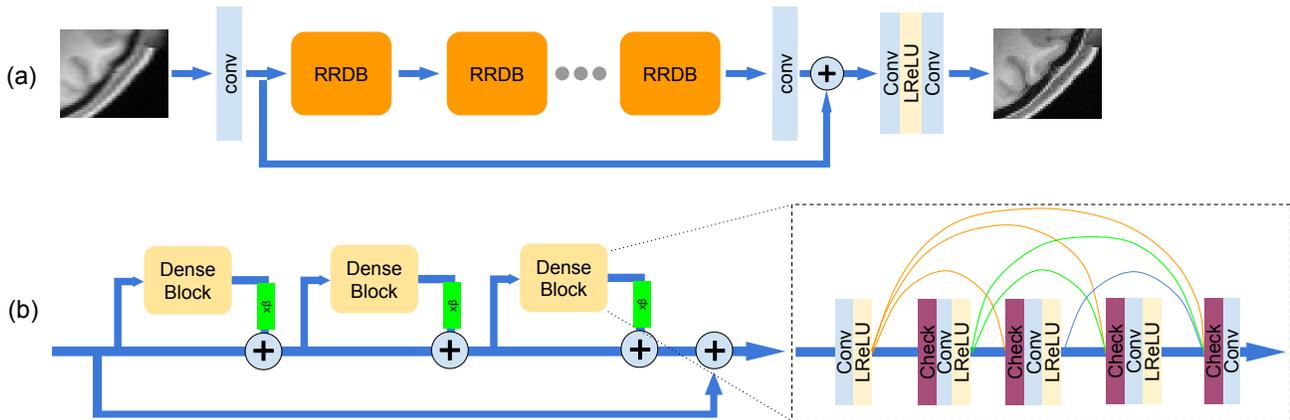


Figure 3: Architecture of the proposed MRDB and the full MRDG network. Like SRResNet [15], MRDG consists of a global residual connection and consecutive basic blocks, except residual blocks are replaced by MRDB. Within each MRDB, three consecutive memory efficient dense blocks (M-Dense Block) are chained by a scaled residual connection and a block level residual connection. Each concatenation within the M-Dense block is gradient-checkpointed and this saves roughly 30% of the graphic memory and allows us to deepen the network correspondingly, without introducing severe training time penalty. We also reported that the removal of batch norm layer improves the SSIM/PSNR/NRMSE metrics by roughly 0.5%.

consists of a pyramid feature extraction path (upper path) which subsequently down-samples the input images by a factor of 2 with striated convolutions. Instead of outputting only the feature at the final coarsest scale like [15], intermediate features at different scale are up-sampled by trilinear interpolation to the original input size and concatenated together (lower path) after passing through additional convolutions. The concatenated feature passes through a final convolution and results in a cost volume. We found that the PPD is able to simultaneously capture details on different scales. On 2D natural images, an alternative could be training multiple independent patch-GAN discriminators with various depths, as in [14, 20], which however, is intensive in graphic memory consumption and not suitable for 3D images.

For completeness of discussion, we have also tried a discriminator ending with two dense layers, similar to one in [3, 19]. We found that the large number of parameters in

two dense layers results in numerical instability and slow convergence. We therefore stick to fully-convolutional discriminators.

WGAN-GP training [10] was used in our implementation for added stability. Specifically, the discriminator ends with plain convolution rather than sigmoid layer; it has instance norm rather than batch norm; a gradient penalty is added to the discriminator for randomly interpolated $\{I_{hr}, I_{sr}\}$ (high resolution ground truth, super-resolution) inputs, and the discriminator is scheduled to always run ahead of the generator.

The loss functions L_G and L_D for the generator and discriminator, respectively, are defined as follows:

$$L_G = L_1(I_{sr}, I_{hr}) + \lambda_D D(I_{sr}), \quad (1)$$

where L_1 is the element-wise L_1 loss, D is the discriminator and λ_D being the weighting factor between the two

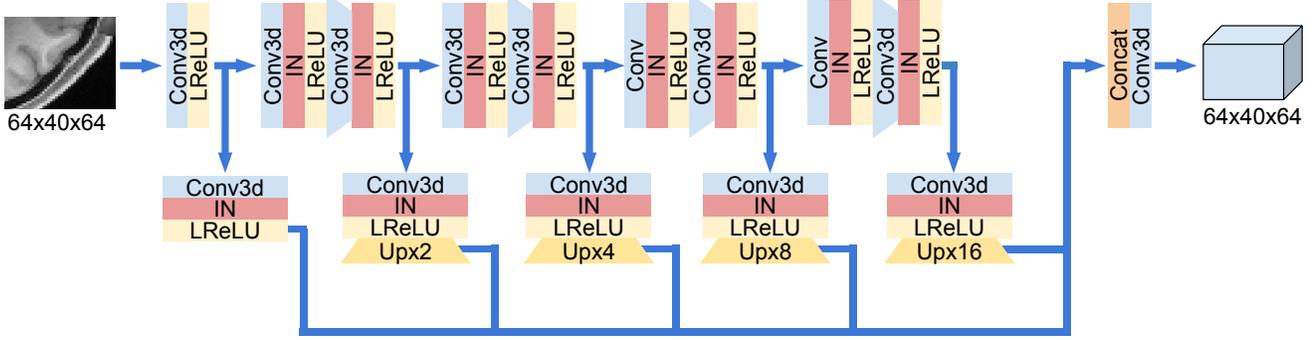


Figure 4: Architecture of the proposed pyramid pooling discriminator. It consists of a pyramid path with subsequent down sampling like [15]. To allow the discriminator capture difference between SR and HR images at different scales, we adopted a pyramid pooling structure similar to [40]. Intermediate features on different resolutions are upsampled to the size of input by trilinear interpolation, concatenated and then passed through a final 1x1x1 convolution.

terms;

$$L_D = D(I_{hr}) - D(I_{sr}) + \lambda_g \|\dot{D}(\gamma I_{sr} + (1-\gamma)I_{hr})\|_2, \quad (2)$$

where \dot{D} is the derivative of the discriminator, λ_g is the weighting factor added on the gradient penalty, and γ is a random number drawn from a uniform distribution, $\gamma \sim (0, 1)$.

3.3. Linear blending of PSNR oriented model and GAN model

One may train multiple models separately to fulfill needs for high PSNR versus realistic texture. The alternative we develop here, following the work in [33, 34], is to blend model through linear interpolation of model parameters for our PSNR and GAN models, as shown in Fig.2. Let θ_G represent parameters of the generator incorporating both PSNR and GAN models, and α , a user defined linear blending weight. We define the blended model as:

$$\theta_G^\alpha = \alpha \theta_G^{PSNR} + (1 - \alpha) \theta_G^{GAN}. \quad (3)$$

Compared to blending model output images, mixing model parameters yields smoother results [34], and, equally important, is also more computational efficient. Blending model parameters requires minor computations compared to actually running the model, and only runs the generator once regardless of the number of model blended, while blending model outputs among different models requires the generator to be run multiple times.

4. Experiment and results

4.1. Data, preprocessing and metrics

Ground truth images were obtained from the Human Connectome Project (HCP). Specifically, it includes 1,113

3D T1 MR images from 1,200 healthy young subjects on Siemens 3T platform. HCP images were downsampled to 1 mm³ resolution using spline interpolation for our SISR experiments. Low resolution versions of these images were created by further downsampling the resolution in coronal and sagittal planes by one half in k-space following exactly the procedure in [3]. In details, the HR image was transformed into k-space by FFT, downgraded in resolution by truncating the outer portion of 3D k-space with a factor of 2x2, and converted back to image space by inverse FFT, then linearly interpolated to the original image size. This mimics the actual imaging and reconstruction process in Siemens MRI machine.

The same number of splits was used as in [3, 4]; specifically, 780 for training, 111 for validation, 111 for evaluation, and 111 for testing. Results are reported on the test set, which was not used in model training or parameter optimization. We used a patch size of 64×40×64 as input due to GPU memory constraints, and cropped 3 voxels around the boundary of the output to avoid discontinuity around edges, resulting in 58×32×58 output patches. The complete output image is assembled by stitching together non-overlapping output patches to avoid blurring caused by averaging patches. We implemented our model in PyTorch 1.0 and trained the model on a workstation with 4 GTX 1080 Ti GPUs.

4.2. Ablation study on model architecture

Quantitatively, we calculate SSIM, PSNR, NRMSE metrics between superresolution output and ground truth HR images and reported the number in Table.2. We also reported the number of parameters and run time for superresolution on single images with a single GTX 1080Ti. Among all 4 configurations, the MRDB48 has performance closed to the largest 64 configuration and has marginally longer

Models	SSIM \uparrow	PSNR \uparrow	NRMSE \downarrow	#param	Time (s)
3D FSRCNN [8]	0.9282 ± 0.0068	33.83 ± 1.0376	0.1138 ± 0.0046	64,893	7.4
3D SRResNet [19]	0.9399 ± 0.0068	34.06 ± 0.9775	0.1104 ± 0.0055	2,004,620	80.3
3D RCAN [37]	0.9542 ± 0.0063	36.35 ± 1.0068	0.08509 ± 0.0042	15,200,513	129.4
3D DRLN [1]	0.9563 ± 0.0067	36.76 ± 0.9870	0.08112 ± 0.0047	30,315,401	139.8
mDCSRN(b8u4) [3, 4]	0.9485 ± 0.0059	35.38 ± 1.0634	0.0954 ± 0.0042	625,969	23.0
MRDG16 (ours)	0.9573 ± 0.0057	36.73 ± 1.0434	0.08151 ± 0.0041	876,049	13.5
MRDG32 (ours)	0.9601 ± 0.0052	37.20 ± 1.0513	0.07727 ± 0.0041	1,665,121	18.9
MRDG48 (ours)	0.9610 ± 0.0051	37.33 ± 1.0459	0.07622 ± 0.0041	2,647,729	26.6
MRDG64 (ours)	0.9623 ± 0.0050	37.38 ± 1.0645	0.07554 ± 0.0042	6,945,601	53.4

Table 2: SSIM/PSNR/NRMSE (mean and standard deviation), number of parameters and test time on single image for state-of-the-art networks and the 4 MRDG configurations on the test set are reported.

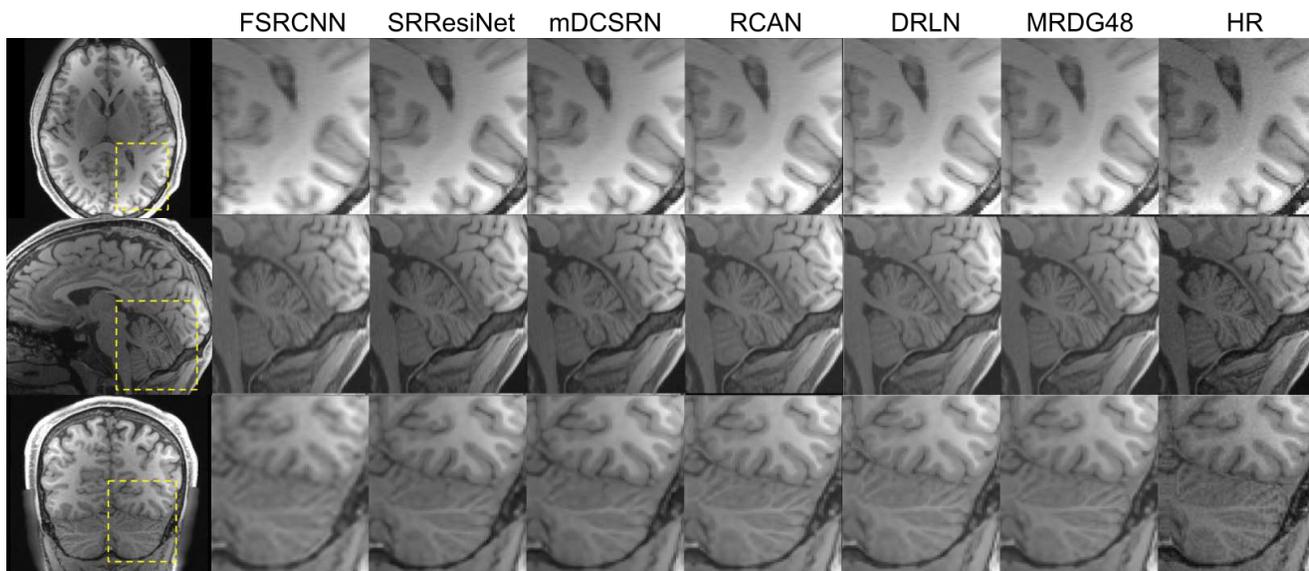


Figure 5: Left to right: SR output from FSRCNN, SRResNet, mDCSRN, RCAN, DRLN, MRDB48 and ground truth HR.

run time than the 32 configurations. We used MRDG48 in the following comparison with state-of-the-art models and the GAN experiment.

4.3. Comparison of MRDG to state-of-art models on brain MRI super resolution

Our proposed MRDG were evaluated against state-of-the-art FSRCNN, SRResNet, mDCSRN, RCAN and DRLN models for SISR reconstruction. The FSRCNN and SRResNet are adapted to 3D directly. For the RCAN and DRLN, we adopted from the official implementations from their authors available on github * and adapted from 2D to 3D by changing the convolution/norm layers to 3D version. We kept the number of feature, the kernel size, the reduction ra-

*<https://github.com/yulunzhang/RCAN>, <https://github.com/saeed-anwar/DRLN>

tio in the squeeze and excitation layer on both RCAN and DRLN completely unchanged. Due to memory limit, we reduces the number of residual block to 8 and number of residual group to 8 (default 10/20 in the 2D cases) in the RCAN, the number of DRLM module to 2 in the DRLN to 2 (default 6 in the 2D cases). The resulting 3D RCAN and 3D DRLN are both largest variants we could fit on a single GTX 1080 Ti GPU. For the mDCSRN, we used the largest and best-performing ever reported b8u4 configuration, 8 dense blocks with 4 dense layers within each block. Qualitatively, examples are illustrated in Fig. 5. More examples are available in the supplemental material. Visually the output from MRDG is sharpest and closest to HR images. Quantitatively, we calculate SSIM, PSNR, NRMSE between SR and HR images and reported them in Table 2. MRDG achieves the best performance. We performed two-

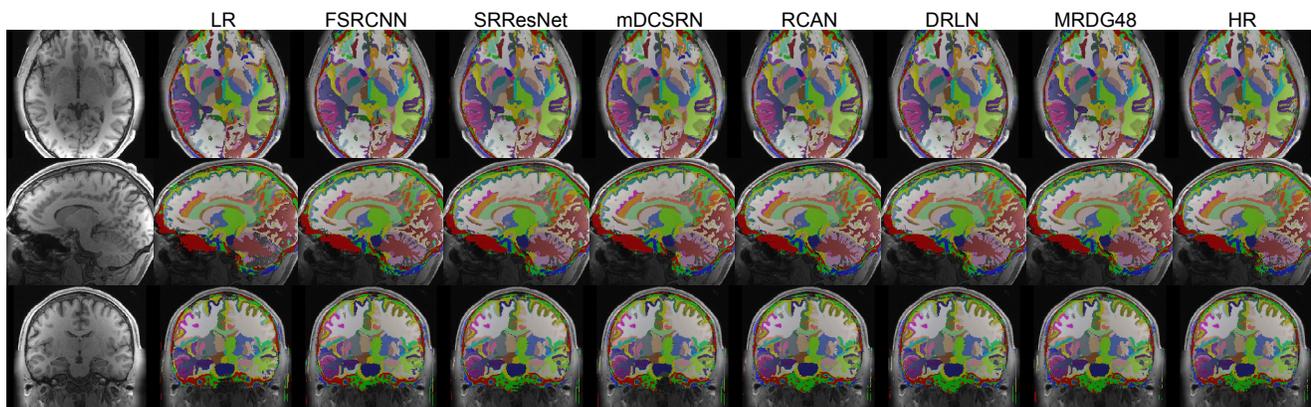


Figure 6: Brain segmentation from HighRes3DNet [21] on LR, SR from different models and ground truth HR image. The brain segmentation on SR image is much closer to the one on HR images than on the LR images.

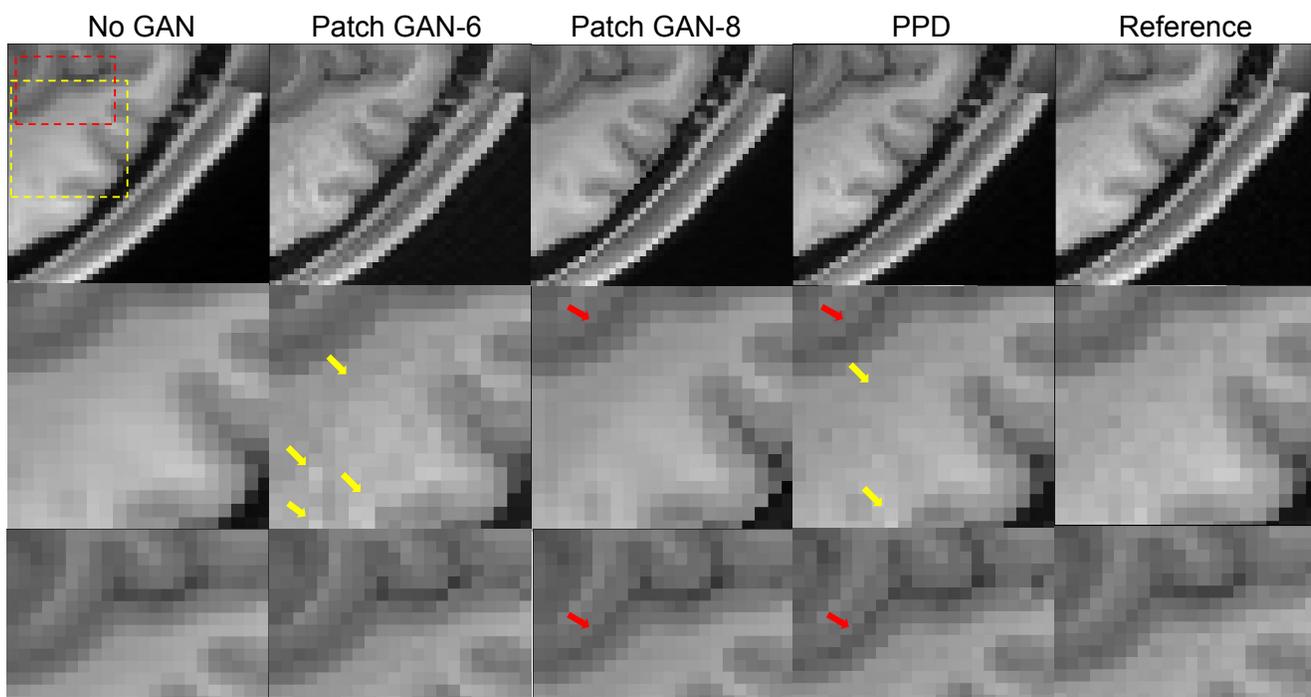


Figure 7: Left to right: MRDG without GAN training, with a 6-layer/ 8-layer patch GAN discriminators, with the proposed pyramid pooling discriminator and the HR images. The second row zooms in on the white matter (the white matter texture is pointed by yellow arrow). The third row zooms in on the CSF-gray matter boundary (the dark groove pointed by the red arrow).

tailed pair-wise t-test and for all metrics, $p < 1.1e - 53$. We also reported the total number of parameters and run time on single image. The MRDG is slower than mDCSRN or FSRCNN but quicker than the SRResNet, 3D RCAN and 3D DRLN.

4.4. Benefit to subsequent brain segmentation

Similarity metrics like SSIM/PSNR/NRMSE are not domain specific for brain MRI image superresolution. For a more clinical-related evaluation, we conducted segmentation of different anatomical regions of the brain with a pre-trained brain segmentation network HighRes3DNet [21], on the LR, the HR image and SR outputs from all networks.

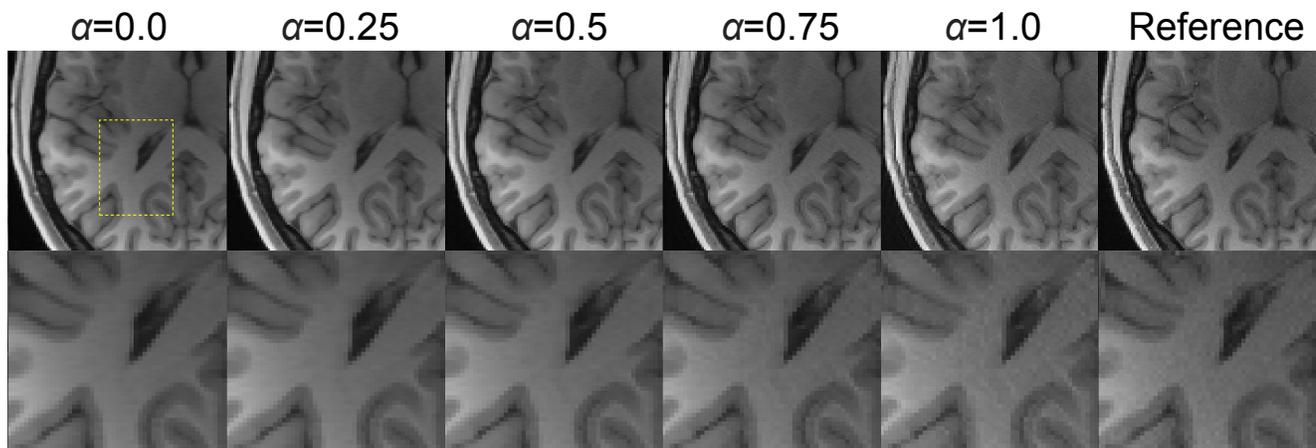


Figure 8: Sample image appearance as a function of blending between GAN oriented model ($\alpha = 1$) and PSNR oriented model ($\alpha = 0$), compared with ground truth.

Models	IoU
3D FSRCNN [8]	0.8677 ± 0.0068
3D SRResNet [19]	0.9048 ± 0.0067
3D RCAN [37]	0.9269 ± 0.0063
3D DRLN [1]	0.9318 ± 0.0066
mDCSRN(b8u4) [3, 4]	0.9153 ± 0.0063
MRDG16 (ours)	0.9268 ± 0.0065
MRDG32 (ours)	0.9335 ± 0.0061
MRDG48 (ours)	0.9351 ± 0.0062
MRDG64 (ours)	0.9380 ± 0.0063

Table 3: Intersection over Union (IoU) (mean and std) of segmentations on HR images and segmentations on SR images by HighRes3DNet [21].

Qualitatively, we show the brain segmentation result from LR, HR and SR on Fig. 6. It is clear that the brain segmentation on the superresolution output is much closer to the segmentation on ground truth than the segmentation on low-resolution image.

Quantitatively, we calculated the intersection over union (IoU) between the segmentation on the HR and the SR from all networks and showed the IoU in Table. 3. Our MRDG models perform all other models.

4.5. Comparison of PPD with patch-GAN style discriminators

We have qualitatively compared the performance the pyramid pooling discriminator against the patch GAN style discriminators at two different depths. Patch GAN-6 is a patch GAN discriminator with 6 convolutions, 3 of them being striated convolutions for down-sampling, before the

final convolution block, and vice versa for Patch GAN-8. Detailed illustration of patch-GAN 6/8 structures are available in supplemental material. Fig. 7 shows the superresolution output from MRDG without GAN training, with the patch GAN-6/8 discriminator and with the proposed pyramid pooling discriminator. The second row zooms in on the white matter. The third row zooms in on the CSF-gray matter boundary. The shallower patch GAN-6 leads to aggressive recovery of white matter texture (1-3 voxel in size) but also introduces artifacts. The deeper patch GAN-8 sharpens the CSF-gray matter boundary (4-10 voxels in size). The PPD is able to capture both of the white matter texture and CSF/gray-matter boundary simultaneously and introduces much less artifacts than patch GAN-6. More examples are available in the supplemental material.

4.6. Effect of model blending

Fig. 8 illustrates the effect of model blending on the generated output image. The output varies smoothly with the interpolation factor α , allowing controllable trade off between PSNR model and GAN model. More examples are available in supplemental material.

5. Discussion

In this work, we significantly improved upon the current state-of-the-art of 3D brain MRI super-resolution. First, the 3D memory efficient residual dense generator exhibits superior performance against the state-of-the-art counterparts. Second, the proposed fully convolutional pyramid pooling discriminator recovers brain image details on different size scales simultaneously while a single patch-GAN discriminator fails. Third, we proposed model blending for computationally efficient and smooth trade off between

PSNR/GAN based model.

There are many directions for future work. Model architectures search remains a major focus of our current work. Another open challenge is artifacts in patch GAN reconstructions, and new solution possibilities include improved generator architectures and the incorporation of domain information such as brain image segmentations [32].

References

- [1] S. Anwar and N. Barnes. Densely residual laplacian super-resolution. *arXiv preprint arXiv:1906.12021*, 2019.
- [2] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017.
- [3] Y. Chen, F. Shi, A. G. Christodoulou, Y. Xie, Z. Zhou, and D. Li. Efficient and accurate MRI super-resolution using a generative adversarial network and 3D multi-Level densely connected network. In *MICCAI*, 2018.
- [4] Y. Chen, Y. Xie, Z. Zhou, F. Shi, A. G. Christodoulou, and D. Li. Brain mri super resolution using 3d deep densely connected neural networks. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pages 739–742. IEEE, 2018.
- [5] T. Dai, J. Cai, Y. Zhang, S.-T. Xia, and L. Zhang. Second-order attention network for single image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11065–11074, 2019.
- [6] C. Dong, C. C. Loy, K. He, and X. Tang. Learning a deep convolutional network for image super-resolution. In *European conference on computer vision*, pages 184–199. Springer, 2014.
- [7] C. Dong, C. C. Loy, K. He, and X. Tang. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 38(2):295–307, 2016.
- [8] C. Dong, C. C. Loy, and X. Tang. Accelerating the super-resolution convolutional neural network. In *ECCV*, 2016.
- [9] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.
- [10] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville. Improved training of wasserstein gans. In *NeurIPS*, 2017.
- [11] M. Haris, G. Shakhnarovich, and N. Ukita. Deep back-projection networks for super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1664–1673, 2018.
- [12] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.
- [13] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In *CVPR*, 2017.
- [14] X. Huang, M.-Y. Liu, S. Belongie, and J. Kautz. Multimodal unsupervised image-to-image translation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 172–189, 2018.
- [15] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017.
- [16] J. Kim, J. Kwon Lee, and K. Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1646–1654, 2016.
- [17] J. Kim, J. Kwon Lee, and K. Mu Lee. Deeply-recursive convolutional network for image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1637–1645, 2016.
- [18] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 624–632, 2017.
- [19] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *CVPR*, 2017.
- [20] H.-Y. Lee, H.-Y. Tseng, J.-B. Huang, M. Singh, and M.-H. Yang. Diverse image-to-image translation via disentangled representations. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 35–51, 2018.
- [21] W. Li, G. Wang, L. Fidon, S. Ourselin, M. J. Cardoso, and T. Vercauteren. On the compactness, efficiency, and representation of 3D convolutional networks: brain parcellation as a pretext task. In *IPMI*, 2017.
- [22] Z. Li, J. Yang, Z. Liu, X. Yang, G. Jeon, and W. Wu. Feedback Network for Image Super-Resolution. *arXiv e-prints*, page arXiv:1903.09814, Mar 2019.
- [23] B. Lim, S. Son, H. Kim, S. Nah, and K. Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 136–144, 2017.
- [24] C.-H. Pham, A. Ducournau, R. Fablet, and F. Rousseau. Brain MRI super-resolution using deep 3D convolutional networks. In *ISBI*, 2017.
- [25] G. Pleiss, D. Chen, G. Huang, T. Li, L. van der Maaten, and K. Q. Weinberger. Memory-efficient implementation of densenets. *arXiv:1707.06990*, 2017.
- [26] E. Plenge, D. H. Poot, M. Bernsen, G. Kotek, G. Houston, P. Wielopolski, L. van der Weerd, W. J. Niessen, and E. Meijering. Super-resolution methods in MRI: Can they improve the trade-off between resolution, signal-to-noise ratio, and acquisition time? *Magnetic resonance in medicine*, 68(6):1983–1993, 2012.
- [27] I. Sánchez and V. Vilaplana. Brain MRI super-resolution using 3D generative adversarial networks. *arXiv preprint arXiv:1812.11440*, 2018.
- [28] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.
- [29] Y. Tai, J. Yang, and X. Liu. Image super-resolution via deep recursive residual network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3147–3155, 2017.

- [30] Y. Tai, J. Yang, X. Liu, and C. Xu. Memnet: A persistent memory network for image restoration. In *Proceedings of the IEEE international conference on computer vision*, pages 4539–4547, 2017.
- [31] T. van der Ouderdaa and D. E. Worrall. Reversible gans for memory-efficient chest ct super-resolution and domain-adaptation in 3d. In *MIDL 2019*.
- [32] X. Wang, K. Yu, C. Dong, and C. Change Loy. Recovering realistic texture in image super-resolution by deep spatial feature transform. In *CVPR*, 2018.
- [33] X. Wang, K. Yu, C. Dong, X. Tang, and C. C. Loy. Deep network interpolation for continuous imagery effect transition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1692–1701, 2019.
- [34] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, and C. C. Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *ECCV*, 2018.
- [35] X. Wei, B. Gong, Z. Liu, W. Lu, and L. Wang. Improving the improved training of wasserstein gans: A consistency term and its dual effect. *arXiv preprint arXiv:1803.01541*, 2018.
- [36] K. Zhang, M. Sun, T. X. Han, X. Yuan, L. Guo, and T. Liu. Residual networks of residual networks: Multilevel residual networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(6):1303–1314, 2017.
- [37] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu. Image super-resolution using very deep residual channel attention networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 286–301, 2018.
- [38] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu. Residual dense network for image super-resolution. In *CVPR*, 2018.
- [39] C. Zhao, A. Carass, B. E. Dewey, and J. L. Prince. Self super-resolution for magnetic resonance images using deep networks. In *ISBI*, 2018.
- [40] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017.