Medical Image Modality Synthesis And Resolution Enhancement Based On Machine Learning Techniques

by

Can Zhao

A dissertation submitted to Johns Hopkins University

in conformity with the requirements for the degree of

Doctor of Philosophy

Baltimore, Maryland

January 2021

Abstract

To achieve satisfactory performance from automatic medical image analysis algorithms such as registration or segmentation, medical imaging data with the desired modality/contrast and high isotropic resolution are preferred, yet they are not always available. We addressed this problem in this thesis using 1) image modality synthesis and 2) resolution enhancement.

The first contribution of this thesis is computed tomography (CT)-to-magnetic resonance imaging (MRI) image synthesis method, which was developed to provide MRI when CT is the only modality that is acquired. The main challenges are that CT has poor contrast as well as high noise in soft tissues and that the CT-to-MR mapping is highly nonlinear. To overcome these challenges, we developed a convolutional neural network (CNN) which is a modified U-net. With this deep network for synthesis, we developed the first segmentation method that provides detailed grey matter anatomical labels on CT neuroimages using synthetic MRI.

The second contribution is a method for resolution enhancement for a common type of acquisition in clinical and research practice, one in which there is high resolution (HR) in the in-plane directions and low resolution (LR) in the through-plane direction. The challenge of improving the through-plane

resolution for such acquisitions is that the state-of-art convolutional neural network (CNN)-based super-resolution methods are sometimes not applicable due to lack of external LR/HR paired training data. To address this challenge, we developed a self super-resolution algorithm called SMORE and its iterative version called iSMORE, which are CNN-based yet do not require LR/HR paired training data other than the subject image itself. SMORE/iSMORE create training data from the HR in-plane slices of the subject image itself, then train and apply CNNs to through-plane slices to improve spatial resolution and remove aliasing. In this thesis, we perform SMORE/iSMORE on multiple simulated and real datasets to demonstrate their accuracy and generalizability. Also, SMORE as a preprocessing step is shown to improve segmentation accuracy.

In summary, CT-to-MR synthesis, SMORE, and iSMORE were demonstrated in this thesis to be effective preprocessing algorithms for visual quality and other automatic medical image analysis such as registration or segmentation.

Primary Reader: Dr. Jerry L. Prince

Secondary Readers: Dr. John I. Goutsias and Dr. Trac D. Tran

Acknowledgments

First, I would like to express my deep and sincere gratitude to my advisor Dr. Prince for the continuous support of my Ph.D study and research. I had very limited background in medical image processing before I met him and he helped me to establish the knowledge background, methodology of research, self-confidence, insights on research problems and, most importantly, the passion to open black boxes and find the fundamental issues behind research problems. He is my mentor in many respects, teaching with what he does, not only what he says. I cannot feel luckier for having such a respectable scholar as my advisor. I will remember his guidance throughout my life.

Apart from Dr. Prince, I would also express my sincere gratitude to Dr. Tran, Dr. Goutsias, and Dr. Pham for giving the encouragement and sharing insightful suggestions for my research proposal and dissertation. It would not have been possible to conduct this research without their precious support. Thanks also goes to Dr. Lee, Dr. Saria, Dr. Reiter, Dr. Khudanpur and Dr. Naiman for helping me build knowledge foundation during qualifying and GBO preparation. I am also pleased to say thank you to my previous advisors Dr. Yu from Tsinghua University and Dr. Khurgin for introducing me to research. It is very lucky for me that every advisor I met is so wonderful. They

all really mean a lot to me.

I would always remember my fellow IACL labmates for the fun time we spent together, the inspiring discussion we had, and the support from them on research and life. Special thanks to Aaron Carass for all the help he gave on my research. I would like to thank my friends from Johns Hopkins University. Special thanks to Muhan Shao and Dan Zhu for being my beautiful bridesmaids. Thanks also go to my doctors, nutritionists, and counselors from Hopkins for helping me fight my eating disorder.

I also thank my family and my parents for their support, especially my mom. Her experience of overcoming the barriers to girls' education and being a lifelong learner gives me endless power. Thanks also to my dog and cats for their unconditional (or biscuit-motivated) support and love.

Finally, my deepest gratitude goes to my best friend, my greatest support, my sunshine, and my significant other: Bowen Li. This dissertation is dedicated to you.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

In this chapter, we introduce the background knowledge of our two main topics: image modality synthesis and super-resolution. We first introduce the basic concepts of image modality/contrast, resolution, and noise. Then we introduce some background of image modality synthesis and super-resolution. Finally, the contributions and organization of this thesis are summarized.

1.1 Introduction to image contrast, resolution, and noise

As machine learning, including deep learning techniques develop, automatic medical image analysis based on these techniques has increasingly gained interest in clinical and research applications. A big challenge in these applications is the diversity of three medical image properties: contrast, resolution, and noise level.

In order to better understand this challenge, we define these properties and describe their effects on automatic medical image analysis as below.

1.1.1 Image contrast

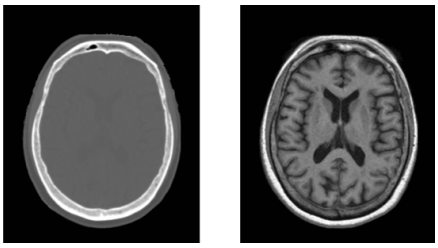
Image contrast describes the difference in the brightness of the object and other objects within the same field of view. There are two types of contrast that we discuss in medical imaging: intensity contrast and tissue contrast, described as below.

- If an object has intensity I and the background has intensity I_b, then the intensity contrast is $\frac{I-I_b}{I_b}$.
- Tissue contrast describes the intensity difference between two types of tissues rather than object and background.

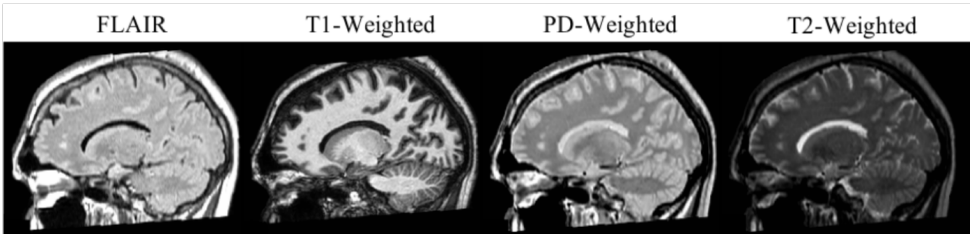
In the rest of this thesis, when we mention contrast, we always mean tissue contrast.

In medical imaging, image contrast is strongly related to image modality, which is a scanning technique to visualize the human body for diagnostic and treatment monitoring purposes. They are strongly related because the differences in scanning methods determine the acquired image contrast. However, there is a small number of image modalities and various of possible contrasts. Diversity in contrast during scanning is due to several reasons:

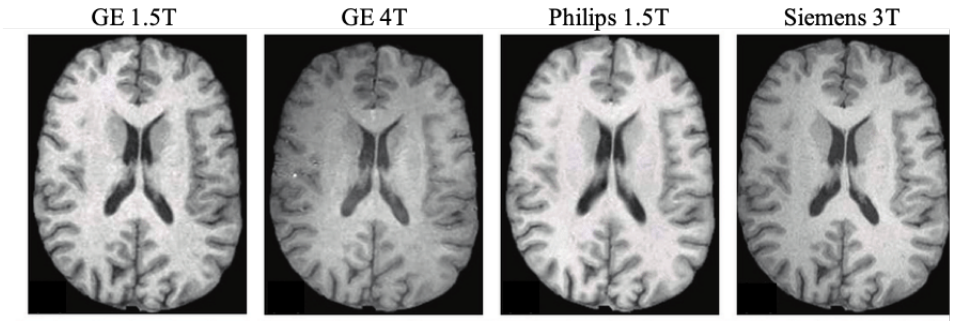
- First, images with different modalities have different contrasts since they respond to different underlying physical parameters of the body tissues. For example, computed tomography (CT) has a large contrast between bones and soft tissues, while magnetic resonance imaging (MRI) has better contrast between different types of soft tissues. Figure 1.1a shows



(a) CT (left) and MR (right) images



(b) MRI with different pulse sequences [2].



(c) T1-w SPGR MRI from different scanners [3].

Figure 1.1: Images with different modalities that have different contrasts

a pair of CT/MR images. They show the same brain, but they visually look very different.

- Second, when images are from the same modality, their contrasts are sometimes different since the scanning parameters may be different. For example, CT images with different X-ray energy distributions can have slightly different contrasts. Also, MR images with different pulse sequences[1] have different contrasts, as shown in Figure 1.1b [2]. Here, T2 Flair, T1-weighted (T1-w), PD-weighted (PD-w), and T2-weighted (T2-w) MR images of the same brain look very different.

- Even when the pulse sequences are the same, the MR scanners can be different, which leads to different contrasts. Figure 1.1c [3] shows T1-w MR images from different manufacturers and with different magnetic fields. Although they are all T1-w MRI with spoiled gradient recalled (SPGR) pulse sequence, their contrasts are different.

1.1.2 Image resolution

There are two kinds of resolution definitions commonly used in medical imaging: digital resolution and spatial resolution. Digital resolution is the voxel or pixel separation of the digital image. Spatial resolution, on the other hand, describes the ability to "resolve", or separate, small details. Images with high spatial resolution are desired as they provide more details. Figure 1.2 shows an example of (a) a brain MR image with low spatial and digital

[1] An MRI pulse sequence is a programmed set of changing magnetic field gradients and radio frequency pulses. A more detailed introduction can be found in Bitar et al. [4]

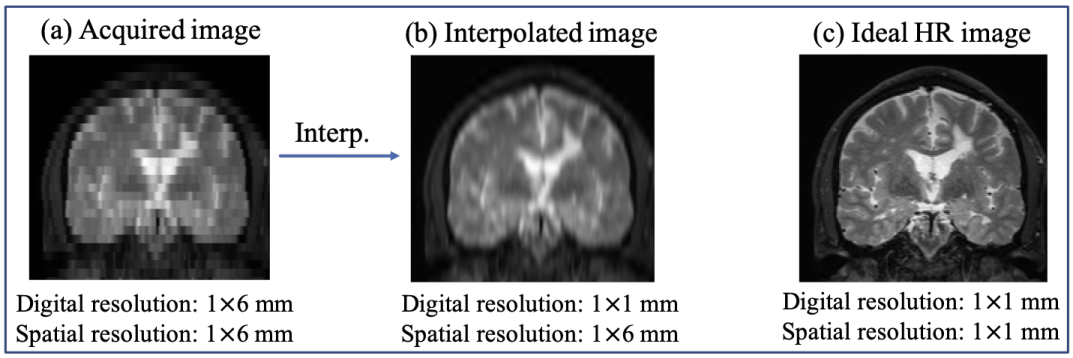


Figure 1.2: Images with different spatial or digital resolution: **(a)** a brain image with low spatial and digital resolution; **(b)** an interpolated image with high digital resolution but still low spatial resolution; **(c)** an ideal image with high spatial and digital resolution

resolution, (b) an interpolated image with high digital resolution but still low spatial resolution, and (c) an ideal image with high spatial and digital resolution.

These two definitions are related. High digital resolution is a necessary but not sufficient condition for images with high spatial resolution. To obtain high digital resolution, image interpolation can be used. To obtain high spatial resolution, however, one requires more advanced imaging devices, a trade-off between resolution and noise, or carrying out the challenging post-processing method called super-resolution, which is introduced in Section 1.3.

1.1.3 Image noise

The third property of importance in medical imaging is noise. Noise is a fundamental characteristic that is present, to some extent, in all images. It reduces the visibility of some structures and objects, especially those with relatively low contrast. If the contrast and resolution remain the same, low noise

level images are always desired. Reducing noise during image acquisition, however, involves a compromise with patient exposure for CT. For MRI, low noise level and high spatial resolution are both desired while both involve long acquisition times, which is not desired. Therefore, for MRI, there is a trade-off among spatial resolution, noise level, and acquisition time.

1.2 Introduction to image synthesis

1.2.1 Motivation

Most automatic medical image analysis tools should be applied to images with similar contrasts, since image contrast affects performance. Some examples are described below.

- For machine learning or deep learning based segmentation/classification, the segmentor or classifier is trained on the image features computed from training images. To correctly apply the trained segmentor or classifier to test images, the contrasts of training and test images must be similar. People have found that image synthesis helps to normalize contrasts in MRI, and thus can improve segmentation accuracy [2].

- For mono-modal image registration[2], the commonly-used loss function cross-correlation (CC) assumes that moving and target images have the same contrast after some linear scaling of intensities. Mean squared error (MSE) and sum of squared distance (SSD), on the other hand,

[2]There are two types of intensity-based registration: mono-modal registration that deals with images with similar contrasts, and multi-modal registration that deals with images with different contrasts.

assume that the contrasts are the same. These assumptions are not always true in reality, especially for MRI data.

- The performance of multi-modal image registration is usually worse than mono-modal registration. Researchers have found that image synthesis can convert multi-modal registration into mono-modal registration, and thus improve performance [5].

Sometimes, acquired images do not have the desired contrasts. In this case, image synthesis algorithms can help to fill the gap. Image synthesis, also called image-to-image translation, is a process that creates a target image that depicts the same anatomy but a different contrast with an acquired source image. People have developed image synthesis methods for MR to CT [6, 1, 7, 8, 9], for between MRI images with contrasts [2, 5, 10, 11], and for other applications [12, 13, 14, 15].

1.2.2 Overview of four synthesis methods

To explore medical image synthesis, it is important to understand how images are acquired. As shown in Figure 1.3, body anatomy determines the underlying physical parameters. Different types of tissues have different x-ray attenuation coefficients, ultrasound reflection coefficients, proton densities (PD), T1 relaxation times, T2 relaxation times, etc. These physical parameters determine image intensity and contrast following different physical rules for different image modalities and different acquisition parameters.

With this understanding of image acquisition, we start to explore medical image synthesis. Suppose we have a source image with contrast A. The goal

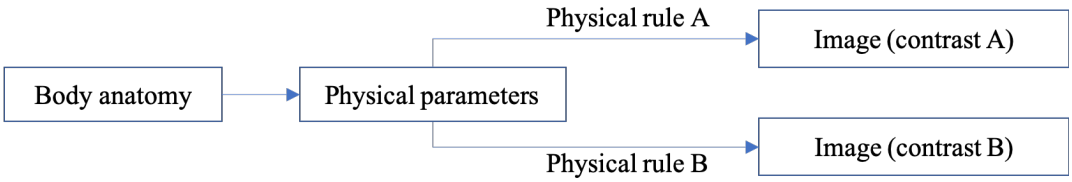


Figure 1.3: From body anatomy to image contrast

Type	Method
Physical-based	Source image (contrast A) → Physical parameters → Synthesized Image (contrast B)
Classification-based	Source Image (contrast A) → Classified body anatomy → Synthesized Image intensities for each class (contrast B)
Registration-based	Atlas Images (contrast B) → Synthesized Image (contrast B)
Example-based	Source Image (contrast A) → Synthesized Image (contrast B)

Table 1.1: Overview of four types of synthesis methods

of image synthesis is to produce an image that has the same anatomy as the source image, but has contrast B. Intuitively, we can keep the anatomy of the source image unchanged and try to convert contrast A into contrast B. Or we can find an atlas image with contrast B but different anatomy and try to align the anatomy of atlas image to the source image. Based on these two ideas, there are four common types of image synthesis methods, summarized in Table 1.1 and discussed below.

- **Physical-based methods**: Medical image synthesis can be done following physical rules and be produced mathematically [16, 17]. One can first estimate the underlying physical parameters from a source image, and then use the estimated parameters to compute a synthesized image.

  A fundamental defect of physical-based methods is that the mapping

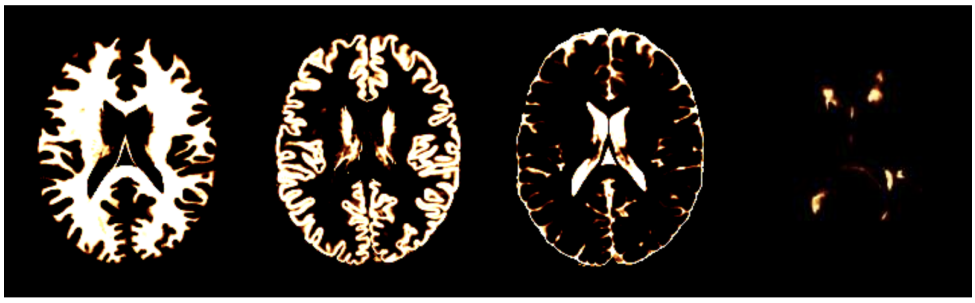
from image intensities to the physical parameters is sometimes not a one-to-one mapping. Therefore, it can be difficult or inaccurate to estimate physical parameters for each pixel/voxel.

- **Classification-based methods**: Estimating physical parameters for each pixel/voxel can be difficult or inaccurate. Fortunately, the physical parameters for pixels/voxels that belong to the same tissue class are often very close. Taking advantage of this fact, classification-based synthesis is a relaxed version of the physical-based methods [18, 19]. These methods first segment the source image to estimate the tissue class for each pixel/voxel, shown in Figure 1.4a. Then for each class, they synthesize the image intensities for pixels/voxels that belong to this class and then combine them into the final image, shown in Figure 1.4b.

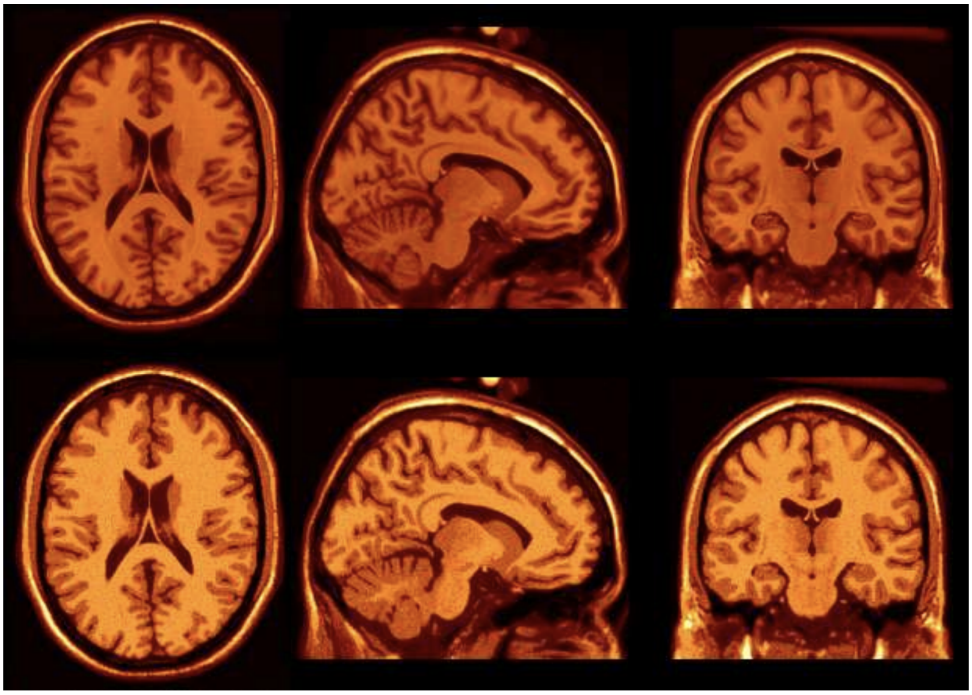
  Considering the fact that sometimes a voxel contains more than one type of tissue, soft segmentation[3] can be used to make the results more smooth. This is at the core of BrainWeb [18], an important MR neuroimage simulation package that has been used extensively in the evaluation of neuroimage processing algorithms.

  The accuracy of classification-based image synthesis strongly depends on the quality of image segmentation. Also, the physical models that estimate image intensities from the estimated physical parameters may not be accurate. In fact, synthesized images from physical-based and classification-based synthesis sometimes look unrealistic.

[3] Soft segmentation, also called membership function, estimates the percentage of several tissue types for each pixel/voxel, rather than a single tissue type.



(a) From Brainweb [18]: example of fuzzy tissue classes from the phantom: (L to R)white matter, gray matter, CSF, MS lesions



(b) From Brainweb [18]: real (top) and simulated (bottom) MRI-s

Figure 1.4: Example of classification-based image synthesis method: Brainweb [18].

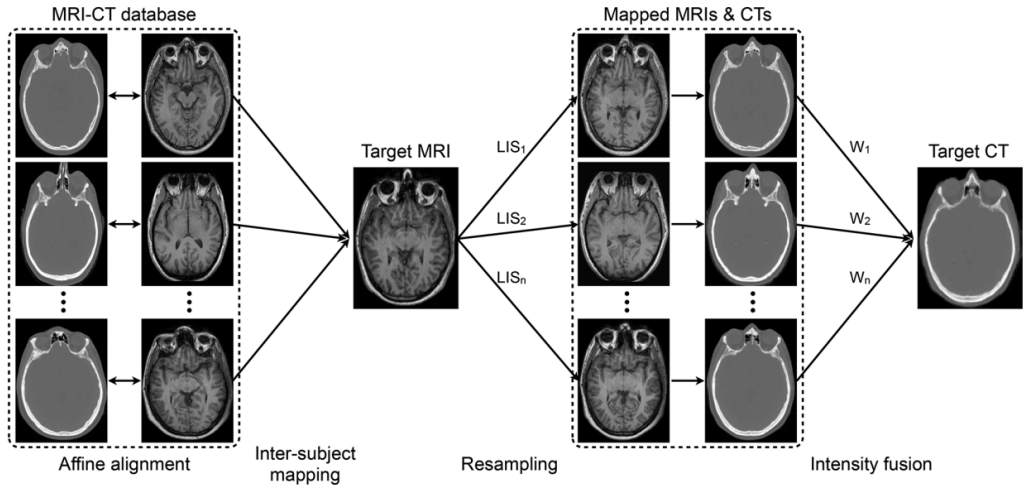


Figure 1.5: Example of registration-based image synthesis method from Burgos et al. [1]: CT synthesis diagram for a given MRI image. All the MRIs in the atlas database are registered to the target MRI (source image). The CTs in the atlas database are then mapped using the same transformation to the target MRI. A local image similarity measure (LIS) between the mapped and target MRIs is converted to weights(W) to reconstruct the target CT.

- **Registration-based methods**: Another way of thinking about image synthesis is to focus on changing the anatomy rather than the contrast [1, 20, 21, 22, 6, 23]. Registration-based methods require at least one atlas image of contrast B or one pair of atlas image of contrast A and B. With these atlas images, we can use deformable registration to align their anatomies to the source image. Figure 1.5 shows an example of MR-to-CT synthesis. The atlas is the 'MRI-CT database' on the left. The source image is the 'Target MRI' in the middle. If an atlas image contain the exactly the same anatomical parts with the same topology as the source image, and the deformable registration is done perfectly, then the warped atlas image will have contrast B and same anatomy as the source image; thus, it will be a perfectly synthesized image.

The advantage of this type of method is that the contrast is guaranteed to be correct. However, it is impossible to perform a perfect deformable registration with current technology. Therefore, the anatomy of the synthesized image is often inaccurate. Also, the computation time can be hours since deformable registration is time consuming.

- **Example-based methods**: Physical-based and classification-based synthesis produce results with accurate anatomy, but inaccurate contrast. Registration-based synthesis produce results with accurate contrast, but inaccurate anatomy. The fourth type of method, example-based synthesis, attempts to consider the accuracy of both anatomy and contrast. We introduce this approach in the following section.

1.2.3 Example-based Synthesis

Example-based synthesis does not consider the underlying physical parameters in medical images. Instead, it directly applies a mapping on the source image, also called a regressor in machine learning, from contrast A to contrast B [24, 10, 2, 9, 25, 26, 27]. To learn the mapping, a training set is required. Usually, this training set contains pairs of images, where each pair has the same anatomy, one with contrast A and the other with contrast B. An intuitive idea is to compute the joint histogram of the two contrasts from training images, and to model this mapping as voxel-wised intensity transformation. However, the mapping between contrasts is rarely monotonic. Depending on the underlying anatomy, the same intensity with contrast A could be mapped

to multiple possible intensities with contrast B. Fortunately, context information in medical imaging is strongly related to body anatomy. Therefore,taking context information into consideration, can help improve the accuracy of the trained regressor.

Traditional machine learning techniques involve designing hand-crafted features to describe the context information and choosing a reasonable regressor that can learn the mapping. This designed feature together with the regressor should be able to distinguish the underlying anatomy and model the mapping correctly. The most straightforward feature is an image patch, which is found effective in practice. For example, the synthesis method MIMECS [24] uses patches as features and sparse reconstruction as the regressor. Another method REPLICA [10] also uses patches as features but uses a random forest as regressor. Compared with sparse reconstruction, the random forest takes much less time and provides a better result.

Recently, deep learning has become the state-of-art method for many applications in image processing. The first contribution of this thesis, which is introduced in Chapter 3, also uses deep learning as the synthesis method. In particular, it uses convolutional neural network (CNN). We introduce the basic conceptions of CNNs in Chapter 2.

1.3 Introduction to super-resolution

1.3.1 Motivation

Low spatial resolution degrades subsequent image analysis. For example:

- researchers have found that low spatial resolution degrades the performance of segmentation [28, 29];
- registration performs best when moving and target images are both of high resolution (HR). This is discussed in Appendix B [30]).

To improve the spatial resolution, people have developed super-resolution (SR) methods including single-image SR and multi-image SR techniques in computer vision and medical imaging. The difference between the two is that, for each subject, single-image SR tries to construct a high-resolution (HR) image from a single low-resolution (LR) image, whereas multi-image SR tries to constract a HR image from multiple LR images [31, 32, 33, 34]. In this thesis, we only discuss single-image SR.

1.3.2 Overview of three super-resolution methods

There are three common types of single-image SR methods:

- **Single-image deconvolution**: Single-image deconvolution is resolution enhancement from only one acquired image. A typical observation model for this technique assumes that the acquired LR image g comes from $g = h * f + \eta$, where f is the HR image, h is the point spread function (PSF) of the blur kernel, and η is an unknown additive noise. The goal of single-image deconvolution is to estimate the HR image f given the acquired LR image g and a known PSF h.[4] This process does not involve any other image besides the subject image g itself, which

[4] A more difficult case is when the PSF is unknown and must also be estimated. This problem is known as blind deconvolution.

is its main advantage. From the observation model $g = h * f + \eta$, f can be estimated by minimizing a loss function. For example, f can be estimated using the L2 loss by

$$\hat{f} = \arg\min_{f} ||g - h * f||^2. \tag{1.1}$$

However, this approach is extremely ill-posed and therefore is highly susceptible to increasing noise. Also, it is often difficult to precisely know the PSF h, which can produce severe artifacts. To address these problems, a regularization term $\mathcal{R}(f)$ is often added to stabilize the solution as follows,

$$\hat{f} = \arg\min_{f} ||g - h * f||^2 + \mathcal{R}(f). \tag{1.2}$$

$\mathcal{R}(f)$ is usually designed from prior knowledge to balance the contribution of smoothness and data fidelity terms [35]. Although there has been much effort for single-image deconvolution, the performance is generally worse than the state-of-art example-based SR algorithms.

- **Example-based SR**: State-of-the-art SR algorithms are example-based, requiring LR/HR paired training data with contrasts and resolutions that closely match the subject data. Such an algorithm learns a mapping from an LR image to an HR image from paired training data and then applies the learned mapping to the subject LR image. Many of these approaches—especially those using CNNs—have reported good results in computer vision [36, 37, 38].

- **SSR**: Example-based SR algorithms perform well in computer vision

applications. Unfortunately, these apporoaches have a major disadvantage that limit their use in medical imaging—paired LR/HR training data are often unavailable in medical imaging. Such training data requires subjects to remain stationary for two acquisitions to avoid motion artifacts, and one acquisition must be HR, which can take a long time and be uncomfortable. Also, collected training data is only useful where their contrast is the same, since the accuracy of the trained SR algorithm degrades for subject images with a different contrast, which is common in MRI as we discussed in Section 1.1. In such cases, example-based SR algorithms that do not require external training data are desirable. In this dissertation, we refer to such example-based SR algorithms as self-supervised super-resolution (SSR) algorithms, which are discussed in the next section.

1.3.3 Self-supervised super-resolution

Without external paired training images, example-based SR methods must find other sources to learn LR to HR mapping. There are three approaches in literature:

- **Self-similarrity:** The self-similarity approach first degrades the subject LR image g using the PSF h to obtain the further degraded LR image $q = h * g$. It then uses q and g as paired training data to learn the mapping from q to g. Finally this mapping is applied to g to estimate f [39]. The underlying assumption is that the mapping from q to g is similar to the mapping from g to f. It may sound plausible at first glance;

but g is LR and does not contain some of the HR features that we expect in f. Thus, the mapping learned from q to g may not be able to restore some HR features in f.

- **Intermodality priors:** The second approach, called brain hallucination, targets a certain MRI acquisition protocol [40, 41] using intermodality priors. In brain MRI acquisition, both an LR T2-weighted image and an HR T1-weighted image are acquired. Brain hallucination takes HR information from HR T1-w MRI to improve the resolution of LR T2-w MR image. The underlying assumption is that the HR features in T1-w MRI are similar to the features in T2-w MRI. To make this assumption valid, the features must be carefully designed to be modality invariant, and it may not be possible to verify this in every case.

- **Elongated voxels:** The third approach targets a common type of MRI acquisition that has HR in the in-plane slices and LR in the through-plane direction. The images that are degraded to create paired training data in this approach are the HR in-plane slices. The learned mapping is then applied to LR through-plane slices. In this way, the learned mapping contains HR features, has the same contrast, and thus avoids the concerns of the other two approaches. This is the approach that this thesis develops in Chapter 4. Since such acquisitions are very common, as long as the performance is good, this approach can have great utility in medical imaging.

In this dissertation, our SSR algortihms are based on elongated voxels. In the remainder of this dissertation, SSR always refers to SSR using elongated

voxels.

1.4 Dissertation Overview

1.4.1 Contributions

There are two main contributions in this dissertation.

- **CT-to-MRI synthesis** The first contribution is a CNN-based CT-to-MRI image synthesis method, which was developed to provide MRI when CT is the only modality that is acquired.
- **Self-supervised Super-resolution (SSR)** The second contribution includes two SSR algorithms, SMORE and its iterative version iSMORE, which improve the through-plane resolution with a common type of MRI acquisition that has HR for in-plane slices and LR along the through-plane direction. We applied SMORE on various of MRI datasets, and applied iSMORE on both MRI and two-photon fluorescence microscopy, which demonstrate their generalizability.

In addition, a super-voxel and random forest based CT-to-MRI image synthesis method is presented in the Appendix A. We also performed a theoretical analysis of the effects of resolution on image registration, which is presented in the Appendix B.

1.4.2 Organization

- Chapter 2 introduces some basic concepts and insights about convolutional neural networks (CNNs), as they are the main regressors used in

this thesis.

- Chapter 3 discusses our CNN-based CT-to-MR image synthesis algorithm using material from [42].
- Chapter 4 describes our SSR algorithm SMORE for MR images acquired with 3D and 2D protocols using material from [43, 44].
- Chapter 5 demonstrates the application of SMORE on four different MRI datasets to show its generalizability using material from [45].
- Chapter 6 describes the iterative framework iSMORE as well as its application on two-photon fluorescence microscopy using material from [46].
- Chapter 7 includes a conclusion and discussion.
- Appendix A describes a random forest and classification-based image synthesis algorithm for CT-to-MRI synthesis using material from [47].
- Appendix B includes our theoretical work about the effects of resolution on image registration using material from [30].

Chapter 2

Background on Convolutional Neural Networks (CNNs)

In this chapter, we introduce some background on convolutional neural networks (CNNs), which is the machine learning method we use in this thesis. We start with fully connected neural networks, and then introduce the layers in CNNs. We will see that CNN is a sparse fully connected neural network. Both of fully connected neural networks and CNNs are called deep networks. Finally, the U-net and the ResNet are introduced as they are the two network architectures we use in this dissertation.

2.1 Basics of Fully Connected Neural Network

In this section, we first introduce the layers in a fully connected neural network, and then discuss how this structure forms a nonlinear mapping. In contrast to many deep learning tutorials, we try to explain the non-linearity of fully connected neural network and give a intuitive sense of some observations such as overfitting and adversarial attacks.

2.1.1 Layers

A layer can be considered as a function applied on an $N \times 1$ input vector $\mathbf{x}$, denoted as $f(\mathbf{x})$. Deep networks consist of multiple layers $f_i(\mathbf{x}), i = 1, ..., N$, which can be connected in several ways, some of them are described as below.

- **Sequential:** When two layers f_1 and f_2 are sequentially connected, the output is $f_2(f_1(\mathbf{x}))$.
- **Concatenate:** When two layers f_1 and f_2 are concatenated, the output is $\begin{bmatrix} f_1(\mathbf{x}) \\ f_2(\mathbf{x}) \end{bmatrix}$.
- **Summation:** When two layers f_1 and f_2 are added together, the output is $f_1(\mathbf{x}) + f_2(\mathbf{x})$.
- **Element-wise multiplication:** This is when two layers f_1 and f_2 are multiplied element-wise.

Let us introduce the basic single layers first. Basic layers used in fully connected neural networks include fully connected layers, activation layers, dropout layers, and batch normalization (BN) layers. The definitions, motivations, trainable parameters, and usage of these layers are introduced below. $\mathbf{x}$ denotes the one-dimensional input feature vector. Each element x of $\mathbf{x}$ is called a neuron.

- **Fully connected layer:** With the simple form $f(\mathbf{x}) = W\mathbf{x} + \mathbf{b}$, a fully connected layer applies an affine transform to the input $\mathbf{x}$, where the weight matrix W and bias vector $\mathbf{b}$ are parameters learned from training data. When the output $f(\mathbf{x})$ used for classification is a scalar, this single

fully connected layer works just as a linear classifier support vector machine (SVM). When multiple fully connected layers are sequentially connected and applied on input **x**, it is still a linear mapping.

- **Activation layer:** To perform a nonlinear mapping, a nonlinearity must be introduced into the network; this is usually achieved by adding an activation layer after the fully connected layer. A brief explanation on how activation layers bring in the nonlinearity is introduced in Section 2.1.2. Popular activation layers include the sigmoid function $f(x) = 1 + 1/(1 + e^x)$ and the ReLU function $f(x) = \max(0, x)$ [48]. They are applied to the feature vector **x** element-wise. Most activation layers, except for some more recent ones such as PReLU [49], do not contain trainable parameters.

- **Dropout layer:** Dropout [50] is a form of regularization which can reduce over-fitting. Fully connected networks usually contain a large number of neurons, which causes the problem of overfitting. Dropout randomly zero-outs some proportion of the neurons during training; i.e., for input **x**, a mask vector with the same size as **x** is Bernoulli-sampled and multiplied by **x** element-wise.

  Dropout is applied in order to prevent the network from depending on a sparse collection of powerful neurons, and therefore forces the learned features to be more representative and robust. The disadvantage is that it increases training time. Dropout should be turned off when making predictions, unless it is used to estimate uncertainty. It does not contain trainable parameters, but instead requires a manually-set

dropout rate. The probability of retaining a unit is recommended to be between 0.4–0.8 [50].

- **Batch normalization (BN) layer:** When training a network, the training dataset is usually split into small mini-batches and an update of the trainable parameters is performed for each batch. The number of training samples in one batch is called the batch size. When the batch size is larger than one, adding a BN layer [51] between a fully connected layer and an activation layer helps the network to converge faster during training. It computes the mean and variance of inputs $\mathbf{x}$, and shifts and scales $\mathbf{x}$ to zero-mean and unit variance over mini-batches. We denote the resultant batch normalized inputs using the vector $\hat{\mathbf{x}}$. $\hat{\mathbf{x}}$ is then scaled and shifted based on two trainable parameters γ and β. The output of a learnt BN layer is $\mathbf{y} = \gamma\hat{\mathbf{x}} + \beta$.

  When using a BN layer, one thing we would like to note is that the training and inference modes behave differently in deep learning tools like Tensorflow and PyTorch. By default, during training the mean and variance is computed based on each mini-batch, while during inference they are computed using the tracked mean and variance computed on the whole training dataset. In the inference mode, if specified by the user, this default setting can be changed to computing on each mini-batch.

  It has been proved that the success of BN has little to do with reducing the so-called "internal covariate shift", though it was the original motivation of BN [51]. Instead, BN "improves smoothness of optimization landscape", which "induces a more predictive and stable behavior of

the gradients", allowing for larger learning rates and faster training [52]. Specifically, it was demonstrated that the Lipschitz constants[1] of both the loss and gradients (also known as β-smoothness [53]) are reduced. The original BN paper [51] claims that BN reduces the need for dropout, yet many people still use dropout with BN. Some scholars proposed and proved that dropout should be put after BN layers to avoid variance shift [54].

2.1.2 Nonlinearity and Piecewise Linearity

Consider a trained network that only contains a fully connected layer with trained weights $W = \begin{bmatrix} \mathbf{w}_1^T \\ \mathbf{w}_2^T \\ \vdots \\ \mathbf{w}_N^T \end{bmatrix}$ and bias $\mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_N \end{bmatrix}$ followed by a ReLU activation layer. A fully connected layer is linear and ReLU is piecewise linear, so the composition of them is piecewise linear. To understand how the pieces are divided, let us go into detail.

For an input feature vector $\mathbf{x}^1 = [x_1^1, x_2^1, ..., x_M^1]^T$, the output of fully connected layer is $\mathbf{y}^1 = [y_1^1, y_2^1, ..., y_N^1]^T$. The affine transform in a fully connected layer can be written as:

$$\mathbf{y}^1 = W\mathbf{x}^1 + \mathbf{b} = \begin{bmatrix} \mathbf{w}_1^T \\ \mathbf{w}_2^T \\ \vdots \\ \mathbf{w}_N^T \end{bmatrix} \begin{bmatrix} x_1^1 \\ x_2^1 \\ \vdots \\ x_M^1 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_N \end{bmatrix} = \begin{bmatrix} y_1^1 \\ y_2^1 \\ \vdots \\ y_N^1 \end{bmatrix}. \tag{2.1}$$

After the affine transform, the result $\mathbf{y}^1$ is sent to a ReLU layer. If the neuron

[1] Function f is β-Lipschitz if $|f(x_1) - f(x_2)| \leq \beta|x_1 - x_2|$, for all x_1 and x_2. Lipschitz constant of function f is the upper bound of β.

y_2^1 satisfies $y_2^1 < 0$, then the combination of fully connected layer and ReLU can be written as:

$$\text{ReLU}(W\mathbf{x}^1 + \mathbf{b}) = \text{ReLU}(\begin{bmatrix} y_1^1 \\ y_1^2 \\ \vdots \\ y_N^1 \end{bmatrix}) = \begin{bmatrix} y_1^1 \\ 0 \\ \vdots \\ y_N^1 \end{bmatrix} = \begin{bmatrix} \mathbf{w}_1^T \\ \mathbf{0} \\ \vdots \\ \mathbf{w}_N^T \end{bmatrix} \begin{bmatrix} x_1^1 \\ x_2^1 \\ \vdots \\ x_M^1 \end{bmatrix} + \begin{bmatrix} b_1 \\ 0 \\ \vdots \\ b_N \end{bmatrix}. \quad (2.2)$$

Let $W^1 = \begin{bmatrix} \mathbf{w}_1^T \\ \mathbf{0} \\ \vdots \\ \mathbf{w}_N^T \end{bmatrix}$ and $\mathbf{b}^1 = \begin{bmatrix} b_1 \\ 0 \\ \vdots \\ b_N \end{bmatrix}$. Then for $\mathbf{x} = \mathbf{x}^1$,

$$\text{ReLU}(W\mathbf{x} + \mathbf{b}) = W^1\mathbf{x} + \mathbf{b}^1. \quad (2.3)$$

Since both ReLU and $W\mathbf{x} + \mathbf{b}$ are Lipschitz continuous, $\text{ReLU}(W\mathbf{x} + \mathbf{b})$ is also Lipschitz continuous. For $\mathbf{x} = \mathbf{x}^1$, there exist a neighbor U^1 where $\mathbf{x}^1 \in U^1$, such that $\forall \mathbf{x} \in U^1$, and Equation 2.3 holds true.

If we generalize this argument to $\mathbf{x} \in \mathbb{R}^M$, we see that $\mathbb{R}^M$ can be divided into K convex polytopes $\{U^k | k = 1, 2, ..., K\}$, such that

$$\forall \mathbf{x} \in U^k, \text{ReLU}(W\mathbf{x} + \mathbf{b}) = W^k\mathbf{x} + \mathbf{b}^k, \quad (2.4)$$

with W^k being W with some rows zeroed out, and $\mathbf{b}^k$ being $\mathbf{b}$ with the same rows zeroed out. For each k, $W^k\mathbf{x} + \mathbf{b}^k$ is an affine transform. Therefore, the network $\text{ReLU}(W\mathbf{x} + \mathbf{b})$ is piecewise linear.

More generally, fully connected networks with piecewise linear activations [55] subdivide the input space into convex polytopes. Each convex polytope represents a different linear function [56, 57]. Figure 2.1 shows an example of subdivided input space for a three-layer ReLU activated deep

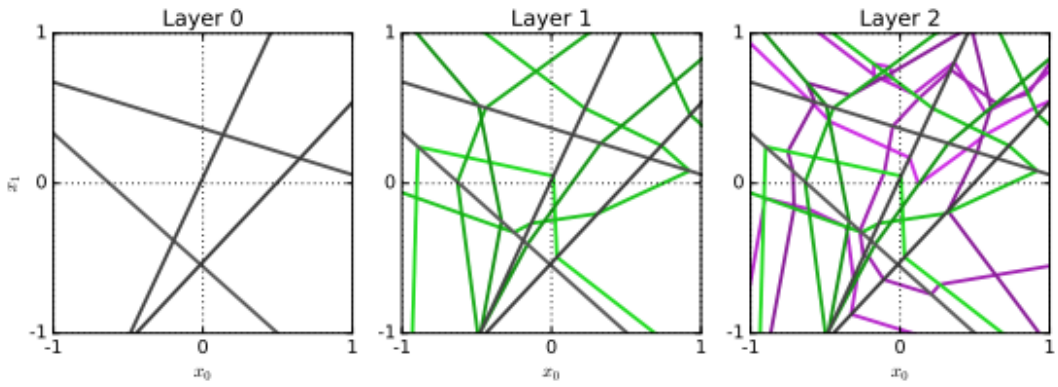


Figure 2.1: An example of subdivided input space for a three layer deep network. Each polytope representing a different linear function. [57]

network.

The input **x** and the corresponding input space for deep network is usually high-dimensional. When we train a deep network, we train both space subdivision and the linear function within each convex polytope. This explains some observations about deep networks. For example,

- **Overfitting:** Overfitting is a phenomenon in which the model performs well on training data but much worse on new test data. When a network is very deep, the input space is divided into very fine polytopes. This makes the deep network behave more like nearest neighbor searching. In this case, if the training data is not enough or not representative, the network will suffer from overfitting.

- **Adversarial attacks:** Recent studies show that deep networks can be vulnerable to subtle perturbations of the inputs [58], which are known as adversarial attacks. Sometimes the perturbations of images can be too small to be observable to human eyes, yet they can lead the trained

model to predict incorrect outputs.

Considering high-dimensional input data, the high-dimensional spaces are so large that most of the training data $\mathbf{x}$ are often concentrated in a very small region known as the manifold. The perturbation for an adversarial attack may be small, but such a perturbation could make the adversarial data leave this manifold. In such cases, the accuracy of prediction is not guaranteed.

- **Data augmentation:** Data augmentation adds random perturbations into the training data. Using this approach, the training data defines a broader manifold, and can make the network more robust to perturbations in the data.

2.2 Basics of Convolutional Neural Networks

Fully connected neural networks require the input and output to be 1D vectors, which do not scale well to 2D and 3D images. If the inputs and outputs are 256×256 images, fully connected neural networks will first flatten them into 65536×1 vectors. The trainable weight matrix W for a single layer that gives an 65536×1 output will have $65536 \times 65536 \approx 4$ billion trainable parameters. A convolutional neural network (CNN) greatly reduces the size of trainable parameters, taking advantage of the fact that the inputs are images rather than vectors.

Basic layers used in CNNs include convolutional layers, activation layers,

dropout layers, BN layers, pooling layers, and upsampling layers. The activation layer has the same definition as in fully connected neural networks. For other layers, the definition, motivation, trainable parameters, and usage are introduced below. Apart from basic dropout and BN layers, we also include some more recent dropout and normalization layers. The input and output of these layers are 2D or 3D images, or feature maps, which can have multiple channels.

- **Convolutional layer**: A convolutional layer consists of a set of learnable filters. The size $K_1 \times K_2$ of each filter is small, usually 3×3 or 5×5. These filters are used to filter the inputs and generate feature maps[2]. Let us first consider the simple case when input images have the size $M \times N$ with one channel and the output also has size $M \times N$ but with C_{out} channels[3]. There are in total C_{out} trainable $K_1 \times K_2$ filters. For each output channel, the result is the input image filtered by a $K_1 \times K_2$ filter plus a trainable bias. There are only $(K_1 \times K_2 + 1) \times C_{\text{out}}$ trainable parameters for this convolutional layer—many fewer parameters than a fully connected layer which has $(M \times N + 1) \times M \times N \times C_{\text{out}}$ trainable parameters.

  For a more general case when inputs have C_{in} channels and outputs have C_{out} channels, there are in total $C_{\text{in}} \times C_{\text{out}}$ trainable $K_1 \times K_2$ filters. In a

[2]Convolutional layer has the name 'convolution', yet the outputs are correlation rather than convolution results between input images and the filters. Correlation and convolution are essentially the same if we flip the filters.

[3]The image size of inputs and outputs are the same, which indicates that some padding is required. Like convolution in standard image processing, the correlation operation that is used in a convolutional layer needs padding if we want the output and input images to have the same size. The default padding for most deep network tools like Keras, Tensorflow, and PyTorch is zero padding.

CNN, this is usually referred to as C_{out} trainable $K_1 \times K_2 \times C_{\text{in}}$ filters. For each output channel, there are C_{in} $K_1 \times K_2$ filters applied to C_{in} $M \times N$ images and result in C_{in} $M \times N$ feature maps. These C_{in} feature maps as well as a trainable bias are added together to form this output channel. This summation operation produces higher level features as the combination of lower level features. For example, summation of edge maps from different orientations can give corner feature maps. The summation of corner feature maps can give more complex shapes. There are only $(K_1 \times K_2 \times C_{\text{in}} + 1) \times C_{\text{out}}$ trainable parameters for this convolutional layer, many fewer than a fully connected layer, which has $(M \times N \times C_{\text{in}} + 1) \times M \times N \times C_{\text{out}}$ trainable parameters.

A convolutional layer is a sparse form of fully connected layer. Suppose the filter K is 2×2, and the 3×3 input image X can be flattened to be a 9×1 vector. The valid correlation of K and X is:

$$K * X = \begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix} * \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix} = \tag{2.5}$$

$$\begin{bmatrix} k_{11}x_{11} + k_{12}x_{12} + k_{21}x_{21} + k_{22}x_{22} & k_{11}x_{12} + k_{12}x_{13} + k_{21}x_{22} + k_{22}x_{23} \\ k_{11}x_{21} + k_{12}x_{22} + k_{21}x_{31} + k_{22}x_{32} & k_{11}x_{22} + k_{12}x_{23} + k_{21}x_{32} + k_{22}x_{33} \end{bmatrix} \tag{2.6}$$

Its flatten version is equivalent to:

$$\begin{bmatrix} k_{11} & k_{12} & 0 & k_{21} & k_{22} & 0 & 0 & 0 & 0 \\ 0 & k_{11} & k_{12} & 0 & k_{21} & k_{22} & 0 & 0 & 0 \\ 0 & 0 & 0 & k_{11} & k_{12} & 0 & k_{21} & k_{22} & 0 \\ 0 & 0 & 0 & 0 & k_{11} & k_{12} & 0 & k_{21} & k_{22} \end{bmatrix} \begin{bmatrix} x_{11} \\ x_{12} \\ x_{13} \\ x_{21} \\ x_{22} \\ x_{23} \\ x_{31} \\ x_{32} \\ x_{33} \end{bmatrix}, \tag{2.7}$$

which is a sparse fully connected layer with repeated weight coefficients.

- **Dropout layer**: In Srivastava et al. [50], exhaustive experimental results show that dropout gives less improvement in CNNs as compared to fully connected networks. However, "the additional gain in performance obtained by adding dropout in the convolutional layers ((loss reduced from) 3.02% to 2.55%) is worth noting." The detailed operation of dropout in a CNN is spatial. For each input, a mask matrix with the same size as input is Bernoulli-sampled and multiplied by the input pixel-wise/voxel-wise. This dropout mechanism has the drawback that it does not consider the fact that the pixels/voxels in feature maps are spatially correlated. Therefore, although there is dropout, information can still flow through convolutional networks [59]. More recent research such as SpatialDropout [60] and DropBlock [59] design more structured dropout mechanisms and claim improvment in performance compared with traditional dropout.

- **Batch normalization (BN) layer and other normalization layers:** Suppose the size of the input image/feature map X of a Batch normalization (BN) layer is $B \times M \times N \times C_{in}$, with B being the batch size. BN in CNNs computes the mean and variance for each channel resulting in C_{in} mean and variance values, each computed over $B \times M \times N$ values [51]. It then shifts and scales each channel of X to zero-mean and unit variance, resulting in normalized inputs $\hat{X}$. Then a trainable affine transform is applied on each channel of $\hat{X}$. In total, there are C_{in} scale and C_{in} bias parameters to train in a BN layer.

  BN has been widely used in CNNs. However, as the network size and data size increases, GPU memory requirements become too large. In such cases, researchers have to reduce batch size B, and for some 3D CNNs, batch size B can be as small as 1. When B is small, BN becomes less effective. Recently, researchers have proposed other normalization methods. For example, Layer normalization [61] computes the mean and variance for each batch, resulting in B mean and variance values, each computed over $M \times N \times C_{in}$ values. Instance normalization [62] computes the mean and variance for each channel and batch, resulting in $B \times C_{in}$ mean and variance values, each computed over $M \times N$ values. Group normalization [63] requires a manually-set group number G and segments C_{in} channels into G groups. It computes the mean and variance for each group, resulting in $B \times G$ mean and variance values, each computed over $M \times N \times C_{in}/G$ values. These normalization methods can be used when B is small.

- **Pooling layer:** A pooling layer is essentially a downsampling layer. It reduces the spatial size of the feature maps to let the network learn both higher resolution and lower resolution information. There are no trainable parameters in pooling layers. Commonly used pooling layers include max pooling and average pooling. For object detection, region of interest (ROI) pooling [64] is used to extract a feature vector of fixed size from a ROI of arbitrary size.

- **Upsampling layer:** As they are the opposite operation to that of pooling, upsampling layers are also widely used. The very basic upsampling layer uses nearest neighbor upsampling, which is the default setting for deep learning tools such as Keras, Tensorflow, and PyTorch. Linear upsampling is also available for these tools. In additional to these two basic upsampling methods, researchers have developed learnable upsampling layers. For example, Shi et al. [65] developed subpixel upsampling, also called pixel shuffling. In order to perform $r_1 \times r_2$ 2D upsampling on a feature map with size $M \times N$, they first used convolutional layers to generate $r_1 \times r_2$ feature maps with size $M \times N$, and then rearranged these feature maps into a single feature map with size $r_1 M \times r_2 N$.

With these layers, researchers have designed numerous of CNN architectures. In the next section, we introduce two types of CNN architectures that are used in this dissertation: U-net and ResNet.

2.3 CNNs used in this thesis: U-net and ResNet

2.3.1 U-net

U-nets [67] mimic the mechanism of image pyramids [68, 69], which are multiresolution image representations. The construction of image pyramids includes two inverse operations: downsampling and upsampling, usually by a factor of 2. Using image pyramids, one can decompose images into information at multiple resolution levels and extract features at multiple resolution levels. The same basic ideas underlie jpeg encoding and many other applications, including the U-net.

The U-net [67] has been one of the most popular CNN architectures in medical imaging. It stems from the first fully convolutional network (FCN) [66].

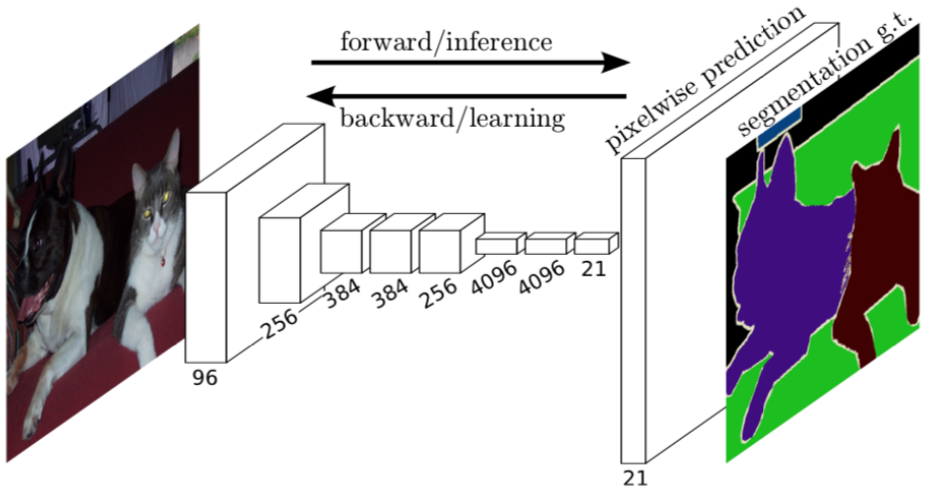


Figure 2.2: FCN architecture [66]. The main contribution of the FCN is that previous CNN architectures use fully connected layer as the final layer to do segmentation. The FCN replaces it with a convolution layer which can make per-pixel prediction more efficient.

In the early CNN architectures, the last layer before activation is fully connected layer. An FCN is a normal CNN, where the last fully connected layer is substituted by another convolution layer. The first FCN [66] consisted of N pooling layers to extract global information and only one upsampling layer at the end to restore high-resolution information. The architecture of this FCN is shown in Figure 2.2.

The FCN outperforms previous networks for segmentation tasks. However, only one upsampling layer at the end is not able to restore localized high-resolution information precisely. The U-net instead consists of N pooling layers in the encoder (the left part of Figure 2.3) but also N upsampling layers in the decoder (the right part of Figure 2.3) to restore high-resolution details gradually. Its architecture is shown in Figure 2.3. Each step in the encoder contains two 3×3 convolutional layers, activated by a rectified linear unit (ReLU), and a 2×2 max pooling operation for downsampling. In the decoder, each step contains a 2×2 upsampling layer followed by a 3×3 convolutional layer and a 3×3 convolutional layer. The two convolutional layers are activated by ReLU. And the final layer is a 1×1 convolutional layer. Another main contribution of the U-net is that it adds N symmetric skip connections from encoder to decoder. Skip connections copy the feature maps in the encoder and concatenate them with the feature maps in the decoder. Adding skip connections can provide more precise high-resolution information to the decoder.

The U-net was originally developed for 2D medical image segmentation in 2015 [67]. Its 3D versions were published in 2016 [70]. Also its ResNet

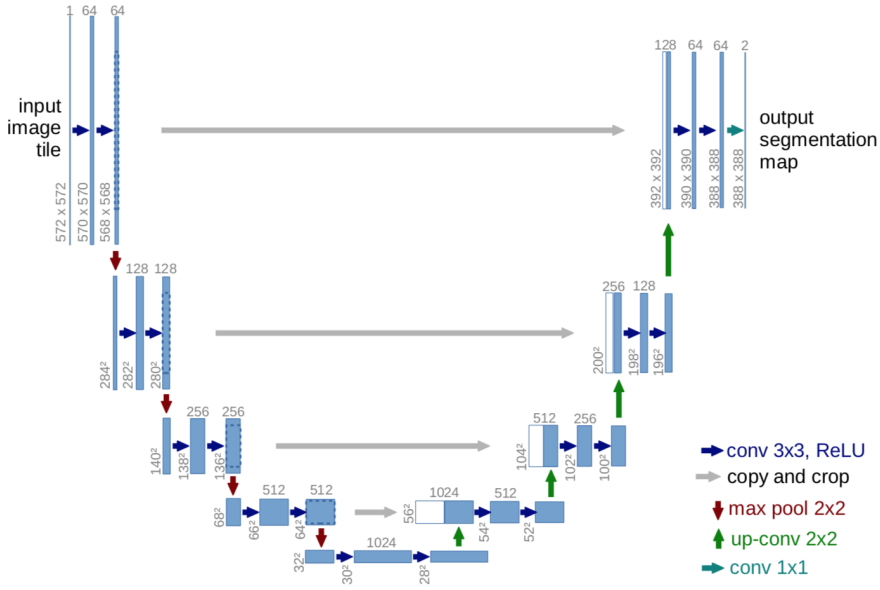


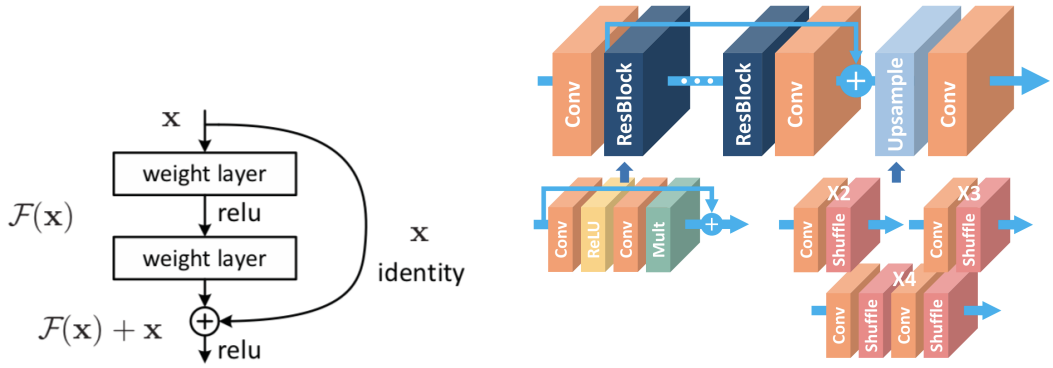
Figure 2.3: U-net architecture [67]. Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower left edge of the box. White boxes represent copied feature maps. The arrows denote the different operations as described in the legend.

variant which replaced convolutional layers with ResNet units (described below) has been widely used [71]. 2D and 3D U-nets have been used in many applications such as segmentation, object detection, classification, registration, image construction/enhancement, image synthesis, etc. [72, 73].

2.3.2 ResNet

The deep residual network (ResNet) was published by He et al. [74] in order to solve the problem of vanishing/exploding gradients during training. It has a simple form, as shown in Figure 2.4a. It adds a shortcut connection to the output of the trainable layer.

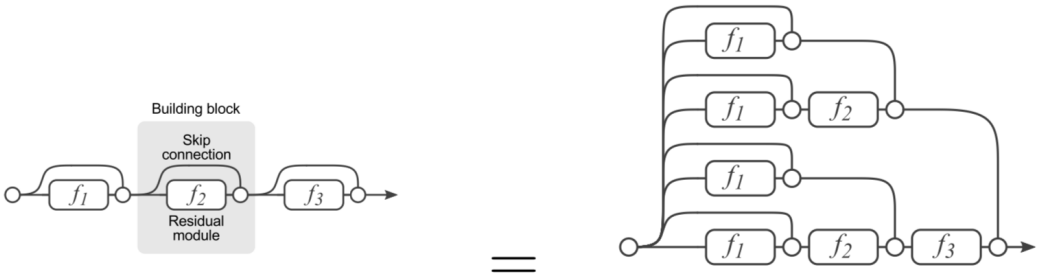
It has been shown by Veil et al. [76] that the success of ResNet is because



(a) ResNet unit [74].

(b) EDSR architecture [75]: The ResBlock is a variant of ResNet unit, in which the green block named 'Mult' means multiply by a fixed scale factor 0.1. The Upsample block is subpixel upsampling [65].

Figure 2.4: ResNet unit and EDSR architecture which was developed using ResNet unit



(a) Conventional 3-block residual network

(b) Unraveled view of (a)

Figure 2.5: Unraveled view of ResNet [76]. Circular nodes represent additions. From this view, it is apparent that residual networks have $O(2^n)$ implicit paths connecting input and output and that adding a block doubles the number of paths.

it behaves like ensembles of relatively shallow networks. Their unraveled view of ResNet is shown in Figure 2.5. Their experiments showed that a ResNet "can be viewed as a collection of many paths, instead of a single ultra deep network". And the gradients during training mainly come from short paths [76, 77]. According to their exploration, boosting is the underlying

theoretical way that ResNet improves deep network performance.

2.3.3 Summary

U-net and ResNet have been very popular networks. But they are used in this dissertation not only because they are popular. The reason of using these two networks are described below.

In Chapter 3, the network architecture we used for CT-to-MR synthesis is a modified U-net. As the mapping between two medical image modalities is dependent on anatomical structures, it makes intuitive sense that any CNN synthesis model designed for medical image modality synthesis should incorporate the ideas of segmentation. Therefore, we have selected the U-net [67] as the basis of our synthesis network; it can achieve state-of-the-art performance for segmentation and preserve high-resolution information by the symmetric skip connections from downsampling blocks to upsampling blocks. Its multi-resolution structure can provide both global and local context information, which is especially helpful for image synthesis.

In Chapters 4–6, we describe our self-supervised super-resolution algorithms, which use a network with the ResNet architecture. The boosting mechanism of ResNet makes it naturally suitable for super-resolution. Many successful super-resolution CNN architectures are ResNets, including VDSR [78], DRRN [79], SRResNet [80], EDSR [75], RCAN [81], and WDSR [82].

The network we used for self-supervised super-resolution in this work is based on the widely used super-resolution network EDSR [75], shown in Figure 2.4b. The EDSR architecture includes convolutional layers, several

ResBlocks, and an Upsample block. The Upsample block is the subpixel upsampling [65] introduced in Sec. 2.2. The ResBlock includes convolutional layers and ReLU activation. The green block named 'Mult' means multiply by a fixed scale factor 0.1.

In summary, this chapter introduced the background on CNNs that are used in the following chapters. In the next chapter, we introduce the first contribution of this dissertation: CT-to-MRI synthesis.

Chapter 3

CT-to-MR synthesis and whole brain segmentation on CT images

3.1 Introduction

In the past few years, the development of CNNs has improved the capability of image synthesis (cf. [83]). Unlike medical image synthesis for other modalities, CT-to-MR synthesis has not been explored much before the development of CNNs, since the mapping is highly nonlinear. Cao et al. [23] claimed that robust and accurate synthesis of MR from CT using a CNN is not feasible. However, in this chapter, we explore the possibility of CT-to-MR synthesis and demonstrate how it improves brain segmentation on CT images.

Gray matter (GM)/white matter (WM) segmentation and labeling on head images is an important research topic in neuroimaging. This research topic has been well studied and several excellent approaches exist, almost exclusively for MR images (cf. [84, 85, 86, 87]). If MR images were always available, then they could always be used for this purpose. Unfortunately, there are many scenarios in which only CT images are available, e.g., emergency situations, lack

of an MR scanner, patient implants or claustrophobia, and cost of obtaining an MR scan.

CT imaging of the head has many clinical and scientific uses including visualization and assessment of head injuries, intracranial bleeding, aneurysms, tumors, headaches, and dizziness as well as for use in surgical planning. Yet due to the poor soft tissue contrast in CT images, there has been very limited work on GM/WM segmentation from CT [88, 89, 90, 91, 92].

CT is easier to obtain compared with MRI. Yet to perform GM/WM segmentation, MRI has better soft tissue contrast and therefore has many existing automatic algorithms developed for it. It is natural to wonder whether we can combine the advantages of CT and MRI by synthesizing pseudo MR images from acquired CT images, and applying an existing automatic segmentation and labeling method on the pseudo MR images. Although Cao et al. [23] has a pessimistic conclusion about CNN-based CT-to-MR synthesis, we believe that because CNNs are resilient to intensity variations [93] and they can model highly nonlinear mappings, they are ideal for CT-to-MR synthesis. In fact, we demonstrate in this chapter that such synthesis is indeed possible and that whole brain segmentation and labeling from these synthetic images is very effective.

3.2 Methods and Data

Training and testing data. Twenty six patients had (T_1-w) MR images acquired using a Siemens Magnetom Espree 1.5 T scanner (Siemens Medical Solutions, Malvern, PA) with geometric distortions corrected within the Siemens

Syngo console workstation. The MR images were processed with N4 [94] to remove any bias field and subsequently had their intensity scales adjusted to align their WM peaks. Contemporaneous CT images were obtained on a Philips Brilliance Big Bore scanner (Philips Medical Systems, Netherlands) under a routine clinical protocol for brain cancer patients treated with either stereotactic-body radiation therapy (SBRT) or radiosurgery (SRS). The CT images were resampled to have the same digital resolution as the MR images, which is $0.7 \times 0.7 \times 1$ mm. Then the MR images were rigidly registered to the CT images.

We used ten patient image pairs as training data for our CNN (see below). Each 3D MRI volume contains hundreds of 2D slices. Ten training image pairs contains thousands of 2D training pairs. From the axial slices in the image domain, 128×128 paired (CT and MR) image patches are randomly cropped and extracted. These patch pairs were used to train a network based on a modified U-net [67] to synthesize MR patches from CT patches. The synthetic MR patches were then used to construct an axial slice of the synthetic MR image. Our network, with 128 x 128 CT patches as input and 128 x 128 synthetic MR patches as output, is shown in Figure 3.1.

Modified U-net for CT-to-MR synthesis. We have selected the U-net [67] as the basis of our synthesis network, and we made the following modifications to it for the synthesis task.

Modification 1: The standard U-net decoder has two 3×3 layers, whereas we use one with a 5×5 layer and a 3×3 layer. We do this because the upsampling layer uses nearest neighbor sampling. Thus, a 3×3 layer in the

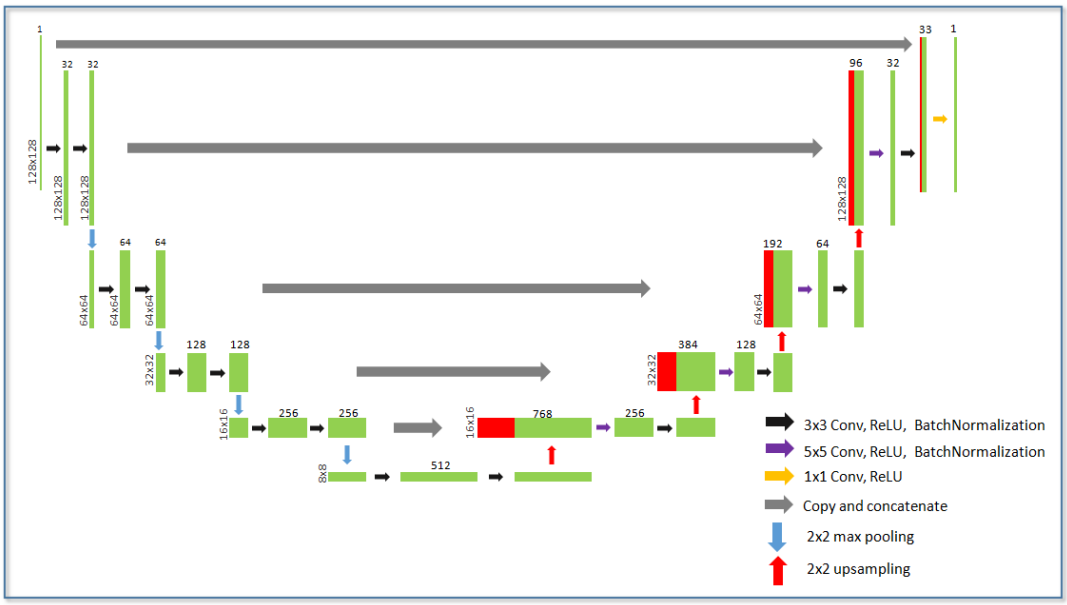


Figure 3.1: Four differences from the standard U-net: (1) The U-net decoder has two 3×3 layers, whereas we use one 5×5 layer and one 3×3 layer. (2) We exchange the order of the first convolutional layer and the merging layer in the decoder so that both are convolved twice. (3) When reconstructing a slice we use only the central 90×90 region of the image patches. (4) We merge the original CT patches before the last convolutional layer. Also, the original U-net used softmax to activate the last layer for segmentation while we use ReLU for regression.

encoder can involve its eight connected neighbors, whereas a 3×3 layer after an upsampling layer only includes three-connected neighbors. By replacing this with a 5×5 layer, we can still involve all eight connected neighbors. There is a slight increase in the number of parameters to estimate, but the result has better accuracy.

Modification 2: CNN vision tasks benefit from increasing model depth; however, deeper models can have vanishing or exploding gradients [74]. In the original U-net, the decoder contains an upsampling layer, a convolutional layer, a layer merging it with high resolution representations, and another two convolutional layers. Thus, the upsampled layer is convolved three times while the high resolution representation is convolved only twice. The upsampled layer is convolved one more time in the original U-net because of the issue we discussed in the first modification. We exchange the order of the first convolutional layer and the merging layer and let them both be convolved twice. With this change, our network can still model nonlinearities without introducing additional obstacles for back-propagation.

Modification 3: Every convolution loses border pixels; thus, the border of the predicted patch may not be as reliable as the center. The standard U-net crops each patch after each convolutional layer so that the predicted patch is smaller than the input patch. Our network keeps the boundary pixels instead of cropping them. However, when reconstructing a slice we use only the central 90×90 region of the image patches (except at the boundaries of the image, where we retain the side of the patch that touches the boundary).

Modification 4: CT numbers have a physical meaning. In order to include

this information, we merge the original CT patches before the last convolutional layer. Also, the standard U-net used a softmax operation to activate the last layer for segmentation, we use ReLU activation to better enable regression.

Automatic Whole-brain Segmentation and Labeling We use MALP-EM [85] to provide whole-brain segmentation and labeling from the synthetic MR images. Since the synthetic MR images are naturally registered with the CT images, the result is a segmentation and labeling of the CT images. MALP-EM uses an atlas cohort of 30 subjects having both MR images and labels from the OASIS database [95]. These atlases are deformably registered to the target and the labels are combined using joint label fusion (JLF) [84]. Finally, these labels are adjusted using an intensity based EM method to provide additional robustness to pathology, especially traumatic brain injury. We used the MALP-EM [85] code that has been made freely available by the original authors of the method.

3.3 Experiments and Results

Image Synthesis. Our network was trained on 45,575 128×128 image patch pairs derived from ten of the co-registered MR and CT images. It took two days to train and 1 min to synthesize one MR image from the input CT on an NVIDIA GPU GTX1070SC. Figs. 3.2(a)–(c) show an example input CT image, the resulting synthetic $T1$-w, and the ground truth $T1$-w. Figure 3.2(d) shows the dynamic range of Figure 3.2(a).

Experiment 1: MALP-EM segmentation We applied MALP-EM on both synthetic and ground truth $T1$-w images. Figure 3.2(e) shows the segmentation

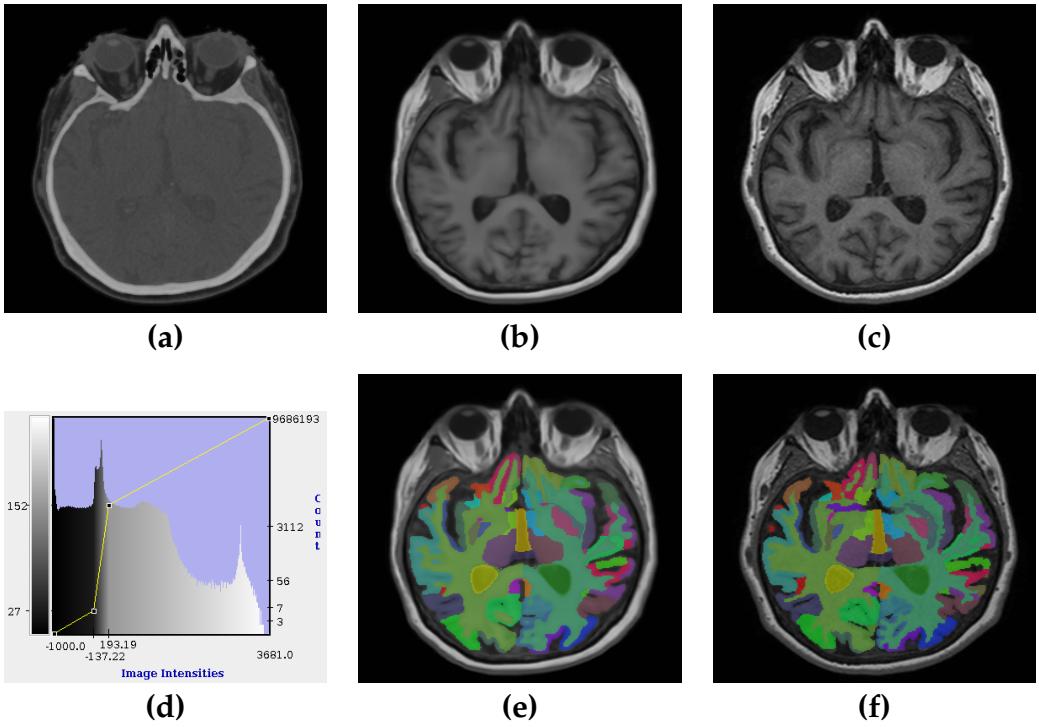


Figure 3.2: For one subject, we show the **(a)** input CT image, the **(b)** output synthetic $T1$-w, and the **(c)** ground truth $T1$-w image. **(d)** is the dynamic range of **(a)**. Shown in **(e)** and **(f)** are the MALP-EM segmentations of the synthetic and ground truth $T1$-w images, respectively.

result from the synthetic $T1$-w in Figure 3.2(b), while Figure 3.2(f) shows the result from the ground truth $T1$-w in Figure 3.2(c). There are differences between the two results, but this is the first result showing such a detailed labeling of CT brain images.

We used Dice coefficients to evaluate segmentation quantitatively. Dice coefficient is defined as $\text{DICE} = 2|X \cap Y|/(|X| + |Y|)$, with X and Y being two binary segmentation masks. We computed Dice coefficients between segmentation results obtained using synthetic $T1$-w and those obtained using the true $T1$-w. Table 3.1 shows mean Dice coefficients for a few small brain

structures. After merging the labels, box plots of the Dice coefficients for four labels: non-cortical GM, cortical GM, ventricles, and WM, are shown in Figure 3.3 (yellow graphics).

Dice	Hippocampus	Precentral gyrus	Postcentral gyrus	Caudate
Right	0.62	0.52	0.51	0.70
Left	0.59	0.55	0.52	0.73

Table 3.1: Mean Dice coefficients for a few brain structures.

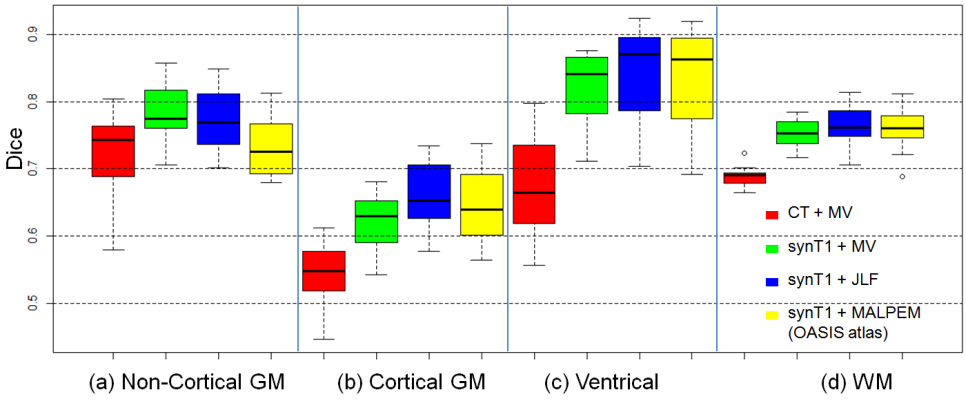


Figure 3.3: With MALP-EM processing of the ground truth *T*1-w as the reference, we computed the Dice coefficient between multi-atlas segmentations using either the subject CT images with MV label fusion (red), or synthetic T1-w with MV (green) or JLF (blue), as the label fusion, and MALP-EM (yellow). Note that MALP-EM (yellow) uses the OASIS atlas with manually delineated labels, while the other three use the remaining 15 images with MALP-EM computed labels from the true T1-w images as atlases.

Experiment 2: Comparison to direct multi-atlas segmentation. To demonstrate the benefits of our approach, we carried out a set of algorithm comparisons. Ideally, we would like to evaluate how well our CT images could be labeled directly from the OASIS atlases; but there are no CT data associated with OASIS. Instead, we used 16 subjects (which do not overlap the 10 subjects used to train our network) in a set of leave-one-out experiments and used the

MALP-EM labels as being "ground truth". For each of the 16 subjects, we used the remaining 15 (having T1-w and MALP-EM labels) as atlases. To mimic the desired experiment, we first carried out multi-modal registration from each of the 15 T1-w atlases to the target CT using the mutual information (MI) as the registration cost metric. Because this is a multi-modal registration task, JLF, which uses intensity similarity to compute weights, is not suitable to combine labels. So we used majority voting (MV) [96] instead. We next computed a synthetic $T1$-w image from the target CT image and registered each atlas to this target using the mean squared error (MSE) as the registration metric. To provide a richer comparison, we combined these 15 labels using both MV and JLF.

Given these three leave-one-out results, we computed Dice coefficients on four labels: non-cortical GM, cortical GM, ventricles, and WM. The results are shown in Figure 3.3 (using the red, green, and blue boxplots). We evaluated the significance of the improvement using the p-value of two-sided paired Sign Test on the **16** test subjects using the Matlab function signtest. The null hypothesis is that the median of the differences of the paired samples is zero. We found that there is a statistically significant differenec in median Dice difference between paired samples when using the synthetic $T1$-w images than the original CT images, and this was true using either MV or JLF ($p < 0.001$). We also found that JLF resulted in a significantly higher median Dice difference than MV in the case of cortical GM, ventricle, and WM ($p < 0.01$). The median performance of MV was not found to be different with JLF for noncortical GM with the synthetic $T1$-w ($p > 0.01$).

3.4 Discussion and Conclusion

We have described a CNN-based CT-to-MR synthesis algorithm, and used the synthetic MR images to improve the performance of whole brain segmentation. The synthetic images that we achieved with the deep network are quite good visually as demonstrated by the single (typical) example shown in Figure 3.2(b), which is visually much better than those shown in Cao et al. [23] (their Figure 7). This speaks very well to the potential of the network architecture for estimating synthetic cross-modality images. Besides whole-brain segmentation and labeling, there are a host of other potential applications for synthetic MR images.

A limitation of our evaluation is our lack of manual brain labels in a CT dataset, as it would be interesting to compare our approach with a top multi-atlas segmentation algorithm that would use only CT data. The fact that our method appears to perform worse than the straight multi-atlas results in Figure 3.3 is because the MALP-EM result is using manually delineated OASIS labels to estimate automatically generated MALP-EM labels, whereas the other two approaches are estimating MALP-EM labels from MALP-EM atlases. In the future, a more thorough evaluation including a quantitative comparison with Cao et al. [23] is warranted.

Past research using contrast-enhanced 4D CT brain segmentation achieves slightly higher mean Dice than ours, with 0.81 and 0.79 for WM and GM [88], compared to our result as 0.77 and 0.76, respectively. However, because their data was 4D CT, their combined 3D image probably has lower noise than ours and also enables them to use temporal features, which we do not have.

Furthermore, theirs was a contrast CT study while ours is a non-contrast study.

In this chapter, we have used a modified U-net to synthesize $T1$-w images from CT, and then directly segmented the synthetic $T1$-w using either MALP-EM or a multi-atlas label fusion scheme. Our results show that using synthetic MR can significantly improve the segmentation over using the CT image directly. This is the first work to provide GM anatomical labels on a CT neuroimage. Also, despite previous assertions that CT-to-MR synthesis is impossible from CNNs, we show that it is not only possible but it can be done with sufficient quality to open up new clinical and scientific opportunities in neuroimaging.

Chapter 4

SMORE: Synthetic Multi-Orientation Resolution Enhancement

4.1 Introduction

Exploring modality synthesis is one contribution of this thesis. Another important area that we explore is image spatial resolution. In this chapter, we discuss how to improve through-plane resolution for a common type of MR acquisition that has high in-plane resolution and lower through-plane resolution.

Before we provide background about resolution enhancement in MR images, we answer the question: why is this type of MR acquisition common? Although hardware and software improvements have led to significant improvements in MRI resolution over the years [97], further efforts to improve resolution inevitably involve tradeoffs with signal-to-noise ratio (SNR) and acquisition time. Generally, to achieve very high resolution (HR) MRI with adequate SNR, the acquisition time must be very long, which costs money,

lowers patient throughput, and can lead to both patient discomfort and motion artifacts. As a consequence of this tradeoff, it is quite common in both clinical and research practice to acquire MR images with high in-plane resolution and lower through-plane resolution. The resulting elongated voxels have sufficient tissue volume to yield adequate SNR while providing one HR view with acceptable diagnostic quality.

To improve spatial resolution, many learning-based SR algorithms have been reported for MRI, including sparse coding [98], random forests [99], and convolutional neural networks (CNNs) [100, 101]. However, self-supervised super-resolution (SSR) is more desirable since it avoids the requirement of external training data. Here, we focus on MR acquisitions that have elongated voxels, which makes an important class of SSR algorithms possible. In 2016, Jog et al. [102] developed an SSR framework (which we call JogSSR) that extracts training patches from the LR MRI and blurred LR_2 images, trains an anchored neighborhood regression [103], and then applies the trained regressor to LR_2 images in different directions. The resultant images are LR, but have low resolution in different directions. Thus, each of them contributes high frequency information to a different region of Fourier space. This framework is then analogous to the multi-image SR methods [31, 32, 33, 34] but does not require actual acquisition of the additional LR images or registration. Finally, these images are combined through Fourier burst accumulation (FBA) [104] to obtain an HR image. Taking the basic idea of JogSSR which extracts training patches from the LR MRI itself, the more recent CNNs further improve the SSR results [105, 43, 44, 45, 46].

To apply SSR on this type of MR image, the mechanism of blurring which creates LR images from HR in-plane slices should agree with the actual acquisition model in the through-plane direction. This requires us to understand the acquisition model for MRI. MRI acquisition protocols can be divided into two broad categories that are both commonly used in clinical and research scanning:

- 3D protocols acquire MR data in 3D *k*-space, i.e. the image signals are acquired in 3D Fourier domain. The three spatial resolutions are proportional to the inverses of the frequencies covered in 3D *k*-space as determined by two phase encoding directions and one read-out direction. We refer such MRI as 3D MRI.

- 2D protocols acquire MR data in 2D *k*-space after slice selection, and then form 3D volumes by stacking the 2D images in their through-plane direction. The in-plane resolutions are proportional to the inverses of the phase encode and read-out frequency ranges while the through-plane resolution is given by the full-width-half-max (FWHM) of the slice profile—i.e., the slice thickness. We refer such MRI as 2D MRI.

There is a visual difference between MR images from these two type of acquisitions, described as below.

- In 2D MRI, with inadequate slice separation (also called slice increment), which is equivalent to undersampling in through-plane axis of image domain, there is nearly always spatial aliasing. Spatial aliasing appears as moiré patterns in image domain and overlapped high-frequency

contents in k-space.

- In 3D MRI, there is no such spatial aliasing, since there is no overlapping of high-frequency contents in k-space. 3D MRI may suffer from another type of aliasing, which results from undersampling in k-space and appears as wrap-around artifacts in the image domain, but we do not consider this type of aliasing in this chapter.

With this understanding of MRI acquisition protocols, in this chapter we describe an SSR method SMORE, which is the first SSR method that distinguishes MRI 2D and 3D protocols and performs anti-aliasing on 2D MRI.

4.2 Method

We refer to our basic algorithm as Synthetic Multi-Orientation Resolution Enhancement (SMORE). The basic idea of SMORE is to train a super-resolution network on HR x-y plane patches, and then to apply it to LR x-z and y-z plane slices. In order to make it work,

- we carefully prepare training data to mimic the way in which through-plane resolution is degraded in actual MRI, which is very different for 2D and 3D protocols;

- we use a ResNet-based network using a small patch size in order to focus on local structures rather than global structures in the x-y plane.

Because MRI acquired in 2D and 3D protocols are different, the details of

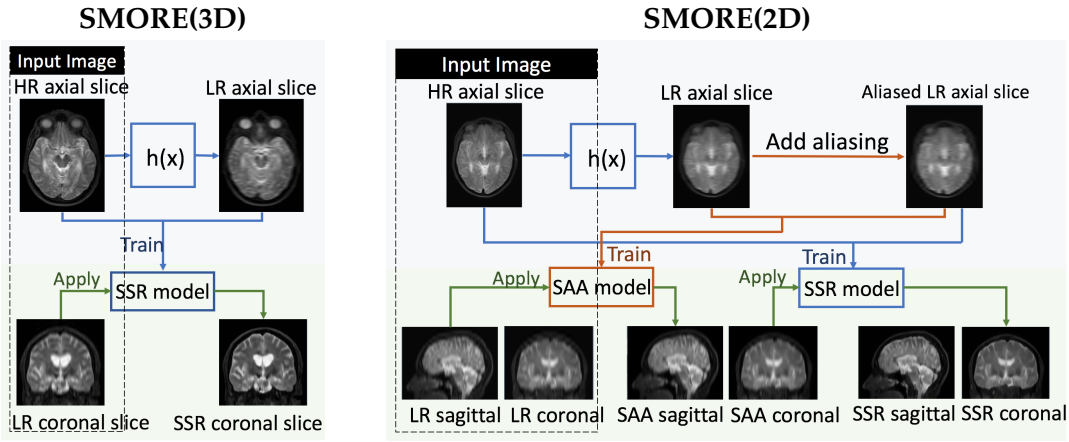


Figure 4.1: Overview of SMORE. A simplified workflow of SMORE for MRI acquired with 3D protocols and 2D protocols, referred as SMORE(3D) and SMORE(2D).

SMORE are different, and therefore we refer to them as SMORE(2D) and SMORE(3D), respectively.

Consider an anisotropic input image with spatial resolution (full-width-half-max or FWHM) of $a \times a \times c$ where $c > a$. We denote the resolution ratio as $r = c/a > 1$, which needs not be an integer. We model this image as a low resolution (LR) version of a high resolution (HR) image $f(x, y, z)$, which has spatial resolution of $a \times a \times a$ and therefore resolution ratio $r = 1$. Here x, y, and z are spatial coordinates and the through-plane direction is defined to be along the z-axis. Note that the images are digitized, yet we use continuous notation for simplicity. Fourier space (also known as k-space) has coordinates u, v, and w corresponding to the x, y, and z spatial coordinates respectively. The goal of this work is to restore the HR image $f(x, y, z)$ from the acquired LR anisotropic image without using any external training data.

Fig. 4.1 shows the overview of SMORE for MRI acquired with 3D and 2D protocols assuming that axial slices are in-plane slices. Next we explain them

in detail.

4.2.1 Simplified SMORE(3D)

In this section, we describe simplified SMORE(3D), which is provided for reference as a pseudocode in Algorithm 1. In an anisotropic subject image with resolution of $a \times a \times c$, only the z-axis is LR, while the x-y plane slices are $a \times a$ HR. In SMORE(3D), we first learn a LR to HR mapping using 2D HR x-y plane slices as training data, then apply the mapping to 2D LR y-z plane or x-z plane slices and restore HR information. While this description captures the essence of SMORE(3D), the following text provides the details.

Algorithm 1: SIMPLIFIED SMORE(3D)

Data: LR image with voxel size and spatial resolution of $a \times a \times c$, $r = c/a > 1$

Result: Estimate $f(x, y, z)$ with target spatial resolution $a \times a \times a$

(Step 1) Preprocessing:
In k-space, zero-fill the LR image to make it isotropic with voxel size of $a \times a \times a$. Then apply N4 correction on the image if necessary. This yields $g(x, y, z)$.

(Step 2) Construct Training Data:
Blur the image in the x-axis: $b(x, y, z) := h(x; r) * g(x, y, z)$;

(Step 3) Train a SSR network:
Randomly extract 2D paired patches from x-y plane slices of $\{b(x, y, z), g(x, y, z)\}$;
Feed the patches into a CNN model to train a self-supervised super-resolution network (SSR).

(Step 4) Apply the SSR network:
Apply trained SSR on the x-z plane: $s(z, x) := \text{SSR} \circ g(z, x)$;
Stack slices $s(z, x)$ to reconstruct $s(x, y, z)$;
$\hat{f}(x, y, z) := s(x, y, z)$

4.2.1.1 (Step 1) Preprocessing

For 3D MRI, the original LR image does not acquire high frequency signals in the z direction of k-space. We use the traditional way of interpolation for 3D MR images. We first zero-pad the acquired image in k-space so that its voxels have digital resolution (voxel size) equal to $a \times a \times a$, which is also called sinc interpolation. If the scan has non-uniform intensities (generally true in MRI) we apply N4 inhomogeneity correction [94]. The resultant image, which has a voxel size equal to $a \times a \times a$ and a physical resolution equal to $a \times a \times c$, is denoted by $g(x, y, z)$ and is used as input in the following steps.

4.2.1.2 (Step 2) Construct Training Data

The idea behind SMORE(3D), like in [102], is that 2D axial slices $g_z(x, y)$ of $g(x, y, z)$ can be thought of as $a \times a$ HR slices, which are used to construct paired LR/HR training data.

The way we construct paired LR data from 2D HR axial slices $g_z(x, y)$ is using an impulse response $h(x)$. We use $h(x)$ to obtain the through-plane resolution c in the x-direction, i.e., $b_z(x, y) = h(x) * g_z(x, y)$. Then corresponding image patches from b_z and g_z can be used to train a regression that will restore—i.e., super-resolve—g_z from b_z.

Let us analyze what kind of filter $h(x)$ should be. In 3D MRI, $G(u, v, w)$, the k-space of the LR image $g(x, y, z)$, has the high frequency region in k-space filled with zeros. $F(u, v, w)$, the k-space of HR image $f(x, y, z)$, has high frequency signals acquired. Comparing $G(u, v, w)$ and $F(u, v, w)$, we can find that $G(u, v, w)$ is simply the multiplication of $F(u, v, w)$ by the rect

function. Thus, ideally the Fourier transform $H(u;r)$ of $h(x)$ should be a rect function, i.e., $H(u;r) = \text{rect}(ru)$. In practice, however, the acquired Fourier data is typically multiplied by an anti-ringing window such as a Fermi filter [106]. Since the parameter of the Fermi filter is usually unknown, we use a conservative setting (to avoid excessive filtering) as in Bernstein et al. [106], given by Equation (4.1), where L is the image size in the x direction and r is the ratio between through-plane and in-plane resolution,

$$\text{Fermi}(u;r) = \frac{1}{1+\exp\left(\frac{|u|-1/2r}{10/L}\right)}. \tag{4.1}$$

Accordingly, we model $G(u,v,w)$ (in the w direction) as the product of a rect, a Fermi filter, and $F(u,v,w)$. The combination of the rect filter and the Fermi filter is

$$H(u;r) = \text{rect}(ru)\text{Fermi}(u;r)\,, \tag{4.2}$$

so the blurred image can be expressed as

$$b(x,y,z) = h(x;r) * g(x,y,z)\,, \tag{4.3}$$

where $h(x;r)$ is the inverse Fourier transform of $H(u;r)$.

4.2.1.3 (Step 3) Train a SSR network

We now train a patch-based SSR regression from paired patches selected from the x-y plane (and any z slice) in the images $b(x,y,z)$ (input) and $g(x,y,z)$ (output). Note that we are not focused on designing a deep network architecture in this chapter; instead have chosen to use EDSR [75] (Enhanced Deep Residual Networks for Single Image Super-Resolution), which was evaluated

as the most accurate method in the NTIRE CVPR 2017 [36] and PIRM ECCV 2018 super-resolution challenges [107]. EDSR is a ResNet-based network that removes batch normalization and includes residual scaling to improve performance. The EDSR architecture does not contain any pooling blocks, which makes it focus more on local features rather than global features as in the U-net. This is beneficial as we require the network to enhance edges without structural specificity and to better preserve pathology. We use mean absolute error as the loss function. According to [36, 75, 108], optimizing with L1 loss works better than L2 for super-resolution even when evaluating with L2 loss like PSNR.

The original EDSR takes a downsampled image as input, and outputs an upsampled image. We removed the upsampling layer in EDSR since it only supports an integer ratio r, and made the network output be an image with the same size as the input image. Another benefit of removing the upsampling layer (from the original EDSR) is that the trained network can be used as a pre-trained network for an image from a new dataset. Other than this change, EDSR is used without change; we note that a different super-resolution network could be used instead if desired.

There is no external training data needed for training. However, we found that starting from a pre-trained network can accelerate training even when it was trained with images derived from a different dataset. In addition, each time we experiment with SMORE on a new dataset, a pre-trained network can be used as the initial network and then trained with the new data to improve its performance. In these evaluations, we trained the network from scratch on

the first subject image from Experiment 4.3.1 and then used this network as the pre-trained network for all subsequent experiments.

Since the anatomies of in-plane slices and through-plane slices are different, there is a concern that the features learned from the in-plane slices might not be appropriate for the through-plane slices. To solve this issue, we train the network with relatively small patches (32×32), which forces the network to learn to enhance edges without seeing the corresponding anatomies. Thus, the learnt features are not related to large anatomical features. It is our observation that such small patches restrict the effective receptive field [109], enhance edges without structural specificity, and better preserve pathology. The largest resolution ratio r that we studied in these experiments is 6.6667; for a larger ratio r, larger patches might be necessary.

As for other hyper-parameters, the batch size depends on the GPU and a larger batch size is almost always better. The optimizer we used is Adam with a learning rate of 10^{-4}. The way we chose these hyper-parameters is explained in Sec. 4.2.2.1.

4.2.1.4 (Step 4) Apply the SSR network

The trained EDSR network is applied to $g(z, x)$, the x-z planes of g, which yields a super-resolved estimate $\hat{s}(z, x)$. After stacking $\hat{s}(z, x)$ in the y direction, we have a super-resolved result $\hat{f}(x, y, z) = s(x, y, z)$.

4.2.2 SMORE(3D)

There are two issues to consider with simplified SMORE(3D): there is no validation data during training to avoid overfitting and there is no self-ensemble during testing as suggested in [75, 36]. Therefore, we add rotation during training and testing to simplified SMORE(3D), yielding SMORE(3D), which for reference is provided as pseudocode in Algorithm 2. Note that the rotation angle during training and testing are unrelated.

4.2.2.1 Rotation during training

This is a common data augmentation technique in deep learning. CNNs are not invariant to rotation. Thus, rotating training images effectively enlarges our training data and provides validation data.

We rotate the image $g(x, y, z)$ about the z-axis by θ and repeat this process to yield $g_\theta(x, y, z)$ and a corresponding set of blurred images $b_\theta(x, y, z)$. In this evaluation we use seven rotations where $\theta = n\pi/12$ for $n = 0, \ldots, 6$, but this generalizes for any number and arrangement of rotations. The extracted paired training patches are then randomly flipped as additional data augmentation.

Part of the training data is used separately as validation data. In particular, we use training samples obtained from one rotation angle as validation data and the other six rotation angles as training data. During training, we save the model with the best validation loss. With this validation data, one can choose hyper-parameters (such as the learning rate) based on a validation loss rather than using HR ground truth data, which is not available in a real

scenario. Note that for self-supervised super-resolution, even with pre-trained weights, each subject requires extra fine-tuning. The optimal learning rate may vary from data to data. But it is not practical to choose hyper-parameters separately for every single image. In the presented experiments, we chose hyper-parameters including batch size, patch size, and learning rate, based on one subject—the first subject in Experiment 4.3.1—and then used them for all subsequent experiments. Empirically the same parameter settings provided good performance for the datasets in this chapter.

In SMORE, training patches are not extracted using a sliding window. Instead, for each training batch, we randomly select a rotated training sample and then randomly crop training patches from it. This gives a larger randomness than cropping training patches from a sliding window. With this training strategy, the total number of training batches is the main parameter that controls the training time rather than the number of rotations from data augmentation. We did not explicitly explore the impact on performance of this data augmentation strategy, but this is certainly an aspect that could be explored in future optimization efforts.

Algorithm 2: SMORE(3D) PSEUDOCODE

Data: LR image with voxel size and spatial resolution of $a \times a \times c$, $r = c/a > 1$

Result: Estimate $f(x, y, z)$ with target spatial resolution $a \times a \times a$

(Step 1) Preprocessing:

In k-space, zero-fill the LR image to make it isotropic with voxel size of $a \times a \times a$. Then apply N4 correction on the image if necessary. This yields $g(x, y, z)$.

(Step 2) Construct Training Data:

for $n = 0, \ldots, N; \quad \theta := \frac{n\pi}{2N}$ **do**

  Rotate image about the z-axis: $g_\theta(x, y, z) := R_z(\theta) \circ g(x, y, z)$;

  Blur the image in the x-axis: $b_\theta(x, y, z) := h(x; r) * g_\theta(x, y, z)$;

end

(Step 3) Train a SSR network:

Randomly extract 2D paired patches from x-y plane slices of $\{b_\theta(x, y, z), g_\theta(x, y, z)\}$;

Randomly flip the extracted paired patches;

Feed the patches into a CNN model to train a self-supervised super-resolution network (SSR).

(Step 4) Apply the SSR network:

for $m = 0, \ldots, M-1; \quad \alpha := \begin{cases} 0 & \text{if } M = 1 \\ \frac{m\pi}{2(M-1)} & \text{otherwise} \end{cases}$ **do**

  Rotate image about the z-axis: $g_\alpha(x, y, z) := R_z(\alpha) \circ g(x, y, z)$;

  Apply trained SSR on the x-z plane: $s_\alpha(z, x) := \text{SSR} \circ g_\alpha(z, x)$;

  Stack slices $s_\alpha(z, x)$ to reconstruct $s_\alpha(x, y, z)$;

  Rotate it back: $\hat{f}_\alpha(x, y, z) := R_z(-\alpha) \circ s_\alpha(x, y, z)$.

end

(Step 4) FBA:

$\hat{f}(x, y, z) := \text{FBA}(\{\alpha : \hat{f}_\alpha(x, y, z)\})$

4.2.2.2 Rotation during testing

In the simplified SMORE(3D), the trained SSR network is applied only to LR x-z plane slices. It is natural to ask why not apply it to LR y-z plane slices. In fact, the trained SSR network can be applied to any collection of patches that are orthogonal to the x-y plane. To implement this idea, we apply it to the x-z planes of rotated versions g_α of g that have been rotated about the z-axis by α. The x-z planes of g_α are the x-z planes of g when $\alpha = 0$ and are the y-z planes of g when $\alpha = \frac{\pi}{2}$. Each rotated image g_α yields a super-resolved estimate $\hat{s}_\alpha(x, y, z)$, which is then rotated back to yield the estimate $\hat{f}_\alpha(x, y, z)$. We do this for M different values of α, obtaining M estimated HR volumes, which are combined using Fourier burst accumulation (FBA) [104]. FBA is a self-ensemble technique, which has been described in previous SSR literature [110, 44, 43]. We explore the effect of this self-ensemble technique in Section 4.3.4. In our previous work [44, 43], we used $M = 2$, which implies that we applied the trained network to only coronal and sagittal slices. In order to show whether applying the network in multi-orientations improves the result, we show a comparison between $M = 1$, 2, and 7 in Sec. 4.3.4. All other experiments in this chapter and the next chapter use $M = 1$, which is much faster to compute and adequate for most applications.

4.2.3 SMORE(2D)

In this section, we describe the difference between SMORE(2D) and SMORE(3D); pseudocode for SMORE(2D) is shown in Algorithm 3.

To make an LR image have isotropic voxel spacing, the interpolation

methods for 3D MRI and 2D MRI are different. 3D MRI yields LR images by cutting off frequencies in the Fourier domain, thus we use sinc interpolation in SMORE(3D). 2D MRI yields through-plane LR due to large slice separation in the image domain, thus the interpolation should be implemented in the image domain. In SMORE(2D), we use B-spline (BSP) interpolation to make isotropic voxel spacing.

Theoretically, aliasing will still exist in the z direction after BSP if the slice separation is large. This is because BSP, as a linear operator, does not remove the overlapped high frequency content in the k-space of an aliased image. BSP is performed only to yield the desired digital resolution; it does not change the effect of aliasing. Given this starting point, the basic idea of SMORE(2D) is the same as SMORE(3D) except for two differences. First, since the image acquisition models are different, the training data extraction step which mimics image acquisition must be changed. Second, when aliasing is too severe, an extra self-supervised anti-aliasing (SAA) step is applied.

4.2.3.1 Training Data Extraction

For 2D MRI, we must model the slice selection process. In order to create training data from the observed data with LR in the x direction, we use a 1D Gaussian filter $h(x; r)$ as the MRI slice selection profile with a full-width at half-maximum (FWHM) equal to r. The filtered image $b(x, y, z)$ has the desired LR components but does not have aliasing.

Aliasing comes from large slice separation, which means low sampling rate. To introduce aliasing when the slice separation is r_s times the slice thickness,

Algorithm 3: SMORE(2D) PSEUDOCODE

Data: LR image with voxel size and spatial resolution of $a \times a \times c$,
$r = c/a > 1$

Result: estimate $\hat{I}(x, y, z)$ with target spatial resolution $a \times a \times a$

(Step 1) Preprocessing:

BSP interpolate the LR image to make it isotropic with voxel size of $a \times a \times a$; and apply N4 correction on the image if necessary, results in $g(x, y, z)$.

(Step 2) Construct Training Data:

for $n = 0, \ldots, N; \quad \theta := \frac{n\pi}{2N}$ **do**

  Rotate image about the z-axis: $g_\theta(x, y, z) := R_z(\theta) \circ g(x, y, z)$;

  Blur the image in the x-axis: $b_\theta(x, y, z) := h(x; r) * g_\theta(x, y, z)$;

  Introduce aliasing in the x-axis: $\tilde{b}_\theta(x, y, z) := \uparrow_x^r (\downarrow_x^r (b_\theta(x, y, z)))$.

end

if SAA_{Bool} **then**

  Randomly extract 2D paired patches from x-y plane slices of $\{\tilde{b}_\theta(x, y, z), b_\theta(x, y, z)\}$, randomly flip them, and train a self-supervised anti-aliasing network (SAA).

Randomly extract 2D paired patches from the x-y plane slices of $\{\tilde{b}_\theta(x, y, z), g_\theta(x, y, z)\}$, randomly flip them, and train a self-supervised super-resolution network (SSR).

(Step 3) SAA+SSR:

for $m = 0, \ldots, M-1; \quad \alpha := \begin{cases} 0 & \text{if } M = 1 \\ \frac{m\pi}{2(M-1)} & \text{otherwise} \end{cases}$ **do**

  Rotate image about the z-axis: $g_\alpha(x, y, z) := R_z(\alpha) \circ g(x, y, z)$;

  if SAA_{Bool} **then**

    Apply trained SAA on y-z plane: $\hat{g}_\alpha(z, y) := \text{SAA} \circ g_\alpha(z, y)$;

    Stack slices $\hat{g}_\alpha(z, y)$ to reconstruct $\hat{g}_\alpha(x, y, z)$;

  else

    $\hat{g}_\alpha(x, y, z) = g_\alpha(x, y, z)$

  end

  Apply trained SSR on x-z plane: $s_\alpha(z, x) := \text{SSR} \circ \hat{g}_\alpha(z, x)$;

  Stack slices $s_\alpha(z, x)$ to reconstruct $s_\alpha(x, y, z)$;

  Rotate it back: $\hat{f}_\alpha(x, y, z) := R_z(-\alpha) \circ s_\alpha(x, y, z)$.

end

(Step 4) FBA:

$\hat{f}(x, y, z) := \text{FBA}(\{\alpha : \hat{f}_\alpha(x, y, z)\})$

the image is downsampled by a factor of $r_s \cdot r$ using linear interpolation. In most MR images, $r_s = 1$. Occasionally, $r_s = 0.5$ and aliasing is less in such acquisition. In this chapter, all image data have $r_s = 1$. We denote the downsampled image as $\downarrow_x^r (b(x,y,z))$. To complete the process we upsample this image by a factor $r_s \cdot r$ using BSP interpolation, yielding $\tilde{b}(x,y,z) = \uparrow_x^r (\downarrow_x^r (b(x,y,z)))$. The image $\tilde{b}(x,y,z)$ contains aliasing artifacts just like the acquired image in the through-plane direction. The training data extraction process is summarized as below:

$$g \xrightarrow{*h} b \xrightarrow{\downarrow\uparrow} \tilde{b}$$

As in SMORE(3D), we rotate this image in the x-y plane with angle θ to increase the amount of training data.

4.2.3.2 SAA when unexpected aliasing exists

In SMORE(2D), we train an SSR network from aliased LR images $\tilde{b}$ and HR images g. The resulting network should therefore both remove aliasing and improve resolution. However, we found experimentally that some residual aliasing artifacts may remain, as shown in Sec. 4.3.3. We found that, in such cases, adding an additional self-supervised anti-aliasing (SAA) network can remove them.

SAA uses an EDSR network as does SSR, but it is trained independently. We extract 32×32 patch pairs randomly from axial slices in $\tilde{b}(x,y,z)$ and $b(x,y,z)$ (i.e., aliased LR slices and LR slices, respectively) to train the SAA network that removes aliasing. During testing, we apply the trained SAA

network to sagittal slices first and then apply the trained SSR network to this result on coronal slices. This removes aliasing in both the coronal and sagittal planes.

Whether to train and apply SAA is controlled with a boolean variable SAA_{Bool}. As there is less observable aliasing for LR images with a small ratio, $r_s \cdot r$, we empirically pick a threshold of 3 and set $\text{SAA}_{\text{Bool}} = \text{False}$ when $r_s \cdot r \leq 3$ in these experiments. This saves computation time and yields nearly identical results.

4.2.4 Comparison with other SSR methods

Compared with JogSSR [102], SMORE uses the state-of-the-art SR deep network EDSR [75]. Compared with Weigert et al. [105], a CNN-based SSR method developed for 3D fluorescence microscopy images, SMORE distinguishes and addresses the different requirements—and also provides different algorithms—for 2D versus 3D MRI, and adds an anti-aliasing network that preceeds the SSR network for 2D MRI acquisitions. The comparison is summarized in Table 4.1.

Table 4.1: Comparison of several SSR methods. ANR is Anchored Neighborhood Regression [103], which is based on sparse coding.

	Modality	Subject image acquisition protocol	Model	Anti-aliasing
JogSSR [102]	MRI	not distinguished	3D ANR	No
Weigert et al. [105]	Microscopy	acquired as 2D, stack to 3D	2D U-net	No
SMORE(3D)	MRI	acquired in 3D *k*-space	2D EDSR	No
SMORE(2D)	MRI	acquired as 2D, stack to 3D	2D EDSR	Yes

4.3 Experiments

4.3.1 Simulation experiments using T_2-weighted brain images

In these experiments, we used 14 T_2-weighted images of multiple sclerosis subjects; acquired on a 3T Philips Achieva scanner using a 3D imaging protocol with $1 \times 1 \times 1$ mm resolution. These images serve as our ground truth HR images from which we simulate LR MRI using both 3D and 2D MRI acquisition protocols. The resulting LR MR images were used as input data to compare various SSR methods.

4.3.1.1 LR data downsampled following a 3D protocol

To simulate LR 3D MRI, we applied an ideal low-pass filter to the isotropic $T2$-weighted images followed by an anti-ringing Fermi filter, both in the z direction (which defines the through-plane direction). These images simulate 3D MR acquisitions having both digital and spatial resolutions equal to $1 \times 1 \times r$ where $r = \{2, \ldots, 6\}$ with no aliasing. The first step in super-resolution processing is to upsample these images to $1 \times 1 \times 1$ mm digital resolution using zero-padding in Fourier space, i.e. sinc interpolation, referred to as sincInterp.

We applied the following methods to the simulated LR MR images: 1) the total variation method LRTV [35], which has publically available code; 2) the SSR method of Jog et al. [102], referred to as JogSSR; and 3) and our SMORE(3D) algorithm. Note that the original code of LRTV [35] can only carry out super-resolution for isotropic ratios $r \times r \times r$, with r restricted to integers. In order to apply the code for anisotropic ratios $1 \times 1 \times r$, we modified a few lines of

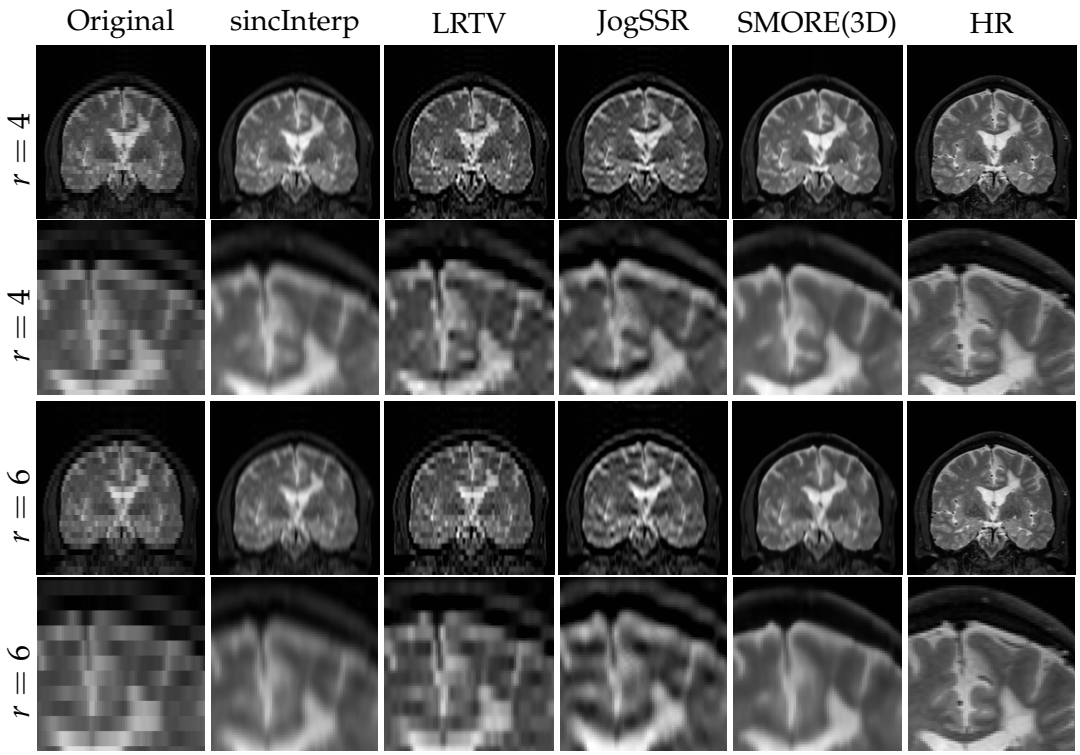


Figure 4.2: Results from MRI downsampled with a 3D protocol: Coronal views of the $1 \times 1 \times r$ mm 3D LR input image restored using sinc interpolation, LRTV [35], JogSSR [102], our method SMORE(3D), and the HR truth image. The zoomed patches show a lesion that is near ventricle.

code in its upsample and downsample functions merely to change upsample/downsample axis from all the three axes to just the z-axis[1]. Despite this, we did not change the code thereby preserving the original LRTV code as much as possible[2].

Example results from this simulation experiment are shown in Fig. 4.2 for $r = 4$ and 6. We zoom in the area around a lesion which is near ventricle to show the details. Visually, SMORE results are significantly better than

[1] Original LRTV: https://bitbucket.org/fengshi421/superresolutiontoolkit/
modified LRTV: https://github.com/volcanofly/LRTV_revision

[2] We note that the upsampling method used in the original code of LRTV is nearest-neighbor interpolation, which we believe will cause artifacts if the resolution ratio r is large. However, we did not change it.

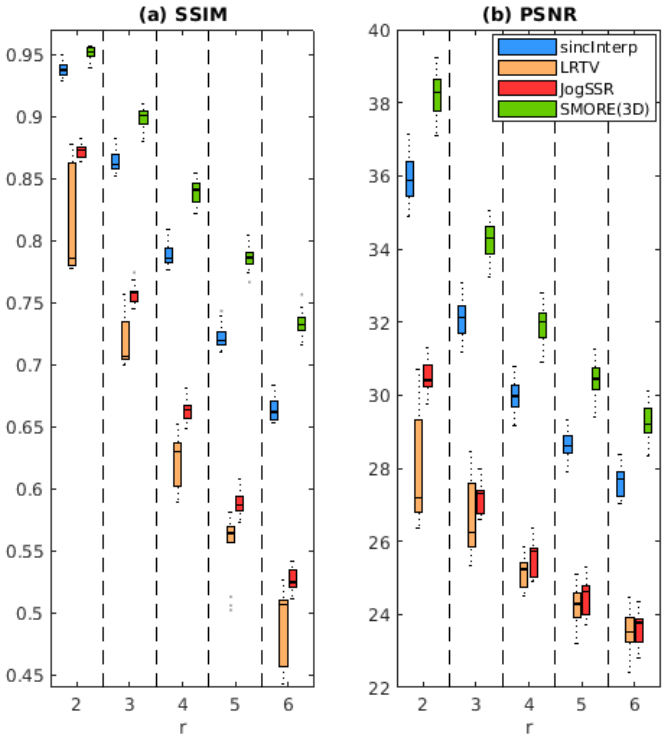


Figure 4.3: Evaluation of accuracy for super-resolution results from LR MRI downsampled with a 3D protocol: (a) SSIM values and (b) PSNR comparing to ground truth for the restored images using *k*-space zero-padding interpolation (blue), LRTV [35] (orange), JogSSR [102] (red), and SMORE(3D) (green). Higher value indicates better accuracy.

other methods, and the lesions near the ventricle are well preserved while the LRTV and JogSSR methods retain artifacts from the elongated voxels. Note that the original experiments in LRTV only dealt with $r = 2$ and JogSSR only dealt with $r = 2$ and $r = 3$, and these artifacts are not obvious in their papers. However, for data with larger values of r, these algorithms do not perform well while our SMORE(3D) shows visually improved resolution and few visible artifacts.

With the $1 \times 1 \times 1$ mm ground truth as the reference, we compute structural

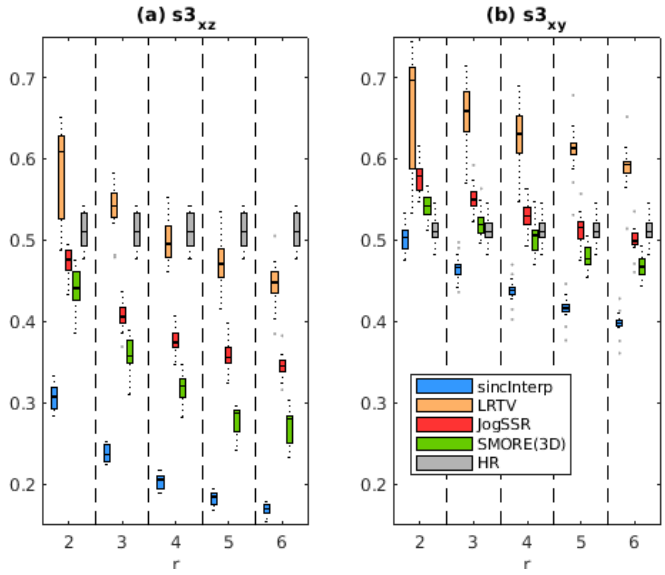


Figure 4.4: Evaluation of sharpness for super-resolution results from LR MRI downsampled with a 3D protocol: S3 sharpness values in (a) x-z plane (LR) and (b) x-y plane (HR) for the restored images using sinc interpolation (blue), LRTV [35] (orange), JogSSR [102] (red), SMORE(3D) (green), and HR truth (grey). Higher value indicates sharper edges.

similarity (SSIM)[111] index and the peak signal-to-noise ratio (PSNR)[3] within a mask of non background voxels, shown in Fig. 4.3.

No-reference measures of algorithm performance can also be computed from these results. We computed S3 [112], a 2D spectral and spatial measure of local perceived sharpness based on the slope of the magnitude spectrum and the total spatial variation[4]. We computed S3 for the coronal slices (x-z) which are LR before applying SSR methods, and the axial slices (x-y) which are HR before applying SSR methods. These results are shown in Fig. 4.4. Results from LRTV [35] have the highest sharpness, sometimes even higher than the HR ground truth. But in this case, higher computed sharpness does

[3]SSIM and PSNR code: https://github.com/volcanofly/ssim_and_psnr_3d.
[4]S3 code: http://vision.eng.shizuoka.ac.jp/s3.

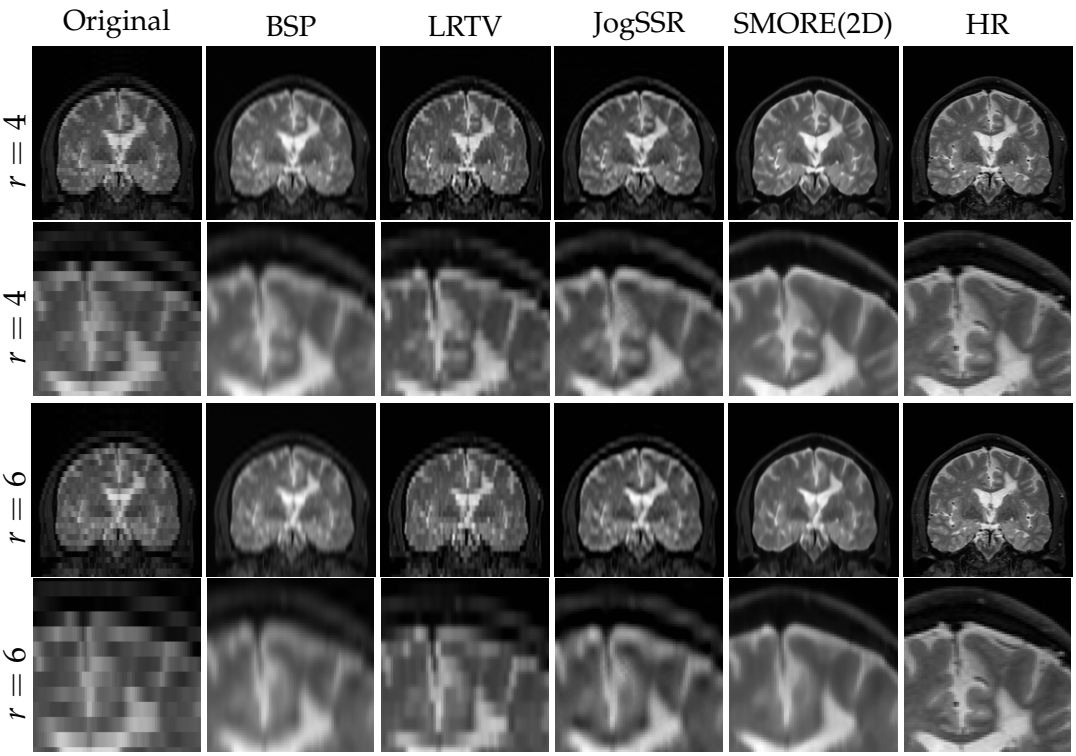


Figure 4.5: Results from MRI downsampled with a 2D protocol: Coronal views of the $1 \times 1 \times r$ mm 2D LR input image interpolated with cubic b-spline, restored using methods from LRTV [35], JogSSR [102], SMORE(2D), and the HR ground truth image. The zoomed patches show a lesion that is near ventricle.

not indicate a better result. Instead, as we can see from the visualizations in Fig. 4.2, LRTV [35], and to a lesser extent JogSSR, tends to emphasize sampling artifacts rather than true edges within the image. It shows that the sharpness measure sometimes gives an opposite quality score to that of SSIM/PSNR. In PIRM ECCV 2018 super-resolution challenge [107], participants also found that perceptual quality measures sometimes disagree with SSIM/PSNR. Nevertheless, SMORE(3D) produces results with the highest SSIM/PSNR scores and also higher sharpness than interpolation.

4.3.1.2 LR data downsampled following a 2D protocol

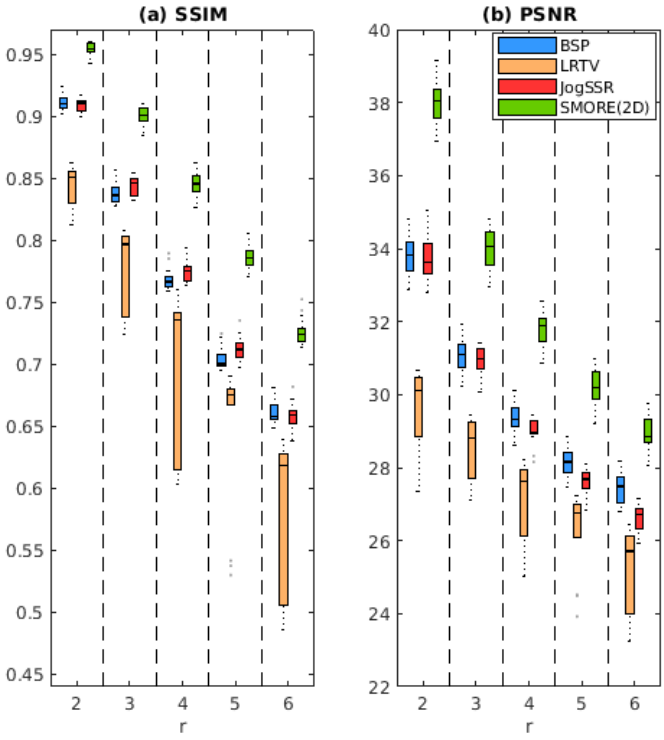


Figure 4.6: Evaluation of accuracy for super-resolution results from LR MRI downsampled with a 2D protocol: (a) SSIM values and (b) PSNR comparing to ground truth for the restored images using cubic b-spline interpolation (blue), LRTV [35] (orange), JogSSR [102] (red), and SMORE(2D) (green). Higher value indicates better accuracy.

To simulate LR 2D MRI, we Gaussian blurred and downsampled the $1 \times 1 \times 1$ mm HR images by factors of $r = \{2, \ldots, 6\}$ in the z-axis to simulate thick-slice MR images. We then restored these images to an isotropic digital resolution of $1 \times 1 \times 1$ mm using cubic B-spline interpolation (BSP), LRTV [35], JogSSR [102], and SMORE(2D). A visual comparison is shown in Fig. 4.5 for $r = 4$ and $r = 6$. Evaluation of these results using SSIM and PSNR is shown in Fig. 4.6 and evaluation using S3 is shown in Fig. 4.7. We observe that SMORE(2D) outperforms all other methods visually and also quantitatively with both SSIM and PSNR. Sharpness yields a similar result wherein LRTV [35] has the highest

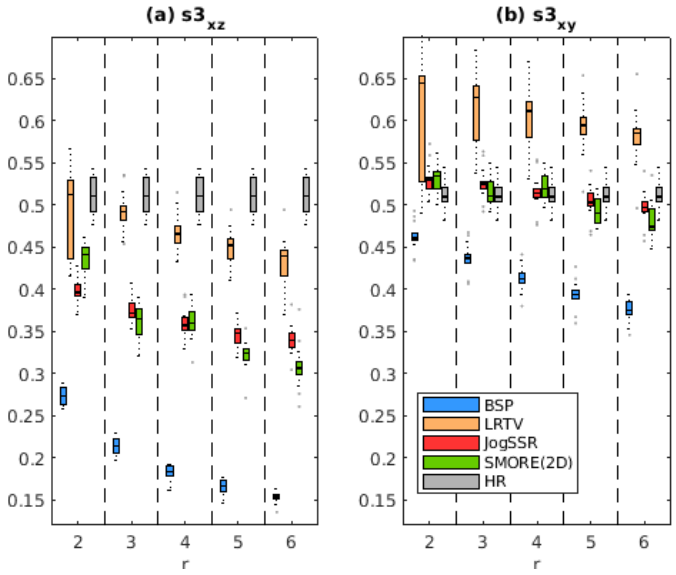


Figure 4.7: Evaluation of sharpness for super-resolution results from LR MRI downsampled with a 2D protocol: S3 sharpness values in (a) x-z plane (LR) and (b) x-y plane (HR) for the restored images using cubic b-spline interpolation (blue), LRTV [35] (orange), JogSSR [102] (red), SMORE(2D) (green), and HR ground truth (grey). Higher value indicates sharper edges.

sharpness while both JogSSR [102] and SMORE(2D) yield higher sharpness in comparison to BSP. As in the 3D simulation, the method of LRTV [35] is clearly emphasizing artifacts rather than true edges (see Fig. 4.5). We also see from Fig. 4.5 that both the SSIM and PSNR measures for SMORE(2D) at $r = 3$ are comparable to the BSP result at $r = 2$. This implies that it may be possible to acquire thicker slices ($r = 3$), apply SMORE(2D), and get comparable SSIM and PSNR as interpolated thinner slices ($r = 2$), which could lead to faster scan times.

To evaluate the effect of SAA, we also perform the experiment with $\text{SAA}_{\text{Bool}} = \text{False}$, which we show in Sec. 4.3.3.

4.3.2 Robustness to noise

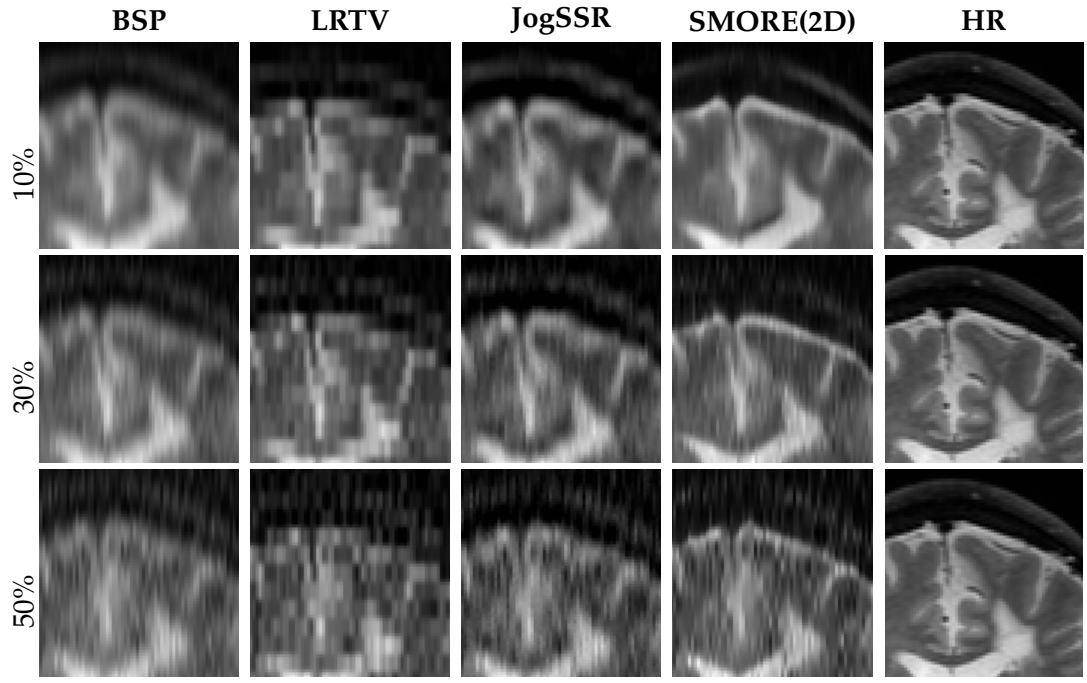


Figure 4.8: Results from MRI downsampled with a 2D protocol and Rician noise added: Zoomed coronal views of the $1 \times 1 \times 6$ mm 2D noisy LR input image interpolated with a cubic b-spline, restored using methods from LRTV [35], JogSSR [102], and SMORE(2D), and the HR ground truth image.

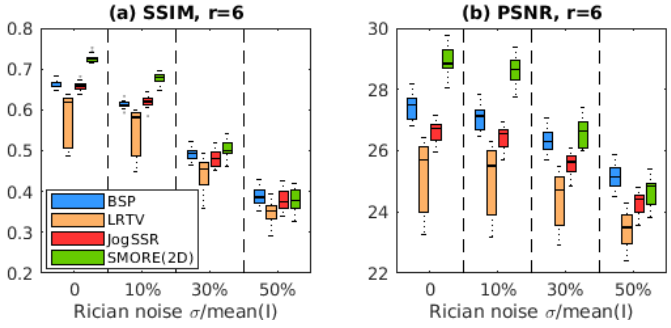


Figure 4.9: Robustness to noise: (a) SSIM and (b) PSNR for super-resolution results from noisy $1 \times 1 \times 6$ mm MRI compared to ground truth, with the restored images using cubic b-spline interpolation (blue), LRTV [35] (orange), JogSSR [102] (red), and SMORE(2D) (green).

The architecture of EDSR mimics a high-pass filter, which suggests that SMORE might not be robust to noise. Although LR MR images generally have low noise (because they have large voxels), we evaluated the impact of

noise on SMORE, potentially for applications to faster imaging scenarios such as dynamic MRI. We experimented on simulated $r = 6$ 2D input images and added Rician noise with standard deviation σ of 10%, 30%, 50% of the mean intensity. The noisy image I_σ are simulated from clean image I using Rician noise model $I_\sigma = \sqrt{(I + \eta_1)^2 + \eta_2^2}$, with $\eta_1, \eta_2 \sim \mathcal{N}(0, \sigma^2)$. Example results are shown in Fig. 4.8 and the performance measures SSIM and PSNR are shown in Fig. 4.9. It is observed that SMORE(2D) is robust to noise levels of 10% and 30% but fails at a noise level of 50%. For 50% noise, BSP gives better results. This is because interpolation can have a smoothing effect as it does not recover high frequency information, and thus it reduces noise. On the other hand, super-resolution tends to sharpen edges, which increases noise. However, a large level of noise like 30% or 50% is uncommon in structural MRI except for ultra-fast acquisitions. The results for 10% noise demonstrates the robustness of SMORE for a normal level of noise encountered in structural MRI.

4.3.3 Impact of SAA

There is a boolean variable SAA_{Bool} in Algorithm 3 that controls whether to train and apply SAA for SMORE(2D). In order to show the effect of the SAA network, we performed a comparison on simulated $1 \times 1 \times 6$ mm 2D MRI data from Sec. 4.3.1. We ran SMORE(2D) twice, first with SAA enabled, and then with it disabled. An example of the sagittal slices is shown in Fig. 4.10. We can see that there is aliasing in Fig. 4.10(b) (red arrow) while Fig. 4.10(c) removes the aliasing. Given the HR ground truth for this simulation, we computed the

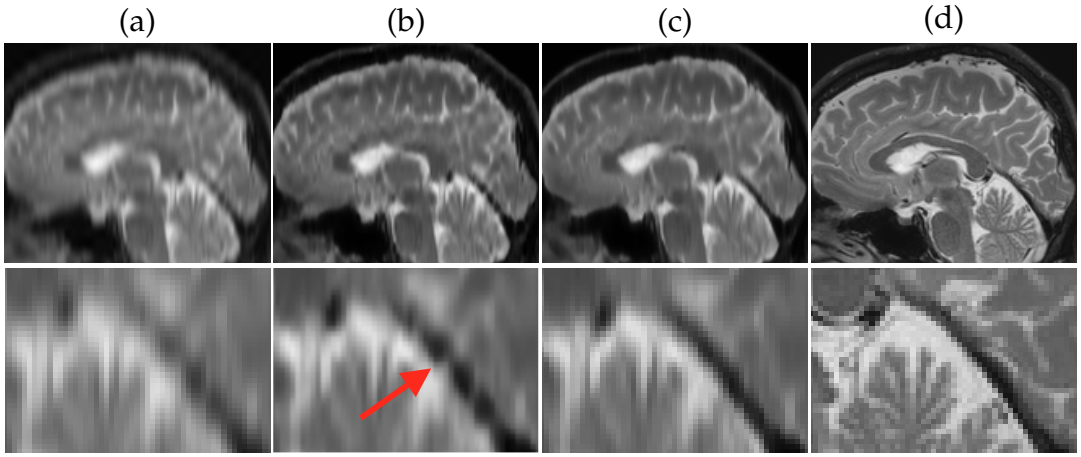


Figure 4.10: Impact of SAA on simulated LR image: Sagittal views of the $1 \times 1 \times 6$ mm MR T2-w image (simulated with a 2D protocol) restored with (a) BSP, (b) SMORE(2D) with $\text{SAA}_{\text{Bool}} = \text{False}$ and (c) $\text{SAA}_{\text{Bool}} = \text{True}$, and (d) HR ground truth. The red arrow points to aliasing in (b).

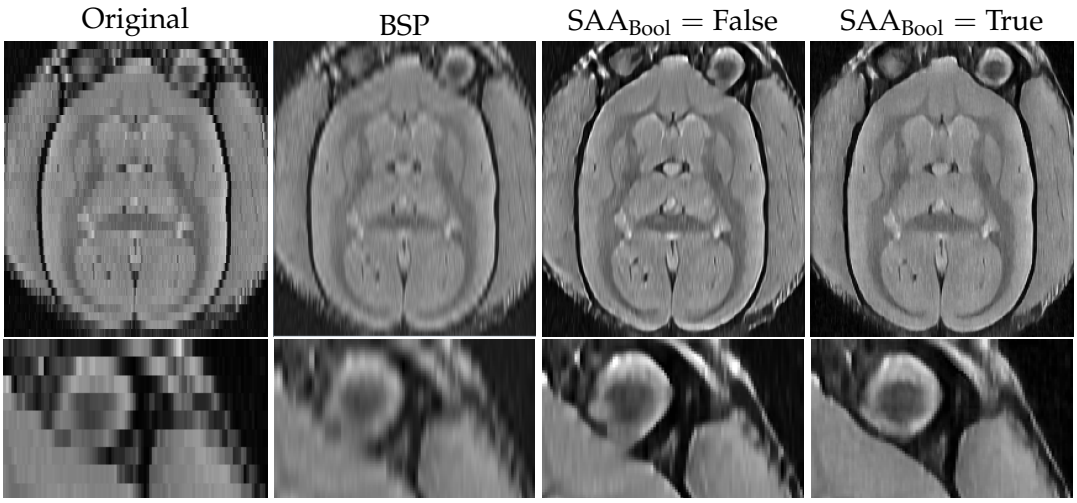


Figure 4.11: Impact of SAA on acquired LR image: x-z plane slice of the $0.15 \times 0.15 \times 1$ mm original marmoset MR PD image (acquired with 2D protocol), restored images using BSP, SMORE without ($\text{SAA}_{\text{Bool}} = \text{False}$) and with SAA ($\text{SAA}_{\text{Bool}} = \text{True}$).

SSIM, PSNR, and mean squared error (MSE) for these 14 subjects. The PSNR and MSE of SMORE(2D) with $\text{SAA}_{\text{Bool}} = \text{True}$ and False are not significantly different. However, for SSIM, SMORE(2D) with $\text{SAA}_{\text{Bool}} = \text{True}$ gives 1.332%

worse mean SSIM even though the results look cleaner with anti-aliasing. Since the computation of SSIM involves prefiltering, subtle aliasing artifacts may not be captured with the SSIM measure. Considering the visual quality, we believe that setting $\text{SAA}_{\text{Bool}} = \text{True}$ is a reasonable choice when aliasing artifacts are obvious.

We also performed an experiment on a marmoset PD MR image with acquired resolution of $0.15 \times 0.15 \times 1$ mm. The results are shown in Fig. 4.11. It shows that without SAA ($\text{SAA}_{\text{Bool}} = \text{False}$), the results still have sharp edges, yet they appear to suffer from remaining aliasing artifacts.

4.3.4 Choice of M and computation time

4.3.4.1 Computation time

The computation time of SMORE consists of training time and testing time. The training time does not increase with the data augmentation angles N. The testing time is proportional to the self-ensemble angles M.

In practice, training the model for one subject based on pre-trained models from an arbitrary data set takes less than 10 minutes for a Nvidia Tesla K80 GPU, while training from scratch takes about 40 minutes. The computation time for applying the network is proportional to the size of resultant image and M. For an image g of size $180 \times 240 \times 240$ and $M = 1$, the testing time for applying a network is approximately 5 minutes. Therefore, the total time to train and apply the SMORE(3D) network is approximately 15 minutes with $M = 1$ and 45 minutes with $M = 7$.

For SMORE(2D), the computation time with $\text{SAA}_{\text{Bool}} = \text{False}$ is the same

as SMORE(3D);when $\text{SAA}_{\text{Bool}} = \text{True}$, the computation time is nearly twice that of SMORE(3D) due to the additional SAA network.

4.3.4.2 Choice of M

In our conference papers [43, 44], both SAA and SSR are done with $M = 2$. Assuming that the z-axis is the LR axis, SAA is first performed on the y-z plane (only for MRI with 2D protocols) and SSR on the x-z plane, denoted as $\hat{f}^0$. Next, SAA is performed on the x-z plane (only for MRI with 2D protocols) and SSR on the y-z plane, denoted as $\hat{f}^{\pi/2}$. The two resulting 3D volumes are combined using FBA [104]. However, we found that the results of $M = 1$ are also visually good. In fact, all experiments reported in this chapter up to this point presented $\hat{f}^0$. In order to demonstrate the effect of M, we computed $\hat{f}^0$ and $\hat{f}^{\pi/2}$ as well as the FBA result (FBA2) which combines them. We also computed SMORE on an additional five directions to get $\{\hat{f}^0, \hat{f}^{\pi/12}, \hat{f}^{\pi/6}, \hat{f}^{\pi/4}, \hat{f}^{\pi/3}, \hat{f}^{5\pi/12}, \hat{f}^{\pi/2}\}$ and computed the FBA result (FBA7) which combines these seven results.

We carried out a comparison using the simulated dataset from Sec. 4.3.1 with $r = 6$ and applied both 3D and 2D protocols. For both protocols, it is difficult to find visual differences for different choices of M. Nevertheless, we computed both SSIM and PSNR and show them in Figs. 4.12 and 4.13.

From Fig. 4.12, we found that in the 3D MRI experiment, the results of FBA2 have the best mean value of SSIM, while the results of FBA7 have the best mean value of PSNR. However, the differences between these four methods are small. In order to study the significance, we performed sign test

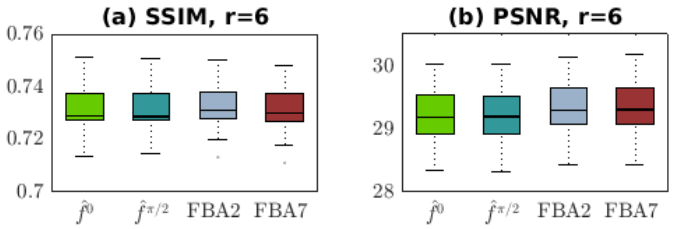


Figure 4.12: Choice of M for FBA (3D): Box plots showing (a) SSIM values and (b) PSNR values compared to the ground truth for the restored images using SMORE(3D) that does SSR on the x-z plane, denoted $\hat{f}^0$ (green); SMORE(3D) performed on orthogonal direction, denoted $\hat{f}^{\pi/2}$ (ocean); the FBA result, FBA2, which combines the results from $\hat{f}^0$ and $\hat{f}^{\pi/2}$ (purple); and the FBA result, FBA7, which combines the results from seven directions (wine).

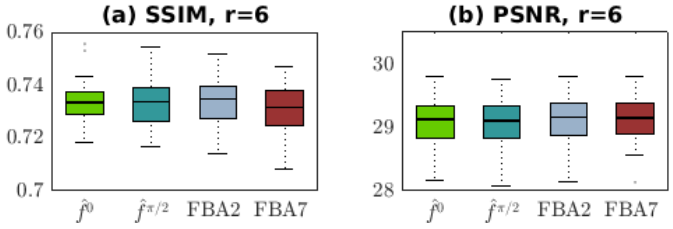


Figure 4.13: Choice of M for FBA (2D): Box plots showing (a) SSIM values and (b) PSNR values compared to ground truth for the restored images using SMORE(2D), denoted $\hat{f}^0$ (green); SMORE(2D) performed on the orthogonal direction, denoted as $\hat{f}^{\pi/2}$ (ocean); the FBA result, FBA2, which combines the results from $\hat{f}^0$ and $\hat{f}^{\pi/2}$ (purple); and the FBA result, FBA7, which combines results from seven directions (wine).

and judged significance at $p < 0.01$ between each pair of these methods. The null hypothesis is that the median of the differences of the paired samples is zero. We found that: 1) for $\hat{f}^0$ and $\hat{f}^{\pi/2}$, the median values of SSIM and PSNR were not significantly different. 2) for FBA2 and FBA7, the median values of PSNR were not significantly different, but FBA2 has significantly better median SSIM; 3) for FBA7 and $\hat{f}^0/\hat{f}^{\pi/2}$, the median values of SSIM and PSNR were not significantly different; 4) for FBA2 and $\hat{f}^0/\hat{f}^{\pi/2}$: the median values of SSIM were not significantly different, but FBA2 has significantly better median PSNR. Therefore, for 3D MRI, the use of $M = 2$ and FBA2 is

recommended if PSNR is of primary concern; $M = 1$ is a reasonable choice if computation time is of primary concern.

From Fig. 4.13, we found that in the 2D MRI experiment, the results of FBA2 have the best mean value of SSIM and PSNR. We performed the same significance test as in 2D MRI experiment and found that neither SSIM nor PSNR of FBA2 has statistically better median than $\hat{f}^0$ and $\hat{f}^{\pi/2}$. Therefore, use of $M = 1$ is recommended for SMORE(2D) since this will save computation time.

4.4 Conclusion and Discussion

This chapter described a deep learning-based self-supervised anti-aliasing (SAA) and self-supervised super-resolution (SSR) algorithm SMORE. In contrast to most other deep learning-based super-resolution (SR) methods for MRI [101], SMORE does not need external training data, which makes it more applicable to a wide variety of acquired MRI pulse sequences. Compared with interpolation and other SSR methods [35, 102], SMORE demonstrates significant improvements in SR accuracy and shows robustness under low levels of Rician noise. The experiments were performed on both simulated and real acquired LR data including T2-w and PD MRI, without any modification on SMORE. In our experiments, the largest SR ratio, r, is 6.6667 (see Fig. 4.10), while other SSR methods [35, 102] for MRI only demonstrated their applications on data with small r, usually no greater than 3.

General SR problems with well-established training datasets in computer vision have been discussed quite a bit in the literature, e.g., in NTIRE CVPR

SR challenges [36]. An important distinction between the computer vision application and the MRI application is that external training data is much more difficult to obtain in MRI than in natural images. Considering this distinction, the biggest advantage of SMORE is the fact that SMORE does not need external training data. In addition to that, SMORE needs no preprocessing step other than N4 inhomogeneity correction [94]. When the acquisition protocol and acquisition parameters like slice thickness and slice separation are known, SMORE only needs two parameters—SAA_{Bool} and M—for which we provide suggested values in Sec. 4.3.3 and Sec. 4.3.4. Also, we used the same deep network hyper-parameter settings for all experiments in this chapter. All these properties are desirable for easy application to new MRI datasets.

Some additional evaluations of SMORE are performed in the next chapter, including SMORE on real acquired T2 FLAIR brain images and T2-w tongue images with tumors, SMORE on real acquired LR/HR paired cardiac data, and the effect of SMORE on image segmentation. Other additional evaluation of SMORE could be performed in the future. First, SMORE uses EDSR as the network architecture since it was evaluated as the state-of-art SR network by extensive comparisons in other literature [36, 107]. These evaluations were performed on natural images, however, not on MRI, and it is possible that there might be better SR network architectures for MRI data. Fortunately, EDSR can be easily replaced under the framework of SMORE if a better SR network becomes available. The second limitation concerns the non-reference metric used in this evaluation. Here, we computed S3 [112], which is a sharpness metric that might not reflect the actual perceived image quality.

We are aware of other open source no-reference perceptual quality metrics such as NIQE [113] and BRISQUE [114], which are designed to correspond to human perception. However, these models were trained on natural images, and not suitable for MRI data. Future work should include more experiments to address these noted limitations.

In conclusion, the SMORE method was described and shown to perform well in comparison to other methods. It produces results that are more accurate than interpolation, and does not need any external training data. This makes it a potentially useful preprocessing step for many MR image analysis tasks. More applications of SMORE as well as a task-specific evaluation (segmentation) are discussed in Chapter 5, which also includes discussion about the limitations of SMORE.

Chapter 5

Application of SMORE on various MRI datasets

5.1 Introduction

In the previous chapter, we introduced the Synthetic Multi-Orientation Resolution Enhancement (SMORE) [44, 43] algorithm. SMORE does not use external training data, there are no parameters to tune besides those hypeparameters used for deep learning training, and the only pre-processing that is required is N4 intensity inhomogeneity correction [94]. In this chapter, we use four applications to demonstrate the potential of SMORE in both research and clinical scenarios. The first application is on T2 FLAIR MR brain images acquired from multiple sclerosis (MS) patients. MS is an auto-immune disease in which myelin, the protective coating of nerves, is damaged and can be visualized as hyperintense lesions in FLAIR images. We show that visualization of MS lesions using SMORE is better than that obtained using cubic b-spline interpolation and JogSSR.

The second application is on cardiac MRI where we explore the visualization of myocardial scarring from cardiac left ventricular remodeling after myocardial infarction. Characterizing such scarring is important factor in assessing the long-term clinical outcome after myocardial infarction [115] and it is challenging due to the competing requirements of high-resolution imaging and rapid scanning due to cardiac motion and breathing. We show improved visualization of such scars when using SMORE.

The third application of SMORE is on multi-orientation MR images of the tongue in tongue tumor patients. Because of the involuntary requirement to swallow during lengthy MR scans, acquisition times are very limited—less than 3 minutes—in tongue imaging. A previous approach to obtaining super-resolution in the tongue used a computational combination of axial, sagittal, and coronal image stacks, each obtained in a separate stationary phase and registered together [31]. We demonstrate how the use of SMORE on a single acquisition is comparable to the result of combining three acquisitions.

The fourth application of SMORE is on brain ventricle labeling in subjects with normal pressure hydrocephalus (NPH). NPH is a brain disorder usually caused by disruption of the cerebrospinal fluid (CSF) flow, leading to ventricle expansion and brain distortion. Having accurate parcellation of the ventricular system into its sub-compartments could potentially help in diagnosis and surgical planning in NPH patients [116]. Both visual and selected quantitative metrics of resolution enhancement are demonstrated.

In this chapter, we make two important contributions about the implementation and utility of SMORE. In order to show the versatility of SMORE,

we present results on four MRI datasets from different pulse sequences and different organs, with three of them being real acquired low resolution MR datasets. Then we demonstrate that the proposed SR algorithm yields improvements not only in apparent image quality but, in the fourth experiment, show quantitative improvements when SMORE is applied as a preprocessing step for a segmentation task.

5.2 Application 1: visual enhancement for MS lesions

In this experiment, we test whether super-resolved T2 FLAIR MR images can give better visualization of white matter lesions in the brain than the acquired images. The 33 T2 Flair MR images were acquired from multiple sclerosis (MS) subjects using a Philips Achieva 3T scanner with a 2D protocol and the following parameters: $0.828 \times 0.828 \times 4.4$ mm, TE = 68 ms, TR = 11 s, TI = 2.8 s, flip angle = 90°, turbo factor = 17, acquisition time = 176 s. We performed cubic b-spline interpolation, JogSSR [102], and SMORE(2D) on the data using a $0.828 \times 0.828 \times 0.828$ mm digital grid.

We first show a visual comparison on the regions of white matter lesions in axial, sagittal, and coronal slices for the three methods. Fig. 5.1 shows an example of T2 FLAIR images reconstructed from the acquired resolution of $0.828 \times 0.828 \times 4.4$ mm input image onto a $0.828 \times 0.828 \times 0.828$ mm digital grid using the three methods. On these images, MS lesions appear as bright regions in the brain's white matter. We can see that both JogSSR and SMORE(2D) give sharper edges than interpolation and the SMORE(2D) result looks more

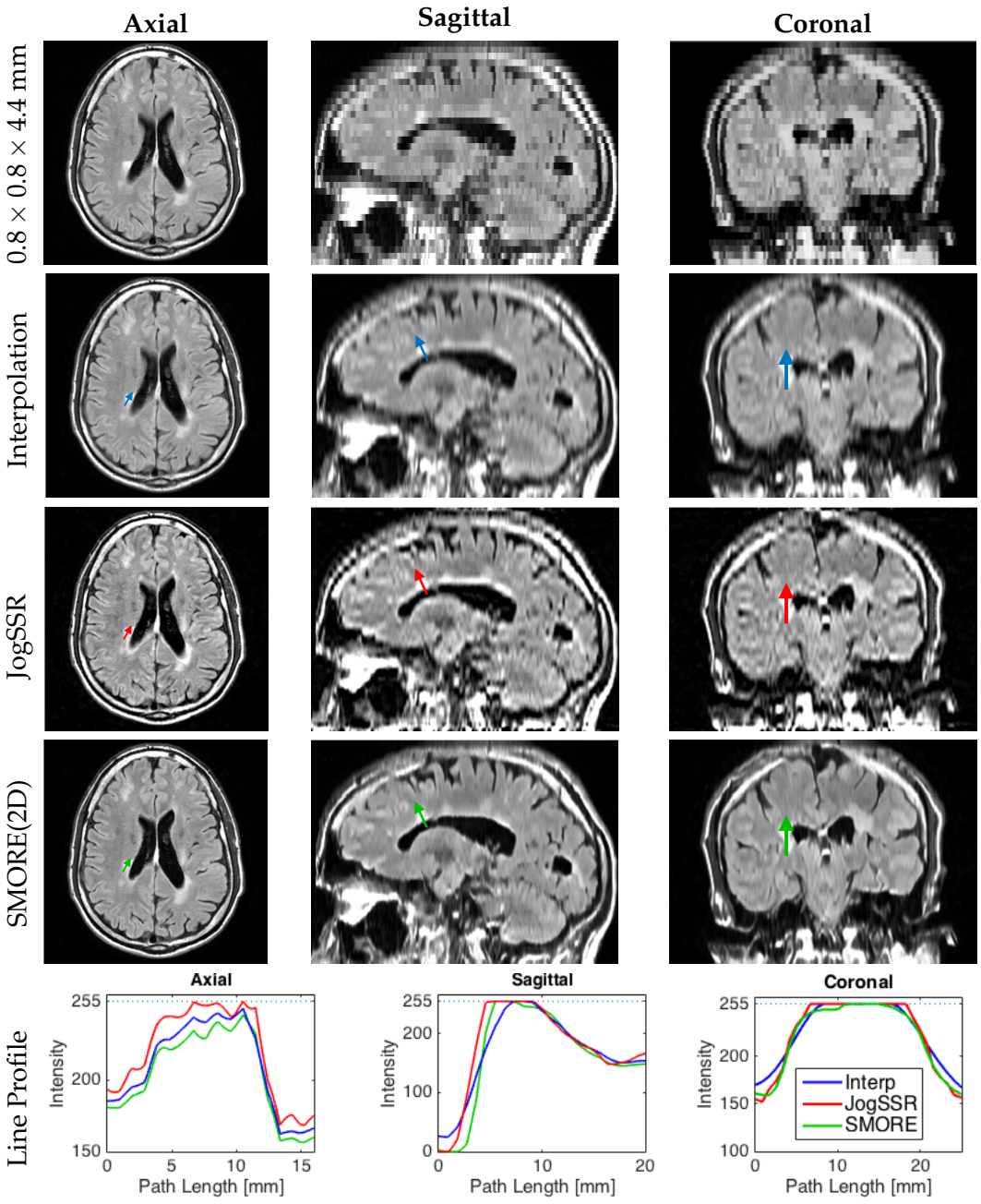


Figure 5.1: T2 Flair MRI from an MS subject: Axial, sagittal, and coronal views of the acquired $0.828 \times 0.828 \times 4.4$ mm image, and the reconstructed volumes with $0.828 \times 0.828 \times 0.828$ mm digital grid through cubic b-spline interpolation, JogSSR, and SMORE(2D). In each view, we pick a path across lesions, shown as colored arrows in the images, and plot the line profiles of the three methods in the same plot on the bottom of each view.

realistic than JogSSR. This is in part because JogSSR does not carry out anti-aliasing, which allows aliasing artifacts, which are seen in the original and interpolated images, to remain. We note that in Fig. 5.1, SMORE also enhances resolution in the axial slice slightly, which is originally 0.828×0.828 mm. Although we apply super-resolution in the through-plane, structures like edges that pass through-plane slices obliquely also gets enhanced, permitting in-plane edges to also be enhanced.

Aside from the visual impression of performance differences gleaned from looking at the images directly, we also plot 1D intensity profiles of the three methods across selected paths through different lesions. Each reconstructed image in Fig. 5.1 contains a small colored arrow. These arrows depict the line segment and direction over which intensity profiles shown in the bottom row of the figure are extracted. For example, the three colored arrows in the axial images of the first column yield the profiles on the bottom right graph. These axial profiles show that other than some differences in overall intensity, the resolutions of the methods appear to be very similar. This is to be expected since the axial image already has good resolution. The profiles through the ventricle and lesion in the sagittal orientation are different. Both super-resolution approaches show a steeper edge than the interpolated image (although the JogSSR result is inexplicably shifted relative to the true position of the edge). The profiles from the lesion in the coronal images show a similar property—steeper edges from the super-resolution approaches. Overall, the selected intensity profiles suggest resolution enhancement from both SMORE(2D) and JogSSR.

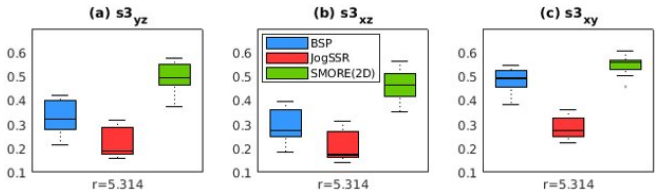


Figure 5.2: Evaluation of sharpness for super-resolution results from LR acquired with a 2D protocol: S3 sharpness values in (a) the y-z plane (LR), (b) the x-z plane (LR) and (c) the x-y plane (HR) for the restored images using cubic b-spline interpolation (blue), JogSSR [102] (red), and SMORE(2D) (green).

Furthermore, we show the S3 sharpness feature for these results in Fig. 5.2 where it is observed that SMORE is better than the other methods. The improved sharpness in the in-plane (x-y plane) is an important feature of super-resolution which should not be considered anomalous. We see in Fig. 5.2(c) that this is true for the comparison with BSP and JogSSR results. Also from Fig. 5.3, we can see partial volume artifacts in the original and BSP images, but much less in the SMORE result. We compared the in-plane (x-y plane) S3 values of the original images and the SMORE results[1]. The mean x-y plane S3 for the original images is 0.5361, while the mean value for the SMORE result is 0.5482 [2]. The reason for the improved sharpness in in-plane slices is that features such as edges that pass through the image plane in oblique orientations do experience blurring from the through-plane impulse response during image acquisition, and effective super-resolution will reduce the blurriness of

[1]For through-plane slices, the original images have different digital resolution from BSP images and SR results. Thus, comparing their S3 values for through-plane slices is meaningless.

[2]Although a Wilcoxon signed-rank test was performed and reported in [117], we later realized that the this statistic should not be used since we cannot prove the difference of paired samples are distributed symmetrically, which is the assumption of Wilcoxon signed-rank test. The correct test is a sign test, but we cannot carry this out as the original data have been lost as of this writing.

those features. This explains why S3 should be larger even in the x-y plane.

5.3 Application 2: visual enhancement of scarring in cardiac left ventricular remodeling

In this experiment, we test whether super-resolved images can give better visualization of the scarring caused by left ventricular remodeling after myocardial infarction than the acquired images. We acquired two T1-weighted MR images from an infarcted pig, each with a different through-plane resolution. One image, which serves as the HR reference image, was acquired with resolution equal to $1.1 \times 1.1 \times 2.2$ mm, and then it was sinc interpolated on the scanner (by zero padding in k-space) to $1.1 \times 1.1 \times 1.1$ mm. The other image was acquired with resolution equal to $1.1 \times 1.1 \times 5$ mm. Both of these images were acquired with a 3D protocol, inversion time = 300 ms, flip angle = 25°, TR = 5.4 ms, TE = 2.5 ms, and GRAPPA acceleration factor $R = 2$. The HR reference image has a segmented centric phase-encoding order with 12 k-space segments per imaging window (heart beat), while the LR subject image has 16 k-space segments.

In our experiment, we performed sinc interpolation, JogSSR [102], and SMORE(3D) on the $1.1 \times 1.1 \times 5.0$ mm data using a $1.1 \times 1.1 \times 1.1$ mm digital grid. These images were then rigidly registered to the reference image for comparison, shown in Fig. 5.3. We are interested in the regions of thinning layer of midwall scar between the endocardial and epicardial layers of normal myocardium and the thin layer of normal myocardium between the scar and epicardial fat. These two regions of interest are cropped and zoomed to

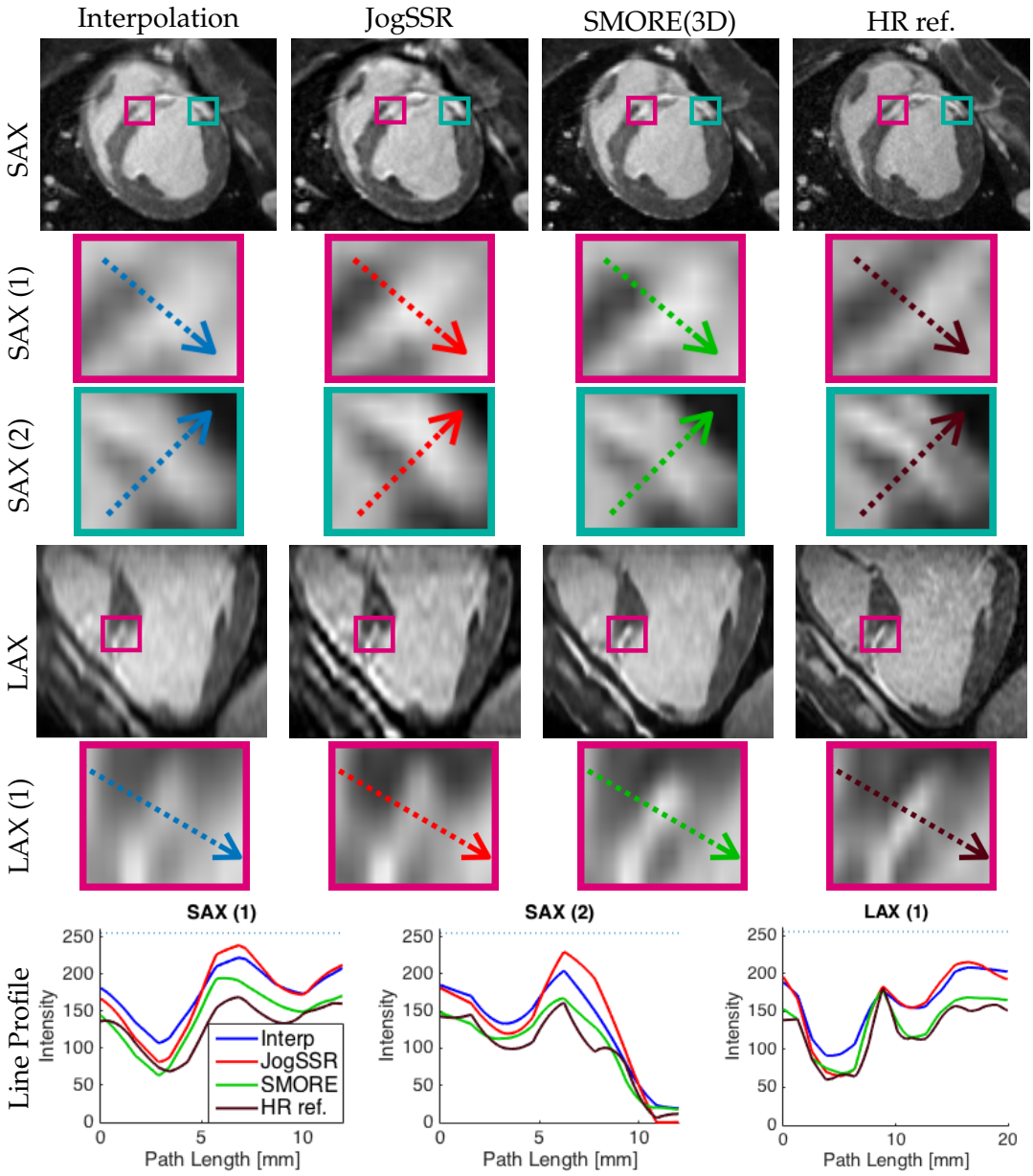


Figure 5.3: Late gadolinium enhancement (LGE) from an infarct swine subject: Short-axis (SAX) and long-axis (LAX) views arranged in columns using $1.1 \times 1.1 \times 1.1$ mm digital grid: output of 1) sinc-interpolation, 2) JogSSR and 3) SMORE(3D) for the subject LR image acquired at $1.1 \times 1.1 \times 5$ mm; 4) sinc-interpolated HR reference image for comparison acquired at $1.1 \times 1.1 \times 2.2$ mm. SAX(1) and LAX(1) boxes contain a thinning layer of enhanced midwall scar between endo- and epi layers of normal myocardium (hypo-intense). SAX(2) boxes contain a thin layer of normal myocardium (hypo-intense) between scar and epicardial fat (both hyper-intense). In each box, we pick a path across the region of interest, shown as colored arrows, and plot the profiles in the last row.

show the details. Specifically, a thin layer of the midwall scar between the endocardium and epicardium of normal myocardium appears as bright strip in the magenta boxes, and a thin layer of normal myocardium between scar and epicardial fat appears as dark strip in the cyan boxes. They are zoomed in to show the details below the short-axis (SAX) slices with acquired resolution of 1.1×1.1 mm and long-axis (LAX) slices with originally acquired resolution of 1.1×5 mm for the first three columns, or 1.1×2.2 mm for the column of "HR ref". Each zoomed box contains a colored arrow which depicts a line segment. The corresponding line profiles are shown on the bottom.

As seen in the long-axis (LAX) images and zoomed regions, the borders between normal myocardium, enhanced scar and blood are clearer in SMORE(3D) compared with JogSSR and interpolation. The intensity profile of SMORE(3D), the green line shown in the magenta box marked "LAX (1)", very closely matches that of the HR reference image. For the short-axis (SAX(1) and SAX(2)), the resolution was already high and there is less to be gained. Nevertheless, it is apparent that the image clarity is slightly improved by SMORE(3D) while faithfully representing the patterns from the input images.

We computed the SSIM and PSNR between each method and HR reference image. The result is shown in Table 5.1. SMORE gives the best SSIM yet worse PSNR than sinc interpolation. Note that the registration cannot be perfect among different sets of cardiac images, due to motion or changing physiological state. When computing SSIM, images are prefiltered. Therefore, SSIM is less sensitive to image distortion. On the other hand, PSNR is a measure of noise level and SMORE does not explicitly consider noise reduction. This

might be why the SMORE result has better SSIM but worse PSNR. Also this evaluation was done on only one pair of LR/HR data, and is not statistically informative.

	sincInterp	JogSSR	SMORE
SSIM	0.5070	0.4770	0.5146
PSNR	25.8816	24.4142	25.3002

Table 5.1: SSIM and PSNR of sinc interpolation, JogSSR [102], and SMORE(3D) on LGE from an infarct swine subject.

5.4 Application 3: multi-view reconstruction

In this experiment, we test whether a super-resolved image from a single acquisition can give a comparable result to a multi-view super-resolution image reconstructed from three acquisitions. MR images of the tongue were acquired from normal speakers and subjects who had tongue cancer surgically resected (glossectomy). Scans were performed on a Siemens 3.0 T Tim Treo system using an eight-channel head and neck coil. A T2-weighted Turbo Spin Echo sequence with an echo train length of 12, TE = 62 ms, and TR = 2500 ms was used. The field-of-view (FOV) was 240 $\times$ 240 mm with an image size of 256 $\times$ 256. Each dataset contained a sagittal, coronal, and axial stack of images containing the tongue and surrounding structures. The image size for the high-resolution MRI was 256 $\times$ 256 $\times$ z, where z ranges from 10 to 24, with 0.9375 $\times$ 0.9375 mm in-plane resolution and 3 mm slice thickness. The datasets were acquired at a rest position and the subjects were required to remain still for 1.5–3 min for each orientation. For each subject, the three axial, sagittal, and coronal acquisitions were interpolated onto a 0.9375 $\times$ 0.9375 $\times$ 0.9375 mm

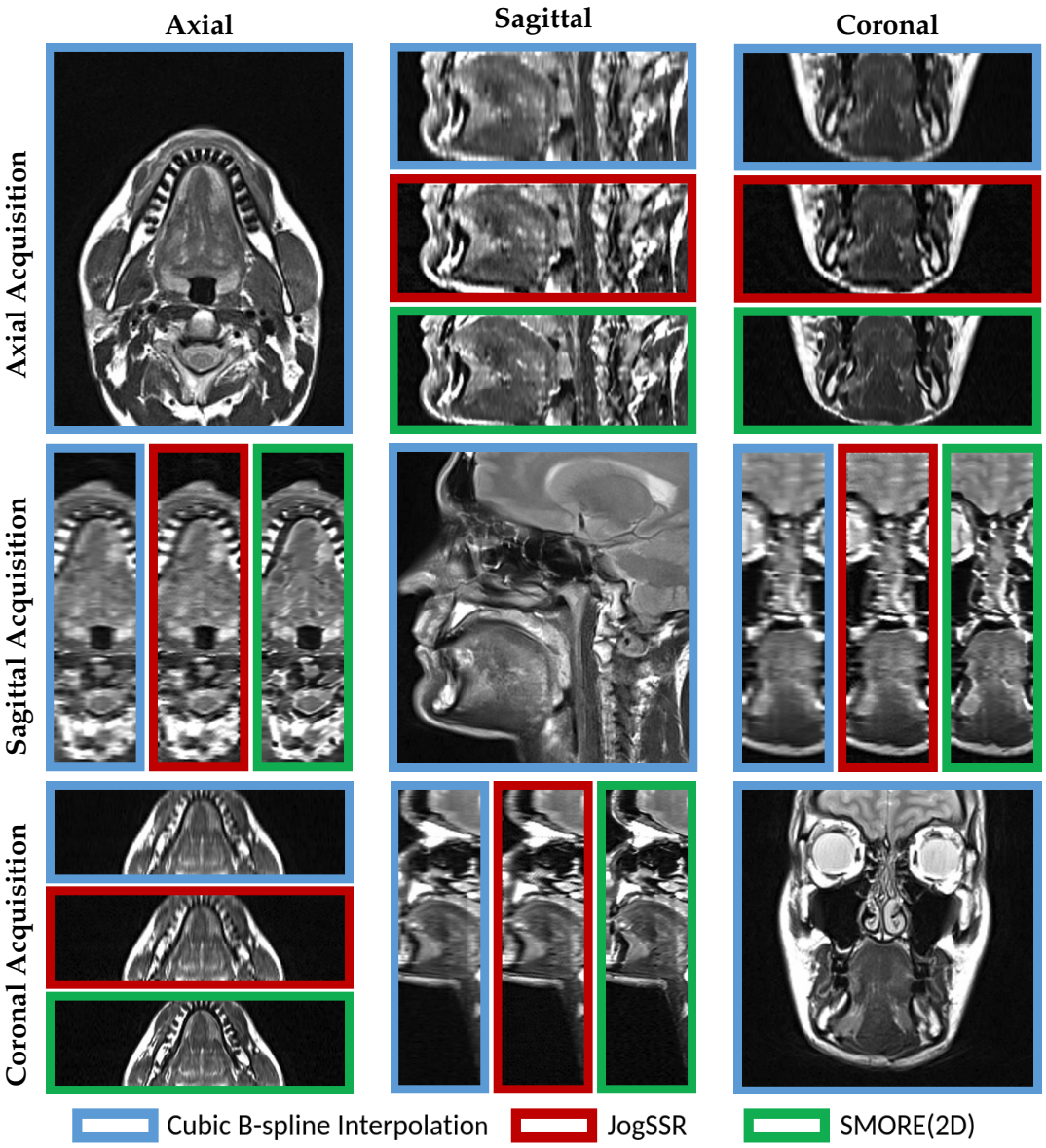


Figure 5.4: T2w MRI from a tongue tumor subject: Axial, sagittal, and coronal views of the three acquisitions in axial, sagittal, and coronal planes (not registered). We show the through-plane views of the resolved volumes with isotropic digital resolution that result from cubic b-spline interpolation (blue boxes), JogSSR (red boxes), and our SMORE(2D) (green boxes). The in-plane views are only shown with interpolation results since they are already HR slices.

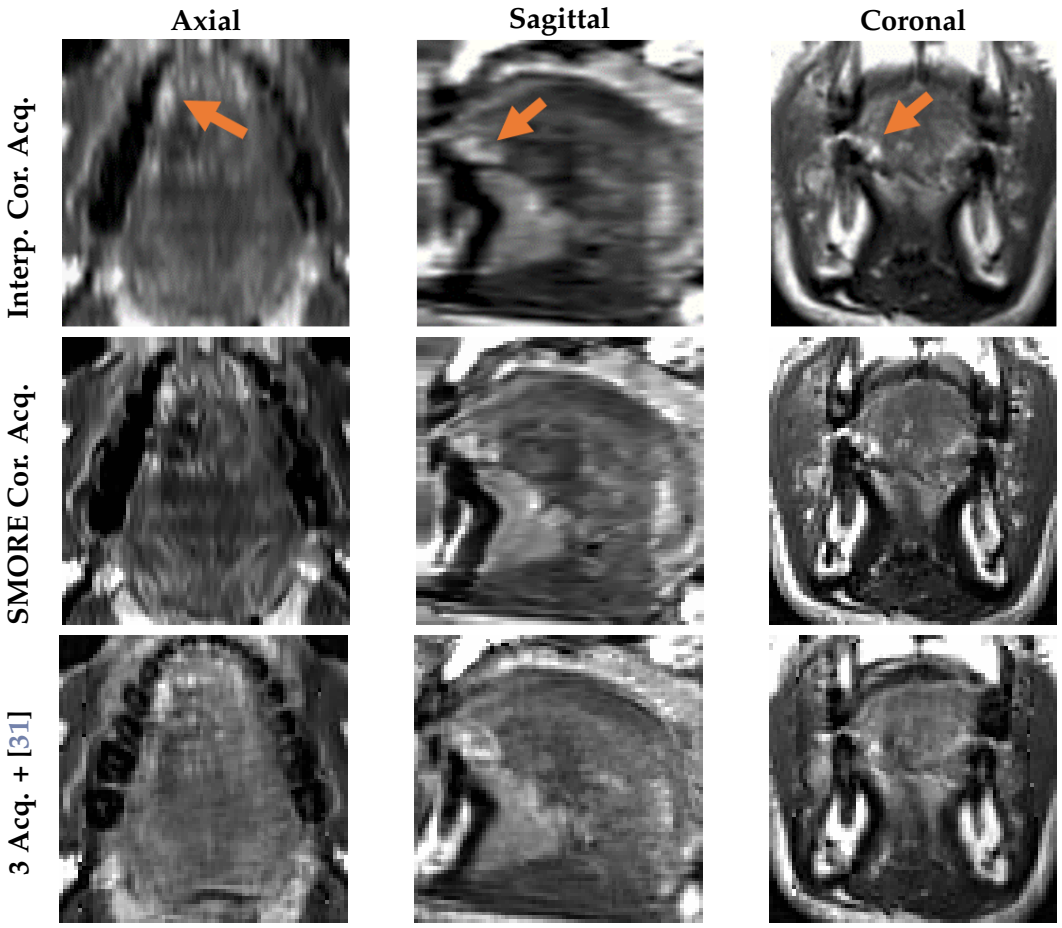


Figure 5.5: Comparison between SMORE(2D) and multi-view reconstruction for a tongue tumor subject: Axial, Sagittal, and Coronal views of the tongue region in cubic b-spline interpolation and SMORE(2D) results for a single coronal acquisition, and multi-view reconstructed image [31] using three acquisitions. The arrows point out the bright looking scar tissue from a removed tumor.

digital grid and N4 corrected [94].

We applied both JogSSR and SMORE(3D) on single acquisitions to compare to the multi-view super-resolution reconstruction. The multi-view reconstruction algorithm we used for comparison is an improved version of the algorithm described in Woo et al. [31]. This approach takes three interpolated image volumes, aligns them using ANTs affine registration [118] and SyN

deformable registration [119], and then uses a Markov random field image restoration algorithm (with edge enhancement) to reconstruct a single HR volume.

Tongue data with 0.9375×0.9375 mm in-plane resolution and 3 mm through-plane resolution were acquired with in-plane view of axial, sagittal, and coronal. They are shown in Fig. 5.4 on a $0.9375 \times 0.9375 \times 0.9375$ mm digital grid after both cubic b-spline interpolation (blue boxes) interpolation, JogSSR (red boxes), and SMORE(2D) (green boxes). The in-plane views are only shown for interpolation since they are already HR slices. In through-plane views, SMORE(2D) always gives visually better results than both interpolation and JogSSR. In particular, we can see the edges are sharper in SMORE(2D) and no artificial structures are created.

A comparison of interpolation and SMORE(2D) (where each used only the coronal image) and the multi-view reconstruction (which used all three images) is shown in Fig 5.5. The arrows point out at the bright pathology region—i.e., scar tissue formed after removing a tumor. We can see that SMORE has visually better resolution than the interpolated image, but several places within the multi-view reconstruction have visually better detail. On the other hand, the pathology region in the multi-view reconstruction appears to be somewhat degraded in appearance over both the SMORE(2D) and interpolation result. We believe that this loss of features may be caused by regional mis-registration between the three acquisitions.

5.5 Application 4: brain ventricle parcellation

This experiment demonstrates the effect of super-resolution on brain ventricle parcellation and labeling using the Ventricle Parcellation Network (VParNet) described in Shao et al. [120]. In particular, we test whether super-resolved images can give better VParNet results than images from either interpolation or JogSSR. The data for this experiment are from an NPH data set containing 95 T1-w MPRAGE MRIs (age range: 26–90 years with mean age of 44.54 years). They were acquired on a 3T Siemens scanner with scanner parameters: TR = 2110 ms, TE = 3.24 ms, FA $= 8^\circ$, TI = 1100 ms, and voxel size of $0.859 \times 0.859 \times 0.9$ mm. There are also 15 healthy controls from the Open Access Series on Imaging Studies (OASIS) dataset involved in this experiment. All the MRIs were interpolated to a $0.8 \times 0.8 \times 0.8$ mm digital grid, and then pre-processed using N4-bias correction [94], rigid registration to MNI 152 atlas space [121], and skull-stripping [122].

VParNet was trained to parcellate the ventricular system of the human brain into its four cavities: the left and right lateral ventricles (LLV and RLV), and the third and the fourth ventricles. It was trained on 25 NPH subjects and 15 healthy controls (not involved in the evaluations). In the original experiment of Shao et al. [120], the remaining 70 NPH subjects were used for testing. In this experiment, we downsampled the 70 NPH subject images first so that we could study the impact of super-resolution. In order to remove the impact of pre-processing, we downsampled the 70 pre-processed test datasets instead of the raw datasets. In particular, we downsampled the data to a resolution of $0.8 \times 0.8 \times 0.8r$ mm following a 2D acquisition protocol,

where r is the through-plane to in-plane resolution ratio. The number of slices in the HR images happens to be a prime number. Since the downsampled images must have an integer number of slices, the downsample ratio r, which is also the ratio between the numbers of slices in HR images and downsampled images, must be a non-integer. In the experiment, we chose r to be 1.50625, 2.41, 3.765625, 4.82, and 6.025. The downsampled images have voxel length ($0.8r$ mm) in the z-axis of 1.205 mm, 1.928 mm, 3.0125 mm, 3.856 mm, and 4.82 mm. To apply VParNet, which was trained on $0.8 \times 0.8 \times 0.8$ mm images, to these downsampled images, we used cubic b-spline interpolation, JogSSR, and SMORE(2D) to produce images on a $0.8 \times 0.8 \times 0.8$ mm digital grid. These images were then used in the same trained VParNet to yield ventricular parcellation results.

The HR NPH images have physical resolution in the z direction of 0.9 mm. We used them as ground truth and evaluated the accuracy of super-resolved images using the SSIM [111] and the PSNR within brain masks. As for the ventricle parcellation performance, we evaluated the automated parcellation results using manual delineations. We computed Dice coefficients [123] to evaluate the parcellation accuracy of the same network on different super-resolved and interpolated images. By comparing the parcellation accuracy, we can evaluate how much improvement we get from SMORE(2D) compared with interpolation.

Example images from an NPH subject, all reconstructed on a $0.8 \times 0.8 \times 0.8$ mm digital grid, are shown in Fig. 5.6. The LR image depicted using cubic-bspline interpolation has resolution $0.8 \times 0.8 \times 3.856$ mm LR while

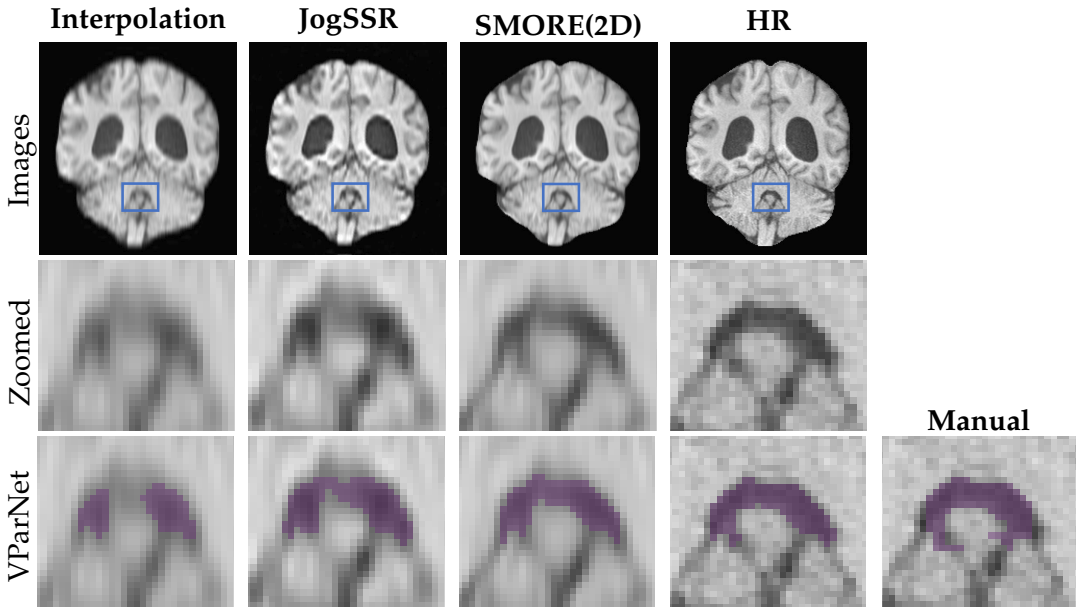


Figure 5.6: Coronal views of brain ventricle parcellation on an NPH subject: The volumes with digital resolution of $0.8 \times 0.8 \times 0.8$ mm that resolved from $0.8 \times 0.8 \times 3.856$ mm LR image using cubic-bspline interpolation, JogSSR, SMORE(2D), and the interpolated $0.8 \times 0.8 \times 0.9$ mm HR image. The patches in blue boxes are zoomed in the second row to show details of the 4th ventricle. The last row shows the VParNet [120] parcellation results and the manual labeling for the 4th ventricle.

the ground truth image has resolution $0.859 \times 0.859 \times 0.9$ mm. The JogSSR and SMORE(2D) results are also shown. To reveal more detail, the second row shows zoomed images of the 4th ventricle, where the zoomed region is shown using blue boxes in the first row. The VParNet [120] parcellations as well as the manually delineated label of the 4th ventricle are shown using purple voxels on the third row. Visually, of all the results derived from the LR data, SMORE(2D) gives the best super-resolution and parcellation results. In particular, the VParNet parcellation on the SMORE(2D) result is very close to the VParNet on the HR image.

We also evaluated these results quantitatively. The mean values of SSIM

and PSNR are shown in Table 5.2 [3]. We can see that the SR results from SMORE(2D) always have better mean SSIM and PSNR than interpolation and JogSSR. The Dice coefficient of the parcellation results of the four cavities (RLV, LLV, 3rd, 4th) and the whole ventricular system are also shown in Table 5.2. From the table, we can find that for example, VParNet on SMORE(2D) results of thickness 4.82 mm is better than interpolation results of 3.856 mm, while the later needs 56.25% longer scanning time. It shows the potential of reducing scanning time by using SMORE. It also shows that acquiring HR images with adequate SNR gives better parcellation results than LR images, even with SMORE(2D) applied to improve spatial resolution. However, if the acquired data are already limited to be anisotropic LR, which is common in clinical and research, SMORE(2D) can give better parcellation than interpolation.

It also shows that acquiring HR images with adequate SNR gives better parcellation results than LR images, even with SMORE(2D) applied to improve spatial resolution. However, if the acquired data are already limited to be anisotropic LR, which is common in clinical and research, SMORE(2D) can give better parcellation than interpolation.

5.6 Discussion and Conclusions

In this chapter, we provided results of the self-supervised super-resolution (SSR) algorithm, SMORE, applied to four different MRI datasets, and showed the

[3] Although a Wilcoxon signed-rank test was performed and reported in [45], we later realized that the this statistic should not be used since we cannot prove the difference of paired samples are distributed symmetrically, which is the assumption of Wilcoxon signed-rank test. The correct test is a sign test, but we cannot carry this out as the original data have been lost as of this writing.

Table 5.2: Evaluation of brain ventricle parcellation on 70 NPH subjects.

Metrics	Thickness	Interp.	JogSSR	SMORE	HR(0.9 mm)
	1.205 mm	0.9494	0.9507	**0.9726**	1
	1.928 mm	0.9013	0.9106	**0.9389**	
SSIM	**3.0125 mm**	0.8290	0.8400	**0.8893**	
	3.856 mm	0.7677	0.7812	**0.8387**	
	4.82 mm	0.7003	0.7170	**0.7817**	
	1.205 mm	35.0407	34.0472	**39.5053**	
	1.928 mm	31.9321	30.6444	**35.7429**	
PSNR	**3.0125 mm**	29.2785	27.4384	**31.9878**	–
	3.856 mm	27.7562	25.7118	**29.6050**	
	4.82 mm	26.4127	24.2377	**28.1593**	
	1.205 mm	0.9704	0.9705	**0.9712**	0.9715
	1.928 mm	0.9678	0.9690	**0.9706**	
Dice(RLV)	**3.0125 mm**	0.9610	0.9635	**0.9693**	
	3.856 mm	0.9527	0.9578	**0.9648**	
	4.82 mm	0.9405	0.9498	**0.9629**	
	1.205 mm	0.9710	0.9709	**0.9715**	0.9717
	1.928 mm	0.9690	0.9693	**0.9710**	
Dice(LLV)	**3.0125 mm**	0.9638	0.9641	**0.9699**	
	3.856 mm	0.9571	0.9585	**0.9663**	
	4.82 mm	0.9469	0.9510	**0.9638**	
	1.205 mm	0.9149	0.9149	**0.9163**	0.9174
	1.928 mm	0.9095	0.9097	**0.9141**	
Dice(3rd)	**3.0125 mm**	0.8945	0.8940	**0.9073**	
	3.856 mm	0.8779	0.8761	**0.8937**	
	4.82 mm	0.8560	0.8545	**0.8832**	
	1.205 mm	0.8954	0.8941	**0.8973**	0.8983
	1.928 mm	0.8891	0.8851	**0.8947**	
Dice(4th)	**3.0125 mm**	0.8741	0.8657	**0.8878**	
	3.856 mm	0.8550	0.8463	**0.8753**	
	4.82 mm	0.8254	0.8216	**0.8629**	
	1.205 mm	0.9690	0.9690	**0.9696**	0.9699
	1.928 mm	0.9665	0.9672	**0.9690**	
Dice(whole)	**3.0125 mm**	0.9602	0.9614	**0.9675**	
	3.856 mm	0.9524	0.9552	**0.9632**	
	4.82 mm	0.9408	0.9470	**0.9607**	

improved MRI resolution both visually and quantitatively. The methodology of SMORE was introduced in the previous chapter. In this chapter, we made important contributions about the utility of SMORE, and show that SMORE can be reliably and widely used in practice. First, this chapter showed

the application of SMORE on MR images produced from different pulse sequences, contrasts, and in different organs. To the best of our knowledge, no other published deep-learning SR method has demonstrated improvement on such diverse MRI data sets without training data. Second, we have demonstrated how SMORE can improve segmentation accuracy and showed there are quantifiable improvements from using SMORE in addition to the visual improvements in image quality.

In this chapter, we demonstrated the application of SMORE in real scenarios for MR images. First, we considered an important distinction between general SR on natural images and SR on real acquired MR images. Although the general SR problem has been discussed extensively in computer vision, the common SR problem setting requires well-established LR/HR paired external training data. In contrast to natural images, such external training data is much more difficult to obtain for MR images. SMORE is an SSR algorithm which requires no external training data; in other words, what it needs is only the input subject image itself. This makes SMORE more applicable in a real scenario. Second, SMORE works for a common type of MRI acquisitions that have high in-plane resolution but low through-plane resolution (thick slices). This type of MRI is widely acquired in clinical and research applications. The four experiments in this chapter were performed on four different MRI datasets with three out of them being real acquired LR datasets. From a visual comparison, we find that SMORE enhances edges but does not create structures out of nothing; this reduces the risk of wrongly altering anatomical structures. Finally, the experiment in Sec. 5.5 showed that SMORE is not only

visually appealing, but it also gives quantitative improvements in SSIM and PSNR. More importantly, applying SMORE as a preprocessing step improves ventricular segmentation accuracy on this brain MRI dataset. Furthermore, we note that sometimes lower resolution images processed with SMORE yield better segmentation results than those from higher resolution images processed with interpolation. This suggests that SMORE post-processing may allow shorter scan times.

There are several limitations when using SMORE. First, the algorithm is based on the assumption that the in-plane slices are high resolution images. This assumption neglects the fact that these in-plane slices are thick. Bad through-plane resolution will make the in-plane blurry. Second, the CNN used in SMORE is 2D, which cannot guarantee slice consistency. These two issues are addressed in Chapter 6.

There are other limitations of SMORE that are not addressed in this thesis. First, SMORE is not robust to motion artifacts. Such artifacts in MRI tend to appear as high-frequency arcs, which can actually be accentuated by SMORE. Preprocessing to reduce these artifacts could offer one approach to permit use of SMORE in these cases. Second, both SMORE(2D) and SMORE(3D) require knowledge of $h(x)$, the point spread function (or slice profile), which may not be known accurately in some cases. Third, we did not apply SMORE(2D) on MR data with slice separation different from slice thickness. Thus SMORE(2D) may not be reliable on such data. In addition, the best resolution that can be achieved by SMORE is limited to the in-plane resolution. Because of this, for example, SMORE cannot be used to enhance images that have been

acquired with isotropic resolution. Also, SMORE does not consider cases where resolution differs in three orientations. Future work may address these issues.

In conclusion, SMORE produces results that are not only visually appealing, but also more accurate than interpolation. More importantly, applying it as a preprocessing step can improve segmentation accuracy. Our SMORE results were obtained without collecting any external training data. This makes SMORE a useful preprocessing step in many MRI analysis tasks.

Chapter 6

iSMORE: an iterative framework of SMORE

6.1 Introduction

In the previous chapter, we explored four applications of SMORE and showed that SMORE brings substantial improvement compared to interpolation. SMORE and other self-supervised super-resolution (SSR) methods, such as the one introduced by Weigert et al. [105], assume that the in-plane slices of the subject image are HR and can therefore be used as HR training data. However, this assumption does not quite hold up to close scrutiny. To explain, consider a thick in-plane slice. Although it has the appearance of HR, it does suffer from through-plane blurring. Edges that pass through the slice orthogonally will appear to be sharp while edges that pass through obliquely will appear to be blurry. An example can be found in Fig. 5.1. In this figure, although the axial plane is considered as HR in-plane, we can see that the axial slice of the subject image suffers from through-plane blurring, especially near the ventricles. This is because the thick in-plane slices can be considered as averaged

HR thin slices, and the averaging brings blurring. Training on thick slices is equivalent to training on averaged true HR images, which is suboptimal. Therefore, using these blurred in-plane slices as HR training data will degrade the performance of the SSR algorithm.

Another issue with the previous CNN-based SSR methods including SMORE is that they all use a 2D CNN on 3D volumes. We know that a 2D CNN cannot guarantee slice consistency. This is especially important for 2D acquisition protocols, i.e., when images are acquired in 2D and then stacked into 3D volumes. Such 2D acquisitions may not have good slice consistency at the outset, and applying a 2D CNN on them can only make the slice consistency worse. For these 2D protocols, a 3D CNN is preferred, yet this has not been reported for SSR.

The third issue is that the previous SSR methods are only applied in a single image modality with no guidance on how to modify them for other modalities. Weigert et al. [105] developed a method for confocal and light-sheet microscopy data of cells. The SSR method in Jog et al. [102] and SMORE(3D) were developed for MRI acquired from 3D protocols, while SMORE(2D) was developed for MRI acquired from 2D protocols.

This chapter describes an extension to SMORE called iSMORE. The chapter describes its four major contributions: 1) an iterative SSR framework, 2) a new 3D CNN for SSR, 3) a new loss function and noise reduction, 4) the application to two image modalities including MRI and two-photon fluorescence microscopy.

6.2 Method

6.2.1 2D iSMORE

A workflow for the iterative framework of iSMORE is shown in Fig. 6.1(a). Consider an input image $g(x, y, z)$ with anisotropic spatial resolution—i.e, three full-width-half-maxima (FWHM) of the point spread function—of $a \times a \times b$, with $a < b$, and let the HR in-plane directions be x and y and the LR through-plane direction be z. Our goal is to restore an HR image f with resolution $a \times a \times a$. Traditional SSR methods extract in-plane (xy-plane) slices with resolution $a \times a$ from input image g, which are considered by these methods to be HR data, apply a point spread function (PSF) which mimics the mechanism of LR in the through-plane direction, and simulate LR data with resolution $b \times a$ from these HR data with resolution $a \times a$. The LR/HR pairs are used as training data for super-resolution (SR) networks. The trained SR networks are then applied to LR zx-plane slices with resolution $b \times a$ to restore HR at (ideally) $a \times a$. Finally, the super-resolved zx-plane slices are stacked in y-axis into a 3D volume, which is the SSR result f_1. This SMORE SSR procedure is the first iteration in iSMORE.

For input image g, the thick in-plane slices are actually blurred, so they are not perfect training data. On the other hand, the SSR result f_1 has thinner slices. Thus, f_1 has better through-plane resolution than input image g, and serves as better training data than g. Taking in-plane slices from f_1 as HR training data, we subsequently fine-tune (in the training sense) the SR network. The fine-tuned network is then applied to input g as in the first iteration, and

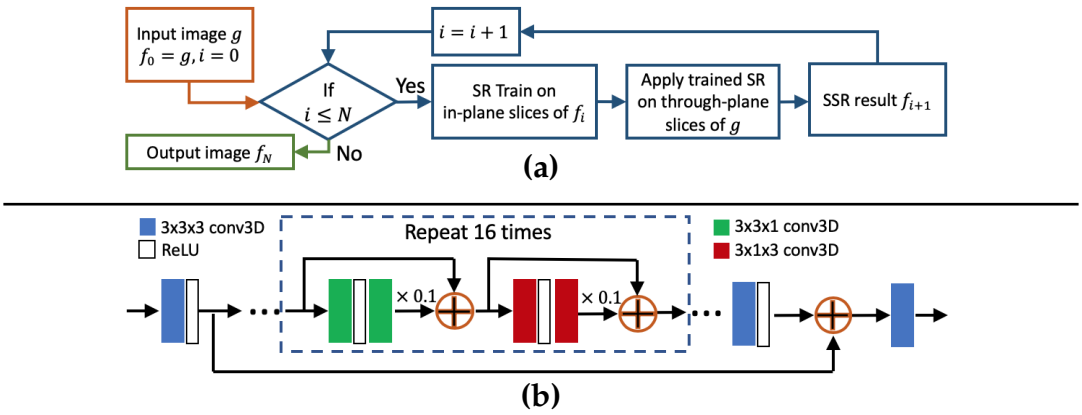


Figure 6.1: (a) The framework of iSMORE. The sequence of super-resolution networks (SR) are trained on the SSR result from the previous iteration, and SR is always applied to the input data.**(b)** Architecture of 3D EDSR.

the SSR result f_2. We iteratively perform these steps until the stop condition is met.

We use SMORE as our baseline SSR method and apply our 2D iterative framework to the SMORE result, yielding 2D iSMORE. Data augmentation for training includes flipping and rotation except for the rotation of 90° which is only used for validation to avoid overfitting.

6.2.2 3D iSMORE and a new 3D network

It is problematic to directly train a 3D network to perform SSR. If we degrade the input image g into an LR image with resolution $b \times a \times b$, and train a 3D network to learn the mapping from the degraded LR image with resolution $b \times a \times b$ to g with resolution $a \times a \times b$, then it is wrong to apply the network to the rotated input image g with resolution $b \times a \times a$ because this resolution does not match the LR training data with resolution $b \times a \times b$. This is the main reason that the previous CNN-based SSR methods all use 2D CNNs instead

of 3D CNNs.

We note that current SSR methods including SMORE give a result f_1 which has improved through-plane resolution. Now let us assume that f_1 has resolution $a \times a \times c$, with $c \approx a$. Then in the second iteration, we degrade f_1 into an image with resolution $b \times a \times c$ and train a 3D network that learns the mapping from the image with resolution $b \times a \times c$ to image with resolution $a \times a \times c$. The network is applied to a rotated input image g with resolution $b \times a \times a$. Although a and c are not exactly the same, we believe that the trained network can tolerate this inconsistency since $c \approx a$. We iteratively perform these steps until the the maximum iteration number is met.

In this 3D iterative framework, the first iteration is SMORE using 2D networks, while the subsequent iterations use 3D networks. We designed a 3D EDSR style network, with the architecture shown in Fig. 6.1(b). Since only the first dimension is LR, making all the convolutional kernels to be $3 \times 3 \times 3$ is a waste of parameters. We therefore only use $3 \times 3 \times 3$ kernels in the beginning and the end, while the repeated residual blocks contain $3 \times 3 \times 1$ and $3 \times 1 \times 3$ kernels. The number of features is 256 as in 2D EDSR. Since 3D networks are more data hungry than 2D networks, we use reflection padding instead of zero padding for convolution to make good use of small training patches.

6.2.3 Modifications for MRI and Two-photon Fluorescence Microscopy

The choice of 2D iSMORE or 3D iSMORE depends on the data. 3D iSMORE uses a 3D CNN, which better preserves slice consistency yet is very time

consuming to train and test. 2D iSMORE on the other hand, saves time and is less prone to overfitting since 2D CNNs are not as data hungry as 3D CNNs. For MRI and two-photon fluorescence microscopy, we made different modifications to iSMORE.

MRI: SMORE(3D) and SMORE(2D) are SSR methods designed for MRI acquired from 3D and 2D protocols. 3D MRI protocols acquire data in 3D Fourier space while 2D MRI protocols acquire data in 2D Fourier space (after slice selection). 3D MRI requires an inverse 3D Fourier transform for reconstruction while 2D MRI requires a set of inverse 2D Fourier transforms for 2D slices which are then stacked to form a 3D volume. For MRI data acquired from 3D and 2D protocols, we use the corresponding SMORE as our baseline SSR method, and apply our iterative framework on the SMORE result, yielding iSMORE.

For further improvement, we made another modification to SMORE. The original method uses L1 loss $\sum_{\mathbf{x}} |f(\mathbf{x}) - \hat{f}(\mathbf{x})|$ to train the CNN to perform SR, with $\mathbf{x}$ being the coordinates, $\hat{f}$ being the output of the network, and f being the ground truth images. Here, we use Sobel filters to compute edge maps in each dimension of images $\hat{f}$ and f, and define a Sobel edge loss function $|\text{Sobel} \circ f(\mathbf{x}) - \text{Sobel} \circ \hat{f}(\mathbf{x})|$, which is previously used in Bei et al. [124] for 2D natural image super-resolution. This loss can emphasize the edges, which are what we want to enhance most. The final loss function we here used is $\sum_{\mathbf{x}} |f(\mathbf{x}) - \hat{f}(\mathbf{x})| + w|\text{Sobel} \circ f(\mathbf{x}) - \text{Sobel} \circ \hat{f}(\mathbf{x})|$, with weight $w = 1$. We demonstrate the effect of Sobel loss in Sec. 6.3.1. In this chapter, we only empirically choose $w = 1$. Future work may explore its effect.

Two-photon Fluorescence Microscopy: Two-photon fluorescence microscopy data are acquired in 2D, and then stacked into 3D volumes, which is a similar strategy as MRI acquired from 2D protocols. Thus we use SMORE(2D) as the baseline method, but we make two modifications. First, the spatial resolution (defined as the FWHM of the PSF) in the z-axis of two-photon fluorescence microscopy data is affected by the optical setting and imaging parameters. Ideally, when the laser is perfectly focused, the PSF can be specified in closed form, which depends on the numerical aperture (NA) of the optical system and the wavelength of the laser used [125][1]. However, in reality, the laser is very difficult to be perfectly focused. Even if the laser is perfectly focused, it is only perfect for a certain depth z. Thus the ideal PSF is unachievable. Fortunately, we know that the orthogonal cross-section of the vessels are close to isotropic circles, and the true isotropic HR image should have the same property. Taking advantage of this fact, we can manually estimate the FWHM of the PSF from those enlongated orthogonal cross-sections in the subject image by computing the fraction between the width and height of these cross-sections. We model the PSF along the z-axis as $A\text{sinc}(\beta z/4)^4$ [125], with β computed from the estimated FWHM.

Second, two-photon fluorescence microscopy data have a much higher noise level than MRI. An example is shown in the first column of Fig. 6.3. SR networks sharpen edges but also emphasize noise. To prevent further noise amplification, we add noise to the LR training data but not to the HR data, thus forcing the network to perform resolution enhancement and denoising

[1]The 3D PSF model for a perfectly focused two-photon fluorescence microscopy is complicated. Yet on the z-axis, PSF$(0,0,z)$ is modeled as $A\text{sinc}(u/4)^4$, with optical unit $u = \frac{8\pi n}{\lambda}\sin^2(\frac{\alpha}{2})z$. n is refractive index, λ is the wavelength, α is the beam conus angle. [125]

at the same time. The noise we add contains both Poisson noise and speckle noise to mimic the noise seen in the LR subject image without noise reduction. Poisson noise is also called shot noise, which is due to the quantum nature of light. Poisson noise is considered in some literature to be the dominating noise source in fluorescence microscopy [126]. However, we found that adding Poisson noise to training data is not adequate to denoise the images. So we considered adding other type of noise source, such as speckle noise. Speckle noise [127] results from the reflection of coherent lights at rough surfaces. The level of speckle noise depends on the imaged object. We tried different levels of speckle noise, and empirically chose one for this microscopy data.

For the two-photon fluorescence microscopy data, we use a 3D network. The two-photon fluorescence microscopy we are studying contains a large number of vessels that pass through planes. For such data, results from CARE [105] and SMORE both show that a 2D network cannot guarantee slice consistency and is not able to capture enough 3D information. An example is shown in Fig. 6.3. Therefore, we use 3D iSMORE applied to the denoised version of SMORE(2D). The 3D network uses the 3D Sobel edge loss[2].

6.2.4 Comparison between SMORE and iSMORE

The comparison between SMORE and iSMORE is shown in Table 6.1, and explained below.

- The first iteration of iSMORE, i.e. iSMORE$_{i=1}$, is modified SMORE. The difference between SMORE and iSMORE$_{i=1}$ is the Sobel edge loss. For

[2]Code is available in https://github.com/volcanofly/tf_Sobel_edge_3D

noisy data like microscopy, iSMORE$_{i=1}$ has an additional denoising module.

- The difference between iSMORE$_{i=1}$ and iSMORE$_{i>1}$ is the iterative framework shown in Fig. 6.1(a).
- The difference between 2D iSMORE$_{i>1}$ and 3D iSMORE$_{i>1}$ is the re-trained network architecture.
- For 3D iSMORE, the first iteration is modified SMORE using 2D network, while the remaining iterations use a 3D network.

Table 6.1: Comparison of SMORE and iSMORE

	Subject image acquisition protocol	Network	Loss	Denoise	iterative
SMORE(2D)	acquired as 2D, stack to 3D	2D	L1	No	No
SMORE(3D)	acquired in 3D *k*-space	2D	L1	No	No
2D iSMORE$_{i=1}$	same with corresponding SMORE	2D	L1 + edge	Optional	No
2D iSMORE$_{i>1}$	same with corresponding SMORE	2D	L1 + edge	Optional	Yes
3D iSMORE$_{i=1}$	same with corresponding SMORE	2D	L1 + edge	Optional	No
3D iSMORE$_{i>1}$	same with corresponding SMORE	3D	L1 + edge	Optional	Yes

6.3 Experiments

6.3.1 2D iSMORE on MRI from 3D protocols

We compare 2D iSMORE (iSMORE using 2D network) to the original SMORE using MRI downsampled following 3D protocols. The ground truth HR images are T_2-weighted images from 14 multiple sclerosis subjects imaged on a 3T Philips Achieva scanner with acquired resolution of $1 \times 1 \times 1$ mm. The high frequency signals in the z-axis are completely zeroed out to simulate 3D protocols. An additional Fermi filter is applied to simulate an anti-ringing filter.

The blurred LR images have resolution $1 \times 1 \times r$ mm, where $r = \{2, 3, \ldots, 6\}$. They are used as input images for methods including zero filling interpolation, SMORE(3D), and 2D iSMORE.

We computed the peak signal to noise ratio (PSNR) and the structural similarity (SSIM) for the results of these methods using the ground truth HR images as references. The mean values are shown in Table. 6.2. We see that the both the Sobel edge loss and the iterative strategy of iSMORE always improves the mean SSIM and PSNR [3].

Table 6.2: Quantitative evaluations for iSMORE using 2D network on MRI from 3D protocols: PSNR and SSIM evaluation on fourteen $1 \times 1 \times r$ mm T2-w subjects down- sampled from $1 \times 1 \times 1$ mm MRI acquired with 3D protocols with different ratio r. We compare the SSIM/PSNR of results from zero filling interpolation, original SMORE, iSMORE using 2D network after iteration from i=1 and i=5.

		interp.	**SMORE**	**iSMORE$_{i=1}$**	**iSMORE$_{i=5}$**
Network		-	2D	2D	2D
Sobel edge loss		-	No	Yes	Yes
Denoise		-	No	No	No
Iterative		-	No	No	Yes
PSNR	**r = 2**	35.9028	38.1731	38.4205	**38.5536**
	r = 3	32.0577	34.2162	34.3696	**34.5477**
	r = 4	29.9575	31.9099	32.0198	**32.2273**
	r = 5	28.6226	30.4116	30.5206	**30.6989**
	r = 6	27.6569	29.1900	29.3343	**29.5421**
SSIM	**r = 2**	0.9377	0.9510	0.9534	**0.9542**
	r = 3	0.8638	0.8990	0.9021	**0.9041**
	r = 4	0.7881	0.8394	0.8432	**0.8461**
	r = 5	0.7224	0.7863	0.7903	**0.7940**
	r = 6	0.6643	0.7379	0.7380	**0.7414**

In Fig. 6.2, we show the ratios between iSMORE$_i$ and iSMORE$_{i=1}$ for the first five iterations of iSMORE. We see that the largest improvement happens

[3] Although a Wilcoxon signed-rank test was performed and reported in [46], we later realized that the this statistic should not be used since we cannot prove the difference of paired samples are distributed symmetrically, which is the assumption of Wilcoxon signed-rank test. The correct test is a sign test, but we cannot carry this out as the original data have been lost as of this writing.

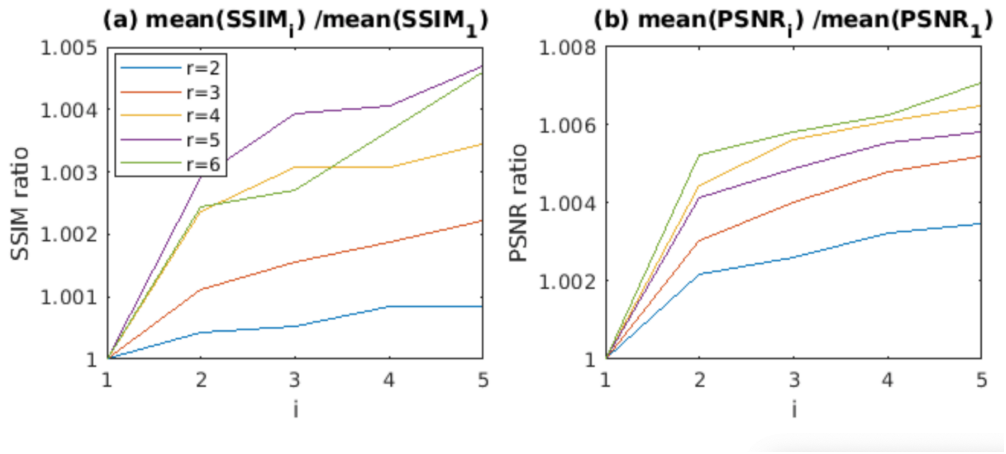


Figure 6.2: The ratio of PSNR/SSIM between iSMORE with $i = 1, 2, 3, 4, 5$ and $i = 1$. The experiment setting is same as Table 6.2.

between $i = 1$ and $i = 2$. If computational time is of concern, then $i = 2$ is a good choice. Another finding is that generally the improvement from the iterative framework is larger when the LR factor r is larger. This finding is not strict, yet holds true in general. A third observation is that improvement in PSNR is larger than that of SSIM. This might come from the fact that the first step in computing the SSIM is to apply a Gaussian filter, which degrades the details.

6.3.2 3D iSMORE on Two-photon Fluorescence Microscopy

We used serial two-photon tomography (STPT) to image brain blood vessel images at cellular resolution in mice. The data was acquired by Dr. Seoyoung Son and Dr. Yongsoo Kim from Penn State University. To label blood vessels, a mouse was transcardially perfused with 0.9% saline followed by 4% paraformaldehyde and a Fluorescein isothiocyanate (FITC)-albumin conjugated gel. Detailed information about STPT imaging was described in Ragan

et al. [128]. Briefly, the brain was embedded in 4% oxidized agarose and the embedded brain was placed on the motorized stage in tissuecyte 1000 (Tissuevision). The brain was imaged at 1 μm (xy-plane) resolution with 5 μm z-axis increment for 200 μm thickness.

In Fig. 6.3, we show the original LR image, and the results of cubic b-spline interpolation (BSP), Content-AwaRE image restoration (CARE) [105] and SMORE(2D) with estimated z-axis FWHM of 15μm, the denoised version of SMORE(2D), and the proposed 3D iSMORE after the third iteration. CARE [105] is an SSR tool designed for fluorescence microscopy with a denoise option, and has publicly available code. Compared with the original LR image, the BSP result is less noisy and blurry. The CARE result is sharp and relatively clean, yet many cross-sections of vessels in that result are not ellipses, which implies that CARE contains sharp artifacts. The SMORE result is much sharper than BSP, but is very noisy. The denoised version of SMORE assumes Poisson noise and 30% speckle noise as described in Sec. 6.2.3, yielding a result with much less noise, which forms our iSMORE$_{i=1}$. The result of the proposed 3D iSMORE$_{i=3}$ has vessels with more isotropic cross-sections, and contains the fewest artifacts in this comparison.

It is very difficult to obtain isotropic HR ground truth for STPT data since acquiring data with an isotropic PSF is generally not possible. Thus, metrics like SSIM and PSNR are not available. In order to show iSMORE's overall performance in the 3D volume, we performed maximum intensity projection (MIP) on three planes, as shown in Fig 6.4. Visually, the MIP of the proposed method iSMORE looks the most isotropic and clear. CARE also

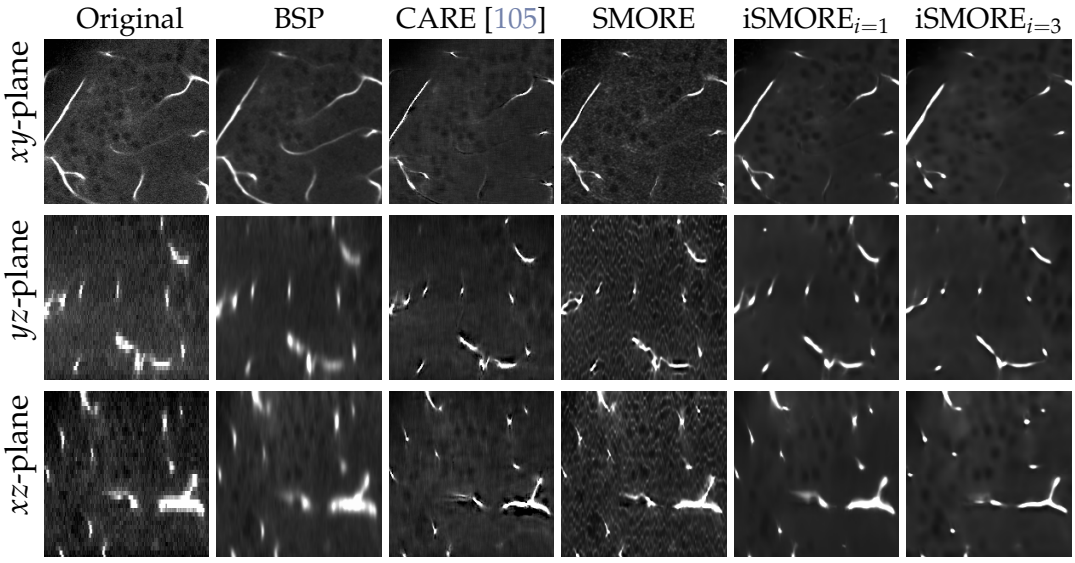


Figure 6.3: Views from three orthogonal planes of the original LR image, the cubic B-spline (BSP) interpolated image, result of CARE [105], SMORE(2D), our denoised version of SMORE(2D) which is also the first iteration of iSMORE, and our proposed 3D iSMORE with $i = 3$.

provides a good MIP, yet the artifacts shown in Fig. 6.3 cannot be ignored.

6.4 Conclusion and Discussion

In this chapter, we described 2D and 3D iSMORE, an iterative framework built upon the SMORE method. The comparison between the methodologies of iSMORE and SMORE is shown in Table. 6.1. The idea behind iSMORE is that thick in-plane slices are not as good as thin slices in training. Using this idea, iSMORE improves the performance of SMORE. And more importantly, it enables a 3D network, which solves the slice consistency issue raised by 2D networks used by previous SSR methods.

There are some limitations of iSMORE that we would like to discuss. First,

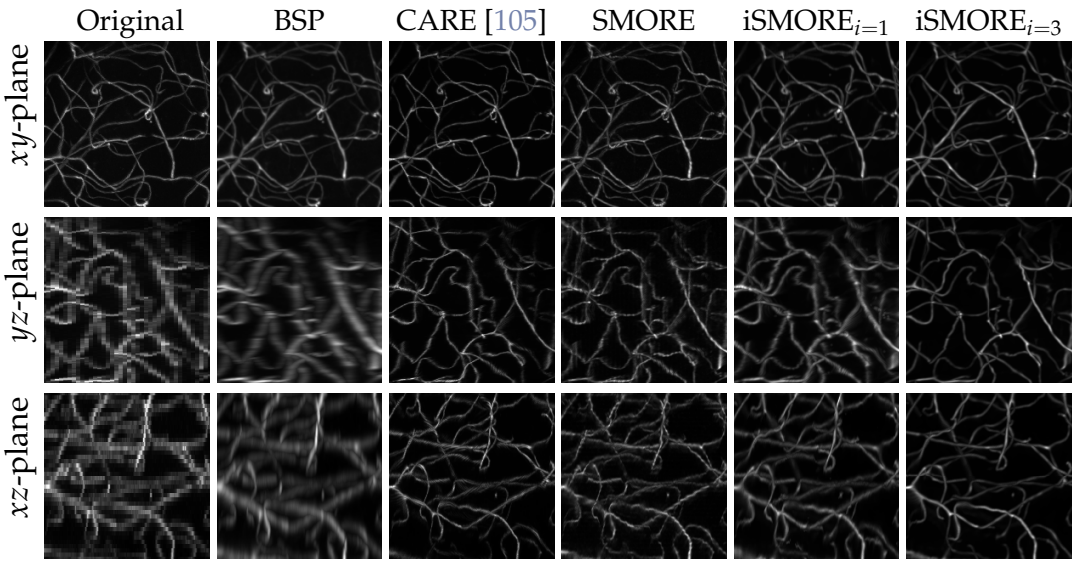


Figure 6.4: Maximum intensity projection (MIP) on three orthogonal planes of the original LR image, the cubic B-spline (BSP) interpolated image, result of CARE [105], SMORE(2D), our denoise version of SMORE(2D) which is also the first iteration of iSMORE, and our proposed 3D iSMORE with $i = 3$.

it might be confusing that we use 2D iSMORE for images with 3D protocols, while using 3D iSMORE for images with 2D protocols. To clarify, 2D/3D protocols describe the MRI acquisition method used and are not the same as 2D/3D iSMORE, where 2D/3D describes the 2D/3D CNN used. Although 3D iSMORE preserves slice consistency, 2D iSMORE uses a 2D CNN, is easier to train, and saves time. One more iteration takes about 20 mins for 2D iSMORE, and more than 1 hour for 3D iSMORE on microscopy data. Images acquired with 3D protocols already have good slice consistency, so 2D iSMORE is able to handle them. For images acquired with 2D protocols, slice consistency is more of a concern. Thus, 3D iSMORE is a better choice in this case, however it requires a larger computation time. Second, the number of iterations of iSMORE that we used was manually set. To clarify, due to computation time,

we do not recommend a large number of iterations. From Fig. 6.2, we found that mean SSIM/PSNR increase monotonically as iteration i increases from 1 to 5. However, we only recommend using of $i = 2$ for this dataset since time increases linearly with i. For microscopy data, we use $i = 3$ since the improvement between the 2nd and 3rd iteration is still large. Future work will include the development of a method for choosing i. Finally, one might be concerned that a large number of iterations might bring overfitting or artifacts. However, from our experiment whose results are shown in Fig. 6.2, both SSIM/PSNR increase with iteration count. Moreover, from Fig. 6.3, we see that iSMORE with $i = 3$ has better cross-sectional shapes and fewer artifacts than iSMORE with $i = 1$ and CARE.

In summary, in this chapter we described a new algorithm called iSMORE. We evaluated this algorithm both quantitatively and qualitatively, and applied it on both downsampled and real acquired low resolution medical images with two very different modalities. We applied iSMORE to downsampled MR images with ground truth HR images to evaluate its accuracy with SSIM and PSNR. The results from Table 6.2 show that both Sobel edge loss and the iterative framework can improve the accuracy in MRI. Furthermore, we adjusted iSMORE to be applied in real two-photon fluorescence microscopy data which have a higher noise level. The result and its maximum intensity projection on three orthogonal planes are visually more isotropic, the vessels are visually clearer and easier to track than the original SMORE. Future work will include a deeper exploration on the parameters used in this algorithm as well as a comparison on different network architectures.

Chapter 7

Discussion, Conclusions, and Future Work

7.1 Summary

In this thesis, we described one algorithm for image modality synthesis and two algorithms for image resolution enhancement. In Chapter 3, we discussed our CNN-based CT-to-MR image synthesis algorithm. Then we described our SSR algorithm SMORE for MRI acquired with 3D and 2D protocols in Chapter 4, and its applications on various MR images in Chapter 5. Finally, in Chapter 6, we described an extension of SMORE — the iterative framework iSMORE — as well as its application on two-photon fluorescence microscopy images. In this chapter, we summarize what we have learned and discuss possible improvements in the future.

7.2 Image Modality Synthesis

7.2.1 Key Points and Results

- The CT-to-MR synthesis we developed is based on the CNN machine learning methodology. As the first CNN-based CT-to-MR synthesis algorithm published in 2017, it refutes the pessimistic assertion about CNN-based CT-to-MR synthesis [23], and shows that it is not only possible but it can be done with sufficient quality to open up new clinical and scientific opportunities in neuroimaging.

- This is the first work to provide grey matter anatomical labels on a CT neuroimage. Through image modality synthesis, this algorithm synthesized MR from CT, converts CT-MR multi-modal registration problem into MR-MR mono-modal registration, and thus significantly improves multi-atlas segmentation, which is based on registration.

7.2.2 Future Work

- As an early exploration, this work used a modified 2D U-net as the neural network architecture. More recent work like attention network, adversial loss, perceptual loss, semi-supervised learning, self-supervised learning, and meta-learning can be explored to improve the synthesis results.

- MRI data have various contrasts. Developing algorithms that can handle such MRI data is a very practical problem, and can be considered in future work.

- The possible misregistration between paired CT/MR training data may require newly designed networks that are robust to such imperfect training data.
- This work is based on a 2D network, which has the issue of slice consistency. A 3D network can keep slice consistency yet requires a large amount of training data. Future work includes developing a few-shot CNN algorithm that does not have slice consistency issue and does not need a large amount of training data.
- This work demonstrates how image synthesis improves multi-atlas segmentation. Future work may explore its applications in other tasks, such as direct CNN-based segmentation, CNN-based registration, computer aided diagnosis, etc.

7.3 Image Resolution Enhancement Method SMORE and iSMORE

7.3.1 Key Points and Results

- SMORE and iSMORE improve through-plane resolution of subject images without using any high-resolution images as training data. The main difference is that iSMORE is an iterative extension of SMORE. Compared with SMORE, iSMORE gives a small but significant improvement on MRI as measured by SSIM/PSNR, and visually much better performance on a real two-photon fluorescence microscopy data.
- SMORE is not an algorithm that simply replaces a traditional regressor

with a deep network. It is the first self-supervised super-resolution method that takes MRI 2D and 3D acquisition models into consideration. Its good performance is based on the understanding of the underlying models of MRI acquisition.

- Several advantages of SMORE make it easy to be applied. The biggest advantage of SMORE is the fact that SMORE does not need external training data. Second, SMORE needs no preprocessing step other than N4 inhomogeneity correction [94]. Third, extensive parameter tuning is not required for SMORE. All these properties are desirable for easy application to new MRI datasets.

- Because of the advantages described above, we easily demonstrated the four applications of SMORE in real world scenarios for MR images in Chapter 5, including a quantitative experiment on segmentation. To the best of our knowledge, no other published deep-learning SR method has demonstrated improvement on such diverse MRI data sets without training data.

7.3.2 Future Work

- iSMORE for two-photon fluorescence microscopy is not as mature as SMORE on MRI. It has only been tested on a small amount of data, and requires further efforts to make it robust on various datasets.

- iSMORE requires manual specification of the blur model and noise level of two-photon fluorescence microscopy data. Future work should make

this step fully automatic.

7.4 Concluding Thoughts

Deep learning has dominated medical image research in the past few years. It outperforms previous methods in many tasks. However, the performance of a deep network largely relies on training data. This drawback cannot be neglected especially in medical imaging since high quality data is more difficult to obtain than in natural images. An ideal set of training data should have high resolution and similar contrast as the testing data. This is a requirement not only in the case of deep networks, but is also desired for visualization and registration.

In this thesis, we have presented one algorithm for image modality synthesis and two algorithms for image resolution enhancement. The goal of this work was to develop image synthesis and resolution enhancement algorithms as effective preprocessing tools both for visual quality and for automatic medical image analysis methods such as registration or segmentation. We have shown that our image synthesis algorithm does improve registration-based segmentation, and our resolution enhancement algorithms do provide visual enhancement and improve CNN-based segmentation.

Our hope is that researchers find these tools worth using as preprocessing steps. Particularly, we would like to make SMORE more user-friendly and available to the community for research purposes. In the long term, SMORE has the potential to be included in the MRI scanner firmware due to its easy

application when the image acquisition parameters are known. As a CNN-based method, it gives excellent results yet does not require the availability of high-resolution training data and thus does not involve privacy issues with patient information. A patent on SMORE has been filed. We hope the usage of SMORE can benefit researchers, doctors, and patients in the future.

Appendix A

A supervoxel-based random forest framework for bidirectional MR/CT synthesis

Abstract

Synthesizing magnetic resonance (MR) and computed tomography (CT) images (from each other) has important implications for clinical neuroimaging. The MR to CT direction is critical for MRI-based radiotherapy planning and dose computation, whereas the CT to MR direction can provide an economic alternative to real MRI for image processing tasks. Additionally, synthesis in both directions can enhance MR/CT multi-modal image registration. Existing approaches have focused on synthesizing CT from MR. In this appendix, we propose a multi-atlas based hybrid method to synthesize T1-weighted MR images from CT and CT images from T1-weighted MR images using a common framework. The task is carried out by: (a) computing a label field based on supervoxels for the subject image using joint label fusion; (b) correcting this result using a random forest classifier (RF-C); (c) spatial smoothing using a

Markov random field; (d) synthesizing intensities using a set of RF regressors, one trained for each label. The algorithm was evaluated using a set of six registered CT and MR image pairs of the whole head.

A.1 Introduction

Synthesizing computed tomography (CT) images from magnetic resonance (MR) images has proven useful in positron emission tomography (PET)-MR image reconstruction [129, 1] and in radiation therapy planning [130]. To overcome the lack of a strong MR signal in bone, one method [129] used specialized MR pulse sequences and another method [1] used multi-atlas registration with paired CT-MR atlas images. The synthesis of MR images from CT images is a new challenge that has not been reported until very recently [23, 131]. Potential uses for this process include 1) intraoperative imaging where visualization of soft tissue from cone-beam CT could be enhanced by generation of a synthetic MR image and 2) in multi-modal registration where use of both modalities can improve the accuracy of registration [132, 133]. The difficulty in CT-to-MR synthesis is the lack of a strong soft-tissue contrast in the source CT images. Given the duality that appears between these tasks, we have discovered a core organizing principle for bi-directional image synthesis and developed a new image synthesis approach.

To synthesize CT images from MR images, Burgos et al. [1] used multiple CT/MR atlas pairs, wherein the atlas MR images are deformably registered to the target MR image. The transformations are then applied to the atlas CT images and fused to form a single CT intensity. Although this approach

can also be used to synthesize MR from CT, some degree of blurring can be expected due to the inaccuracies in registration due to poor soft-tissue contrast in the CT images. Machine-learning approaches that have been developed for image synthesis (cf. [9, 24]) can also be used for synthesizing MR from CT; but image patches by themselves do not contain sufficient information to distinguish tissue types without additional information about the location of the patches.

Image segmentation has long been used for image synthesis [16]. If the tissue type and physical properties are known, then given the forward model of the imaging modality, the corresponding tissue intensity can be estimated. However, in our framework, segmented regions are used to provide *context* wherein synthesis can be carried out through a set of learned regressions that relate the intensities of the input modality to those of the target modality. We demonstrate synthesis in both directions, MR to CT and CT to MR, using our method.

A.2 Methods

Given a subject image of modality 1 (M1), denoted by I^{M1}, our goal is to synthesize an image of modality 2 (M2), denoted by $\hat{I}^{M2}$. To achieve this goal, we have a multi-atlas set, $\mathcal{A} = \{(A_n^{M1}, A_n^{M2}) | \, n = 1, ..., N\}$, which contains N pairs of co-registered images of M1 and M2. An example of an atlas pair, where M1 is CT and M2 is MR (T1-weighted) is depicted in Fig. A.1(a). The two intensities in atlas image pairs are examples of possible synthetic values, when synthesizing in either direction. It is well known that this relationship is

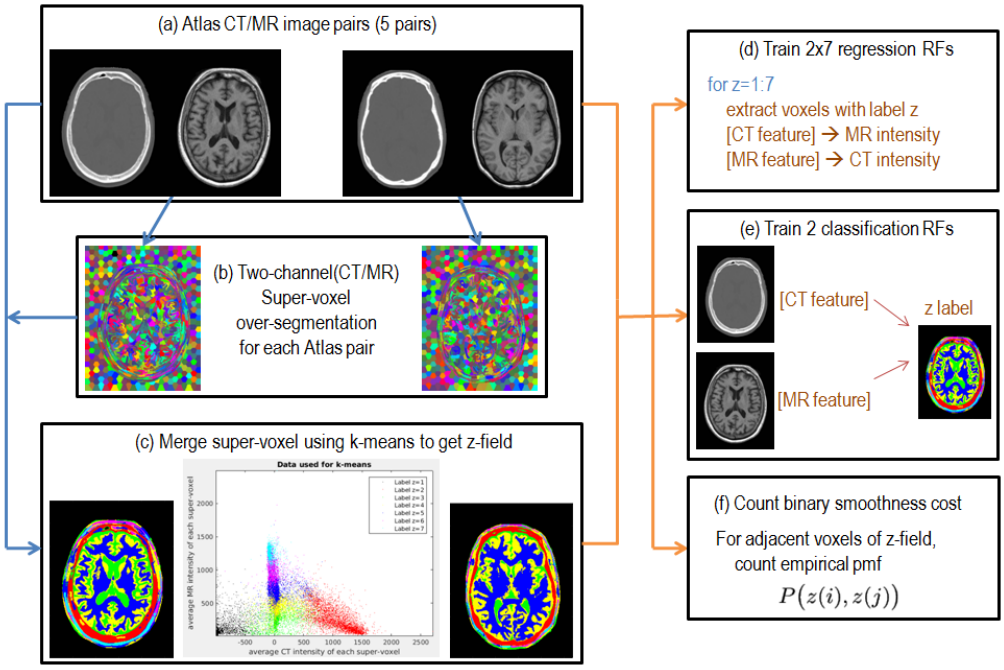


Figure A.1: (a) Two CT/MR atlas pairs; **(b)** result of SLIC over-segmentation; **(c)** k-means clustering of supervoxels yields a z-field image; **(d)** training of $2 \times K$ RF regressors; **(e)** RF-Cs trained to estimate z-fields from single modalities; and **(f)** computation of pairwise potentials for a MRF.

not a bijection; given an intensity in M1 there may be multiple corresponding intensities in M2. However, given a particular tissue (e.g., white matter) the relationship is less ambiguous. We carry out a segmentation on the atlas images that divides them into distinct regions characterized by different paired intensities. Paired intensities from these regions are then used to train separate regressors that predict one modality from the other given the tissue class.

We start with the atlas image set $\mathcal{A}$. Each pair of atlas images is processed using the following steps with the eventual goal of learning regressions that predict the target modality given the input modality. The first step is a supervoxel over-segmentation process using a 3D version of the *simple linear*

iterative clustering (SLIC) method [134] wherein the intensity feature space comprises the M1 and M2 intensity pairs. A result of SLIC on two atlas pairs is shown in Fig. A.1(b). Multichannel k-means and fuzzy k-means have been previously used for tissue classification in neuroimaging [135]. However, it is difficult to obtain spatially contiguous regions using these simple methods. Super-voxel over-segmentation provides us with spatially contiguous regions that have homogeneous intensities.

We combine these homogeneous intensity regions by clustering them on the basis of their average supervoxel intensities taken jointly from both M1 and M2. These are clustered using the k-means clustering algorithm, which yields supervoxels that are labeled as $z = 1, 2, \ldots, K$. The voxels forming each supervoxel inherit the cluster label of the supervoxel and therefore yield an image of labels, which we call the z-field. Two examples of z-fields are shown in Fig. A.1(c), where each label in the z-field is shown using a different color. A random selection of intensity pairs are plotted in the center of Fig. A.1(c) (CT/MR on the horizontal/vertical axis), and colored by the z-field. These intensity pairs and their voxel-wise features, along with their labels provide the training data for regressors that predict the intensity of the target modality given the features of the input modality. Our features consist of $3 \times 3 \times 3$ image patches together with average image values in patches forming a constellation around the given voxel ("context features" similar to those in [136, 10]). We need $2 \times K$ regressors, one each per modality and cluster. For each label z, we extract features from M1 images and pair them with corresponding M2 intensities. This acts as the training data set for a random forest (RF) regressor.

The training step is depicted in Figure A.1(d).

Given the subject image I^{M1} *and* the corresponding z-field that labels its voxels, we can apply the corresponding regressor based on the z value at that voxel to predict the synthetic M2 intensity in image $\hat{I}^{M2}$. Thus, we next describe how to estimate the z-field for the subject image. The z-field of I^{M1} is estimated by fusing two approaches. First, we predict an estimate of the z-field directly from the same image features that were noted above using a *random forest classifier* (RF-C). Shown in Fig. A.1(e), are two random forests designed to synthesize K labels from either M1 or M2, which are trained in analogous fashion to the RF regressors described above. A second estimate of the z-field is generated using a multi-atlas segmentation. In this case, we augment the atlases to include the z-fields found using the supervoxel clustering approach (essentially augmenting the image pairs in Fig. A.1(a) with the label fields in Fig. A.1(c)), deformably register every atlas pair to I^{M1}, apply the learned transformations to the corresponding z-fields, and combine the labels using *joint label fusion* (JLF) [84]. The registration between I^{M1} and the atlas pair uses a two-channel approach in which the first channel uses the cross-correlation metric between I^{M1} and A^{M1} and the second channel uses the mutual information metric between I^{M1} and A^{M2}.

We now have two estimates of the z-field for I^{M1}, $\hat{z}_{\text{RF-C}}$ and $\hat{z}_{\text{JLF}}$, each providing a probability for each label at each voxel, $P_{\text{RF-C}}(z)$ and $P_{\text{JLF}}(z)$. Our experiments reveal that the RF-C yields inferior results in regions where intensities of the labels are ambiguous, while the JLF yields inferior results in areas where the registration is not accurate. We choose the label that maximizes the

product of their probabilities at each voxel with a MRF spatial regularization.

Using a conventional MRF framework, we define the estimated z-field,

$$\hat{\mathbf{z}} = \arg\min_{z(i)} \sum_i E_{\text{unary}}\big(z(i)\big) + \sum_{i,j} E_{\text{pair}}\big(z(i), z(j)\big)\,, \tag{A.1}$$

where $E_{\text{unary}}\big(z(i)\big)$ is the unary potential for voxel i and $E_{\text{pair}}\big(z(i), z(j)\big)$ is the pairwise potential for adjacent (6-connected) voxels i and j. Since this energy will be used in a Gibbs distribution, the unary potential is defined as follows

$$E_{\text{unary}}\big(z(i)\big) = -\log P_{\text{RF}-\text{C}}\big(z(i)\big) - \log P_{\text{JLF}}\big(z(i)\big) \tag{A.2}$$

which yields the desired product of probabilities as the driving objective function for assigning labels to voxels.

Although the Potts model is often used in multi-label MRF models [137]—this is the model in which different labels have unity cost and similar labels have zero cost—we can exploit our atlas and its subsequent analysis to yield a cost function that is highly tailored to our application. Consider the z-fields produced by over-segmentation followed by k-means, as shown in Fig. A.1(c), and consider adjacent voxels i and j. From the full collection of these images, we can compute the empirical joint probability mass function $P(z(i), z(j))$ for all adjacent voxels, as illustrated in Fig. A.1(f). Some labels will almost never appear adjacent to each other and thus should be penalized heavily in the MRF we design. Accordingly, we define the pairwise potential as

$$E_{\text{pair}}\left(z(i), z(j)\right) = -\log P\big(z(i), z(j)\big) + \frac{1}{2}\big(\log P\left(z(i), z(i)\right) + \log P\left(z(j), z(j)\right)\big). \tag{A.3}$$

When the labels are the same the cost is zero and when they are different, the cost increases according to their rarity of occurrence in the atlas. Given these definitions of unary and pairwise potentials (which is a semimetric), the estimated z-field is found by solving (A.1) using the α-β swap graph cut approach [138].

A.3 Experiments

MR images were obtained by Dr. Junghoon Lee from Johns Hopkins School of Medicine using a Siemens Magnetom Espree 1.5 T scanner (Siemens Medical Solutions, Malvern, PA) and CT images were obtained using Philips Brilliance Big Bore scanner (Philips Medical Systems, Netherlands) under the routine clinical protocol from brain cancer patients treated by stereotactic-body radiation therapy (SBRT) or radiosurgery (SRS). Geometric distortions in MR images were then corrected using a 3D correction algorithm available in the Siemens Syngo console workstation. All MR images were then N4 corrected and normalized by aligning white matter peak identified by fuzzy C-means.

We applied our method to six subjects each associated with true CT and MR images to which to compare our results. For algorithm comparison, we implemented Burgos et al. [1] with a different local similarity measure, referred as Burgos+. Burgos et al. [1] employs an intensity fusion method that uses local normalized cross correlation (LNCC) as the local similarity measure. Burgos+ uses SSIM instead as suggested in Lee et al. [139]. Existing work on CT/MR synthesis [1] has focused on synthesizing CT from MR, so we can directly compare. Without a published method for synthesizing MR

from CT, we simply applied Burgos+ in the reverse direction. To evaluate efficacy of synthesis, we computed SSIM and PSNR on the synthetic images with respect to the true images. The result is shown in Figure A.2. In addition to the comparison with Burgos+, we have shown how well modifications of our own algorithm perform. The "JLF" result uses only the z-field computed from JLF, the "RF-C" result uses only the z-field computed from RF-C, the "JLF+RF-C" result uses the product of the two z-field probabilities without MRF; and the "MRF" result is our proposed algorithm. We can see our method produces, in terms of SSIM and PSNR, better synthetic MR in every respect, while the synthetic CT images are better than Burgos+ in terms of SSIM and comparable in terms of PSNR.

Figure A.3 shows the estimated z-fields and final synthetic CT images for two subjects. It shows that our synthetic CT images have higher contrast and no blurry edges as compared to Burgos+, yet look somewhat artificial compared to the truth. Figure A.4 shows the estimated z-fields and final synthetic MR images for the same two subjects. It shows that our synthetic MR images also have high contrast and no blurry edges as compared to Burgos+. We notice in Fig. A.4(e), the result from Burgos+ cannot synthesize the soft tissues correctly. This is because the result depends on the accuracy of registration between atlas image pairs and subject CT images, which is relatively low in areas of soft tissues. Our method is more robust to registration inaccuracies because we use a MRF to predict the z-field and the K random forests used in synthesis overlap in their intensity coverage to some extent.

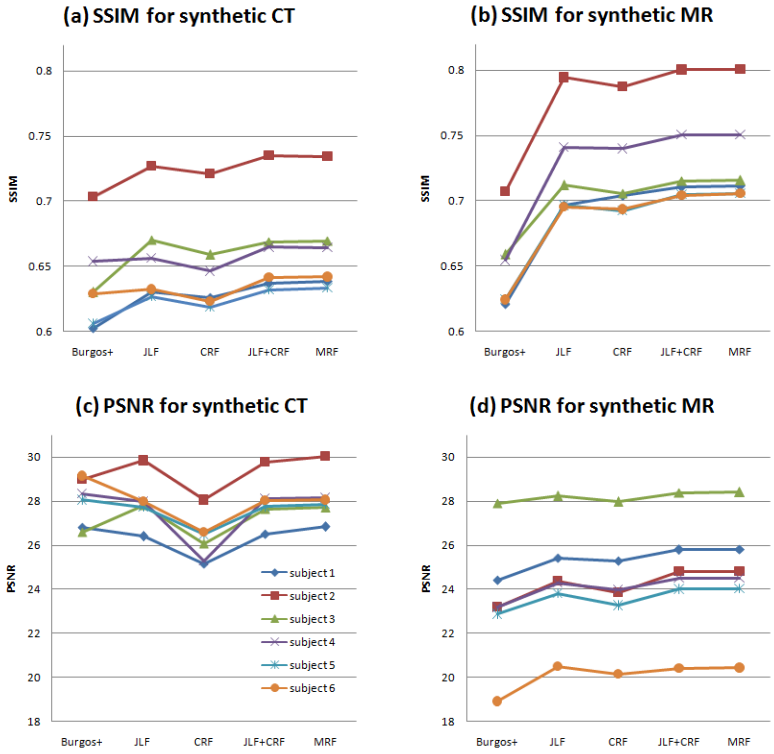


Figure A.2: Evaluation of synthesis result. The six colors are for six subjects.

Table A.1: Evaluation of registration results: Mean (and Std. Dev.) of MSE between reference MR and registered MR image; MI between target CT and registered MR image; p-value of paired-sample t-test for the MSE and MI of the two methods.

	MSE	**MI**
2 Channel CC	$2.746(\pm 0.6492)\times 10^4$	$1.2314(\pm 0.0746)$
Single Channel MI	$3.375(\pm 0.6635)\times 10^4$	$1.2429(\pm 0.1018)$
p-value over Single Channel MI	8.7637e-16	0.1962

To evaluate whether our synthesis method improves multi-modal registration, we carried out a multi-modal registration experiment between the CT image of one subject and the MR image of another subject. The conventional approach for multi-modal registration uses mutual information (MI) as

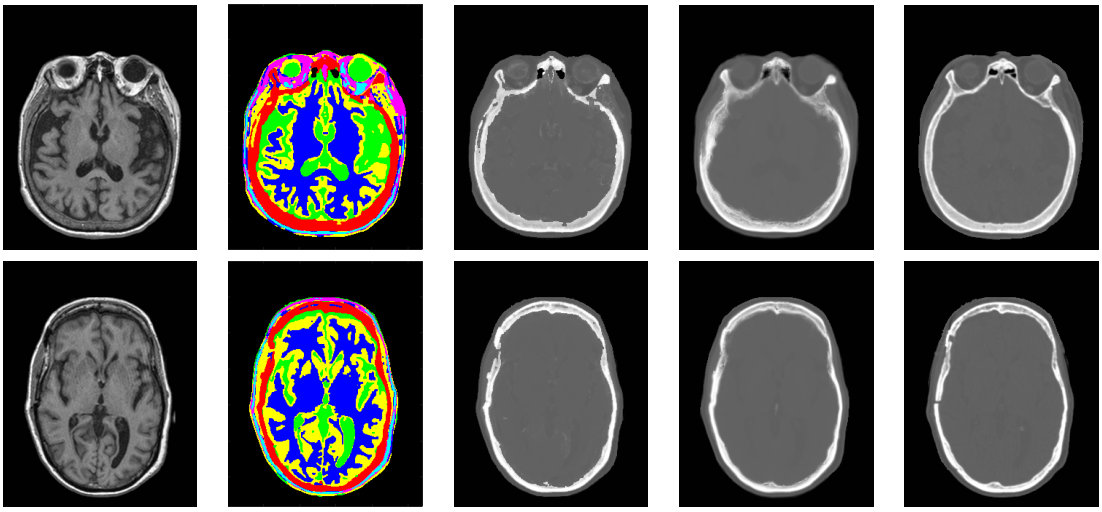


Figure A.3: Synthetic CT images: For two subjects, one in each row, we show the **(a)** input MR image, the **(b)** estimated z-field after MRF smoothing, the CT images generated by **(c)** our method, **(d)** Burgos+, and the **(e)** ground truth.

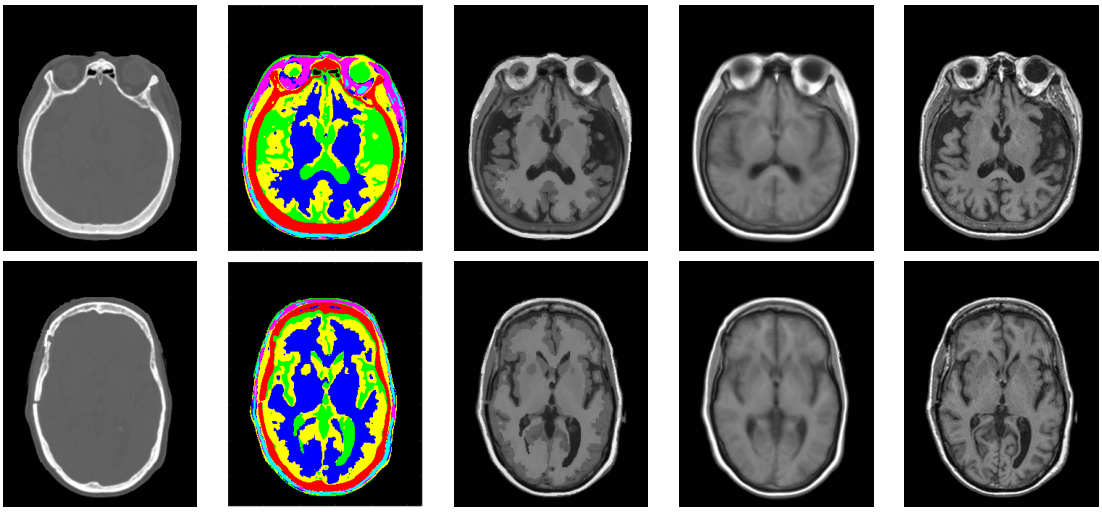


Figure A.4: Synthetic MR images: For two subjects, one in each row, we show the **(a)** input CT image, the **(b)** estimated z-field after MRF smoothing, the MR images generated by **(c)** our method, **(d)** Burgos+, and the **(d)** ground truth.

a similarity metric. With synthetic images, multi-modal registration can be carried out using a two-channel mono-modal registration process [132, 133].

In our case, for registration between Subject 1 and Subject 2, the first channel uses the original CT image of Subject 1 and the synthetic CT for Subject 2. The second channel uses the synthetic MR image of Subject 1 and the original MR image of Subject 2. The metric used in both channels is cross correlation (CC).

We used SyN deformable registration on 6 subjects yielding 30 pairs of registration experiments in all. The single MI registration and two-channel CC registration share the same parameters, including the number of iterations. As the true MR image is known, we compare the transformed MR image to the true MR image for each individual registration experiment. The difference between these two images is measured using both MSE and MI after either two-channel CC or single-channel MI (results are in Table A.1). While the two images are not statistically different according to MI, the two-channel registration approach (which uses our synthetic images) is statistically better than the single-channel MI approach.

A.4 Conclusion

We have presented a bidirectional MR/CT synthesis method based on approximate tissue classification and image segmentation. The method synthesizes CT images from MR images with performance comparable to Burgos et al. [1] and is better than Burgos et al. [1] for synthesizing MR images from CT images. Our method reduces intensity ambiguity by estimating a z-field that is derived from both modalities and can be consistently created given just one modality as input.

Appendix B

Effects of spatial resolution on image registration

Abstract

This appendix presents a theoretical analysis of the effect of spatial resolution on image registration. Based on the assumption of additive Gaussian noise in images, the mean and variance of the distribution of the sum of squared differences (SSD) is computed. Using these computations, we evaluate a distance between the SSD distributions of aligned images and non-aligned images. Experimental results show that by matching the resolutions of the moving and fixed images in the registration one can get a better image registration result. These results agree with our theoretical analysis of SSD, but also reveal that our analysis may be valid for mutual information as well.

B.1 Introduction

Image registration is the process of transforming the coordinate system of a given moving image to that of a fixed image [1]. It is a key component of medical image analysis, including segmentation, multi-modality fusion, longitudinal studies, population modeling, and statistical atlases [140, 141, 142, 143, 144, 145, 146, 147, 148, 149]. Typically, the moving and fixed images have identical digital resolutions, though it is common for interpolation to be used to upsample the low digital resolution image to the higher resolution one. Interpolation blurs edge information; consequently, it is more difficult to align two edges with different spatial resolutions compared to edges with the same resolution. However, the effect of spatial resolution on image registration has not been theoretically discussed before.

There has been a lot of work on using multi-resolution registration schemes based on pyramid representations going back over several years. The advantages of these pyramid representations are reduced computational cost, and the establishment of links between global information and local information [69, 150]. However, the effect of spatial resolution on image registration has not been studied before.

This appendix presents a theoretical analysis of the effect of spatial resolution on image registration. The contributions are summarized as below:

- We develop quantitative guidance to process the images in order to match their resolutions, and a measure for anisotropic spatial resolution

[1] Moving and fixed images are sometimes referred as source and target images.

is additionally discussed based on the idea of isotropic spatial resolution [151].

- We experimentally explore the effects of random noise and spatial resolution in image registration. We assume that the random noise is additive Gaussian, and hence the SSD between the two images can be considered to be a random variable with a distribution. The separability of the SSD distributions of perfectly aligned image pairs and misaligned (in our case, tranlated) image pairs determines how well images can be registered. By assuming that the noise is additive Gaussian, we can estimate the mean and variance of the distribution. From there, we can evaluate a distance between the SSD distributions of aligned images pairs and shifted image pairs. We also present experimental results using the mutual information (MI) [152, 153, 154].

B.2 Theoretical prediction of the effect of spatial resolutions on image registration

B.2.1 Problem setting

Let $\mathbf{x}$ be a voxel coordinate, and $n_1(\mathbf{x})$ and $n_2(\mathbf{x})$ be two independent additive Gaussian $\mathcal{N}(0, \sigma^2)$ noise components. Let $z_1(\mathbf{x})$ be the fixed image and $z_2(\mathbf{x})$ be the moving image, both instances of the same true high resolution (HR) image $f(\mathbf{x})$ with noise $n_1(\mathbf{x})$ and $n_2(\mathbf{x})$, respectively; i.e., $z_i(\mathbf{x}) = f(\mathbf{x}) + n_i(\mathbf{x}) \quad i = 1, 2$. The low resolution (LR) image derived from f is $\tilde{f}$ with corresponding $\tilde{z}_1$ and $\tilde{z}_2$, so that $\tilde{z}_i(\mathbf{x}) = \tilde{f}(\mathbf{x}) + n_i(\mathbf{x}) \quad i = 1, 2$. If $z_2(\mathbf{x} - \mathbf{v}(\mathbf{x}))$

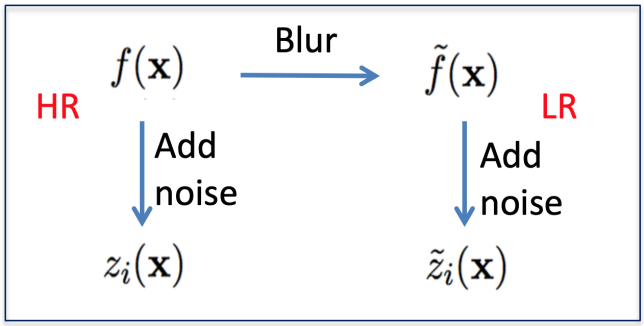


Figure B.1: Explanation of problem setting.

is a transformed noisy version of f, then registering z_1 and z_2 aims to recover $\mathbf{v}(\mathbf{x})$. This setting is summarized in Fig. B.1.

In this section, we take SSD as the metric for registration. In order to analyze the effect on registration, we first analyze the SSD in the following three cases.

Case 1: Registration of two HR images. We wish to compute the mean and variance of SSD $(z_1(\mathbf{x}), z_2(\mathbf{x}-\mathbf{v}(\mathbf{x})))$, which we denote as $\mathcal{S}_{\mathbf{v}}(z_1, z_2)$. Then, $\mathcal{S}_{\mathbf{v}}(z_1, z_2) = \sum_{\mathbf{x}}(z_1(\mathbf{x}) - z_2(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2$, while $\mathcal{S}_0(z_1, z_2) = \sum_{\mathbf{x}}(z_1(\mathbf{x}) - z_2(\mathbf{x}))^2$ is perfect registration.

Case 2: Registration of two LR images. Following the same procedure, we have $\mathcal{S}_{\mathbf{v}}(\tilde{z}_1, \tilde{z}_2) = \text{SSD}\,(\tilde{z}_1(\mathbf{x}), \tilde{z}_2(\mathbf{x}-\mathbf{v}(\mathbf{x})))$.

Case 3: Registration of one HR and one LR images. Following the same procedure, we have $\mathcal{S}_{\mathbf{v}}(z_1, \tilde{z}_2) = \text{SSD}\,(z_1(\mathbf{x}), \tilde{z}_2(\mathbf{x}-\mathbf{v}(\mathbf{x})))$.

Note that $(n_1(\mathbf{x}) - n_2(\mathbf{x})) \sim \mathcal{N}(0, 2\sigma^2)$. Let N denote the number of voxels in the image domain. The mean and variance of $\mathcal{S}_{\mathbf{v}}$ can now be calculated; the results are listed in Table B.1.

In order to obtain a correct registration result, we need $\mathcal{S}_0$ to be less than

Table B.1: Mean and Variance of SSD for different resolution pairs

	Mean $E[\mathcal{S}_\mathbf{v}]$	**Variance** $\mathrm{Var}(\mathcal{S}_\mathbf{v})$
Case 1: $\mathcal{S}_\mathbf{v}(z_1, z_2)$	$\sum_\mathbf{x} (f(\mathbf{x}) - f(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2 + 2N\sigma^2$	$8\sigma^2 \sum_\mathbf{x} (f(\mathbf{x}) - f(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2 + 8N\sigma^4$
Case 2: $\mathcal{S}_\mathbf{v}(\tilde{z}_1, \tilde{z}_2)$	$\sum_\mathbf{x} (\tilde{f}(\mathbf{x}) - \tilde{f}(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2 + 2N\sigma^2$	$8\sigma^2 \sum_\mathbf{x} (\tilde{f}(\mathbf{x}) - \tilde{f}(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2 + 8N\sigma^4$
Case 3: $\mathcal{S}_\mathbf{v}(z_1, \tilde{z}_2)$	$\sum_\mathbf{x} (f(\mathbf{x}) - \tilde{f}(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2 + 2N\sigma^2$	$8\sigma^2 \sum_\mathbf{x} (f(\mathbf{x}) - \tilde{f}(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2 + 8N\sigma^4$

Table B.2: Sensitivity index $d'(\mathcal{S}_\mathbf{v}, \mathcal{S}_0)$ of SSD for images with correct alignment and images with misalignment. A negative value of $d'(\mathcal{S}_\mathbf{v}, \mathcal{S}_0)$ indicates misregistration. A higher value of $d'(\mathcal{S}_\mathbf{v}, \mathcal{S}_0)$ is desired.

Case 1:	$d'_{HR,HR} = d'(\mathcal{S}_\mathbf{v}(z_1, z_2), \mathcal{S}_0(z_1, z_2)) = \frac{\sum_\mathbf{x}(f(\mathbf{x})-f(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2}{2\sigma\sqrt{2N\sigma^2+\sum_\mathbf{x}(f(\mathbf{x})-f(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2}}$
Case 2:	$d'_{LR,LR} = d'(\mathcal{S}_\mathbf{v}(\tilde{z}_1, \tilde{z}_2), \mathcal{S}_0(\tilde{z}_1, \tilde{z}_2)) = \frac{\sum_\mathbf{x}(\tilde{f}(\mathbf{x})-\tilde{f}(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2}{2\sigma\sqrt{2N\sigma^2+\sum_\mathbf{x}(\tilde{f}(\mathbf{x})-\tilde{f}(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2}}$
Case 3:	$d'_{HR,LR} = d'(\mathcal{S}_\mathbf{v}(z_1, \tilde{z}_2), \mathcal{S}_0(z_1, \tilde{z}_2)) = \frac{\sum_\mathbf{x}(f(\mathbf{x})-\tilde{f}(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2-\sum_\mathbf{x}(f(\mathbf{x})-\tilde{f}(\mathbf{x}))^2}{2\sigma\sqrt{2N\sigma^2+\sum_\mathbf{x}(f(\mathbf{x})-\tilde{f}(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2+\sum_\mathbf{x}(f(\mathbf{x})-\tilde{f}(\mathbf{x}))^2}}$

$\mathcal{S}_\mathbf{v}$. How well we can distinguish the distributions of $\mathcal{S}_0(z_1, z_2)$ and $\mathcal{S}_\mathbf{v}(z_1, z_2)$ determines the quality of the registration output. We use the sensitivity index, $d'(\mathcal{S}_\mathbf{v}, \mathcal{S}_0)$, defined as

$$d'(\mathcal{S}_\mathbf{v}, \mathcal{S}_0) = \frac{E[\mathcal{S}_\mathbf{v}] - E[\mathcal{S}_0]}{\sqrt{\frac{1}{2}(\mathrm{Var}(\mathcal{S}_\mathbf{v}) + \mathrm{Var}(\mathcal{S}_0))}}.$$

Taking results from Table B.1, we derive formulas for $d'(\mathcal{S}_\mathbf{v}, \mathcal{S}_0)$ in our three cases, which have shown in Table B.2.

We note that $d'_{HR,LR} < 0$ indicates that the expectation of $\mathcal{S}_v(z_1, \tilde{z}_2)$ is less than $\mathcal{S}_\mathbf{0}(z_1, \tilde{z}_2)$. When this happens, any registration algorithm that uses SSD will misregister the images. The larger we make d' the more confidence we can have in a registration result. This gives us an optimality criterion for matching the resolution of images during registration. We compare d's in order to understand the effect of resolution on image registration.

B.2.2 Claims and Proofs

We now compare $d'_{LR,LR}$, $d'_{HR,HR}$, and $d'_{HR,LR}$.

Claim 1: $\mathbf{d'_{LR,LR} < d'_{HR,HR}}$

Proof 1:

Using a Taylor series expansion and assuming that $\mathbf{v}(\mathbf{x})$ is small,

$$|f(\mathbf{x}) - f(\mathbf{x} - \mathbf{v}(\mathbf{x}))| \quad \sim \quad \mathbf{v}(\mathbf{x})\nabla f(\mathbf{x}),$$

where $\nabla f(\mathbf{x})$ is the gradient of the image.

Since a smoother image has a smaller gradient, we have

$$\sum_{\mathbf{x}} (\tilde{f}(\mathbf{x}) - \tilde{f}(\mathbf{x} - \mathbf{v}(\mathbf{x})))^2 \quad < \quad \sum_{\mathbf{x}} (f(\mathbf{x}) - f(\mathbf{x} - \mathbf{v}(\mathbf{x})))^2.$$

Thus, $d'_{LR,LR} < d'_{HR,HR}$.

Proof 2:

If we assume the image is wide sense stationary (WSS), and the low resolution image $\tilde{f}(\mathbf{x}) = f(\mathbf{x}) * h(\mathbf{x})$, in which $h(\mathbf{x})$ is a low pass filter, then the autocorrelation is $R_{\tilde{f}\tilde{f}}(l) = h(-l) * h(l) * R_{ff}(l)$. In other words, $R_{\tilde{f}\tilde{f}}(l)$ is a

low-pass filtered version of $R_{ff}(l)$. Thus we have:

$$\begin{aligned}
\textstyle\sum_{\mathbf{x}} (f(\mathbf{x}) - f(\mathbf{x} - \mathbf{v}(\mathbf{x})))^2 &= 2 \sum_{\mathbf{x}} \Big(f(\mathbf{x})^2 - f(\mathbf{x}) f(\mathbf{x} - \mathbf{v}(\mathbf{x}) \Big) \\
&\sim 2 \left(R_{ff}(0) - R_{ff}(\mathbf{v}(\mathbf{x})) \right) \\
\textstyle\sum_{\mathbf{x}} (\tilde{f}(\mathbf{x}) - \tilde{f}(\mathbf{x} - \mathbf{v}(\mathbf{x})))^2 &\sim 2 \left(R_{\tilde{f}\tilde{f}}(0) - R_{\tilde{f}\tilde{f}}(\mathbf{v}(\mathbf{x})) \right) \\
&< 2 \left(R_{ff}(0) - R_{ff}(\mathbf{v}(\mathbf{x})) \right)
\end{aligned}$$

Therefore, $\sum_{\mathbf{x}} (\tilde{f}(\mathbf{x}) - \tilde{f}(\mathbf{x} - \mathbf{v}(\mathbf{x})))^2 < \sum_{\mathbf{x}} (f(\mathbf{x}) - f(\mathbf{x} - \mathbf{v}(\mathbf{x})))^2$ and $d'_{LR,LR} < d'_{HR,HR}$. ∎

Implication: We can conclude that $d'_{LR,LR} < d'_{HR,HR}$; thus, when the resolutions of the two images are the same, there is a better chance to obtain a correct registration result with high resolution images.

Claim 2: $\mathbf{d'_{HR,LR} < d'_{HR,HR}}$

Proof:

$$\sum_{\mathbf{x}} (f(\mathbf{x}) - \tilde{f}(\mathbf{x} - \mathbf{v}(\mathbf{x})))^2 + \sum_{\mathbf{x}} (f(\mathbf{x}) - \tilde{f}(\mathbf{x}))^2 \sim 2 \left(R_{ff}(0) + R_{\tilde{f}\tilde{f}}(0) - R_{f\tilde{f}}(0) - R_{f\tilde{f}}(\mathbf{v}(\mathbf{x})) \right)$$

$$2 \left(R_{ff}(0) + R_{\tilde{f}\tilde{f}}(0) - R_{f\tilde{f}}(0) - R_{f\tilde{f}}(\mathbf{v}(\mathbf{x})) \right) < 2 \left(R_{ff}(0) - R_{f\tilde{f}}(\mathbf{v}(\mathbf{x})) \right) < 2 \left(R_{ff}(0) - R_{ff}(\mathbf{v}(\mathbf{x})) \right)$$

Therefore,

$$\sum_{\mathbf{x}} (f(\mathbf{x}) - \tilde{f}(\mathbf{x} - \mathbf{v}(\mathbf{x})))^2 + \sum_{\mathbf{x}} (f(\mathbf{x}) - \tilde{f}(\mathbf{x}))^2 < \sum_{\mathbf{x}} (f(\mathbf{x}) - f(\mathbf{x} - \mathbf{v}(\mathbf{x})))^2$$

and

$$\begin{aligned} & \frac{\sum_{\mathbf{x}}(f(\mathbf{x})-\tilde{f}(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2-\sum_{\mathbf{x}}(f(\mathbf{x})-\tilde{f}(\mathbf{x}))^2}{2\sigma\sqrt{N2\sigma^2+\sum_{\mathbf{x}}(f(\mathbf{x})-\tilde{f}(\mathbf{x}))^2+\sum_{\mathbf{x}}(f(\mathbf{x})-\tilde{f}(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2}} \\ < & \frac{\sum_{\mathbf{x}}(f(\mathbf{x})-\tilde{f}(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2+\sum_{\mathbf{x}}(f(\mathbf{x})-\tilde{f}(\mathbf{x}))^2}{2\sigma\sqrt{N2\sigma^2+\sum_{\mathbf{x}}(f(\mathbf{x})-\tilde{f}(\mathbf{x}))^2+\sum_{\mathbf{x}}(f(\mathbf{x})-\tilde{f}(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2}} \\ < & \frac{\sum_{\mathbf{x}}(f(\mathbf{x})-f(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2}{2\sigma\sqrt{N2\sigma^2+\sum_{\mathbf{x}}(f(\mathbf{x})-f(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2}}. \end{aligned}$$

Thus $d'_{HR,LR} < d'_{HR,HR}$, as required. ∎

Implication: We can conclude that compared to images with different resolutions, there is a higher chance to obtain better registration results for a pair of high resolution images.

Claim 3: $\mathbf{d'_{HR,LR} < d'_{LR,LR}}$ unless the resolution of the two images are only slightly different.

Proof:

$$\begin{aligned} \sum_{\mathbf{x}}(f(\mathbf{x})-\tilde{f}(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2-\sum_{\mathbf{x}}(f(\mathbf{x})-\tilde{f}(\mathbf{x}))^2 & \sim 2\left(R_{\tilde{f}\tilde{f}}(0)-R_{f\tilde{f}}(\mathbf{v}(\mathbf{x}))\right) \\ 2\left(R_{ff}(0)+R_{\tilde{f}\tilde{f}}(0)-R_{f\tilde{f}}(0)-R_{f\tilde{f}}(\mathbf{v}(\mathbf{x}))\right) & > 2\left(R_{\tilde{f}\tilde{f}}(0)-R_{f\tilde{f}}(\mathbf{v}(\mathbf{x}))\right) \\ & > 2\left(R_{\tilde{f}\tilde{f}}(0)-R_{\tilde{f}\tilde{f}}(\mathbf{v}(\mathbf{x}))\right) \\ \sum_{\mathbf{x}}(f(\mathbf{x})-\tilde{f}(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2+\sum_{\mathbf{x}}(f(\mathbf{x})-\tilde{f}(\mathbf{x}))^2 & > \sum_{\mathbf{x}}(f(\mathbf{x})-\tilde{f}(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2-\sum_{\mathbf{x}}(f(\mathbf{x})-\tilde{f}(\mathbf{x}))^2 \\ & > \sum_{\mathbf{x}}(\tilde{f}(\mathbf{x})-\tilde{f}(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2 \end{aligned}$$

Whether

$$\frac{\sum_{\mathbf{x}}(f(\mathbf{x})-\tilde{f}(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2-\sum_{\mathbf{x}}(f(\mathbf{x})-\tilde{f}(\mathbf{x}))^2}{2\sigma\sqrt{N2\sigma^2+\sum_{\mathbf{x}}(f(\mathbf{x})-\tilde{f}(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2+\sum_{\mathbf{x}}(f(\mathbf{x})-\tilde{f}(\mathbf{x}))^2}}$$

$$< \quad \frac{\sum_{\mathbf{x}}(\tilde{f}(\mathbf{x})-\tilde{f}(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2}{2\sigma\sqrt{N2\sigma^2+\sum_{\mathbf{x}}(\tilde{f}(\mathbf{x})-\tilde{f}(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2}}$$

is valid depends on the relationship between $\tilde{f}(\mathbf{x})$ and $f(\mathbf{x})$. ∎

Implications:

1. If there is a sufficient difference between $\tilde{f}(\mathbf{x})$ and $f(\mathbf{x})$, which indicates a large enough $\sum_{\mathbf{x}}(f(\mathbf{x})-\tilde{f}(\mathbf{x}))^2$, then $d'_{HR,LR} < d'_{LR,LR} < d'_{HR,HR}$.

2. If $\tilde{f}(\mathbf{x}) \approx f(\mathbf{x})$, then $d'_{HR,LR} \approx d'_{HR,HR} > d'_{LR,LR}$. However, this is not a situation that concerns us, since it indicates that there is only a slight difference between resolutions.

3. If there is a large difference between $\tilde{f}(\mathbf{x})$ and $f(\mathbf{x})$, which makes $\sum_{\mathbf{x}}(f(\mathbf{x})-\tilde{f}(\mathbf{x}))^2 > \sum_{\mathbf{x}}(f(\mathbf{x})-\tilde{f}(\mathbf{x}-\mathbf{v}(\mathbf{x})))^2$, then $d'_{HR,LR} < 0 < d'_{LR,LR} < d'_{HR,HR}$, which indicates that misregistraion is more likely to occur between HR and LR images.

B.2.3 Conclusions

We claim that $d'_{LR,LR} < d'_{HR,HR}$, which shows that the higher the resolution of the images, the more confidence we are about the registration results. We also claim that $d'_{HR,LR} < d'_{HR,HR}$ and that if the resolution difference between f and $\tilde{f}$ is sufficiently enough, then $d'_{HR,LR} < d'_{LR,LR}$, which may be counterintuitive.

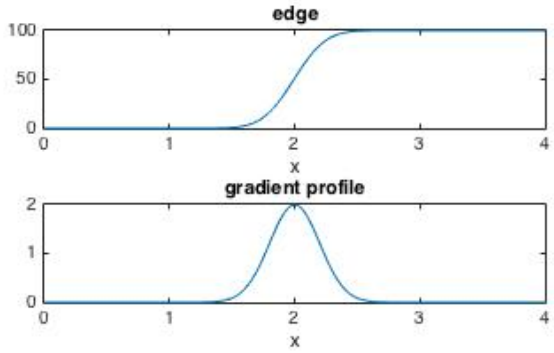


Figure B.2: An example of edge and gradient profile.

It appears that the HR image carries more information; thus, two LR images should have resulted in a worse registration result; however, our analysis reveals the opposite. There is a larger chance to obtain a better registration result when dealing with similar resolution images as compared to images with different resolutions unless the resolution difference is very small.

B.3 An edge-based method to measure resolution

Next we develop a measure of anisotropic spatial resolution based on the idea of isotropic spatial resolution [151]. We have previously shown that matching the resolution of two images increases the confidence of acquiring a more accurate registration result. Thus, we need to be able to measure the resolution of the images. To do so, we consider the edges and the gradient profiles of images. An example of edge and gradient profile is shown in Fig. B.2.

The gradient profile of the LR edge is more spread out; therefore, we can use the full width at half maximum (FWHM) of the gradient curve as a measure of the resolution.

Our algorithm to identify gradient curves and thus their FWHM is:

1. Use the Canny edge detector to identify edge voxels.

2. Find the gradient direction at each edge voxel, then collect the edge voxels that have similar gradient directions with the target direction.

3. Apply blob matching to the gradient profile in order to find the center and range of each edge, and then calculate the FWHM.

B.4 Experiments

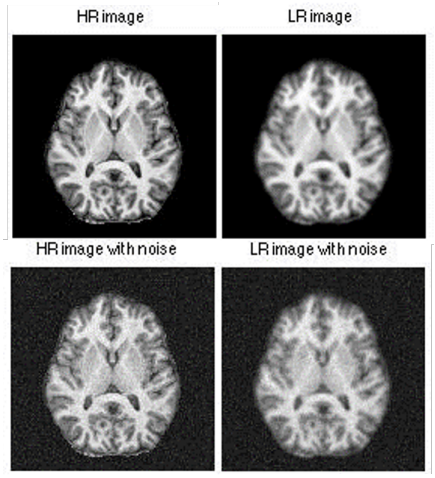
B.4.1 Effect of spatial resolution on image registration

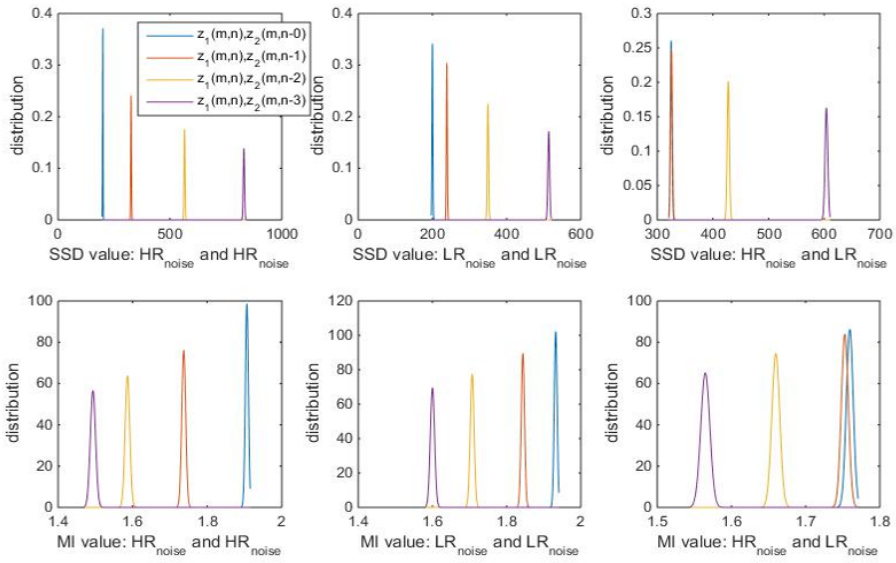
The objective of Experiment 1 is to verify the claims in Sec. B.2 by comparing $d'_{HR,HR}$, $d'_{LR,LR}$ and $d'_{HR,LR}$. Specifically, we aim to verify that (a) $d'_{HR,HR} > d'_{LR,LR}$, (b) $d'_{HR,HR} > d'_{HR,LR}$, and (c) $d'_{LR,LR} > d'_{HR,LR}$.

Using the MRI 2D slices in the multi-modal reproducibility resource dataset [155] as the true HR image f, we simulate a noisy HR image z_1 (see Fig. B.3). The second noisy HR image z_2 is a shifted version of f with different random noise. We consider 4 different shifts (v) which are translations in the y-plane by $0, 1, 2$, and 3 voxels. For each of these shifts, we calculate the SSD between z_1 and z_2. We do this for 500 simulations of z_1 and z_2 and build a distribution of SSD values for each of the 4 shifts. We calculate the sensitivity $d'_{HR,HR}$ index, between the SSD distributions for the different shifts, for HR images. These values are recorded in the first row of the SSD portion of Table B.3.

Similarly, we simulate LR images by blurring f (see Fig B.3) using a Gaussian blur kernel with a standard deviation (std) of 1.5 and by then adding

(a) Subject images: The HR image is a 256×256 2D MR brain image, while the LR image is obtained by filtering the HR image using Gaussian kernel with a standard deviation (std) of 1.5. All the image intensities are normalized between $[0, 255]$. The additive Gaussian noise has a std $\sigma = 10$.





(b) Distribution of SSD and MI: The distributions of $\mathcal{S}_{\mathbf{v}}(z_1, z_2)$ (upper row) and $\mathrm{MI}(z_1, z_2)$ (lower row). The left column represents the results of the experiment implemented on a pair HR images. While the middle column displays the results of a pair of LR images, and the right are the results of a HR image and a LR image. In the right column, the distribution of SSD and MI with $v = 0$ (blue curve) and $v = 1$ (orange curve) are too close to distinguish, which indicates that we are more likely to get misregistration when $v = 1$.

Figure B.3: Experiment results on SSD and MI distributions regard to image pairs with different spatial resolution.

Table B.3: Experiment results of effects of spatial resolution on image registration: d' for SSD and MI for image pairs shifted by $v = 1, 2, 3$ voxels. It agrees with our claim that $d'_{HR,LR} < d'_{LR,LR} < d'_{HR,HR}$.

SSD			
	1	2	3
$d'_{HR,HR}$	96.0	201.5	272.9
$d'_{LR,LR}$	31.8	102.2	172.6
$d'_{HR,LR}$	0.34	61.6	146.0

MI			
v	1	2	3
$d'_{HR,HR}$	37.5	65.0	76.1
$d'_{LR,LR}$	21.0	50.0	68.2
$d'_{HR,LR}$	1.4	20.0	37.4

Gaussian noise. We then calculate $d'_{LR,LR}$, which we show in the second row of the SSD portion of Table B.3. For all the shifts, it is apparent that $d'_{HR,HR} > d'_{LR,LR}$, thus verifying our first claim. Next, we choose z_1 as a noisy HR image, and z_2 as a noisy LR image, carry out the simulations, and calculate $d'_{HR,LR}$, which is the last row of the SSD portion of Table B.3. Comparing this row to the first and second rows, it is clear that for all shifts, $d'_{HR,HR} > d'_{HR,LR}$, $d'_{LR,LR} > d'_{HR,LR}$, thus verifying our second and third claims.

If instead of SSD, we use the mutual information (MI) in our simulations, we observe that our claims are still true, as is demonstrated in the MI part of Table B.3. This is an empirical result which points to an interesting connection between using the SSD and the MI as similarity measures, but we do not have at this point a theoretical proof for the relationships between $d'_{HR,HR}$, $d'_{LR,LR}$, and $d'_{HR,LR}$ on the MI distributions.

Figure. B.3 (b) shows our fits for the SSD (first row) and MI (second row) distributions for four different shifts (0, 1, 2 , 3). Each column corresponds to the pairs HR-HR, LR-LR and HR-LR of images. Visually, we can appreciate the fact that the SSD distribution for $v = 0$ is far apart from the SSD distribution for $v = 1$ in the case of HR-HR and the LR-LR. However, for the HR-LR case,

the SSD distributions for $v = 0$ and $v = 1$ overlap with each other, indicating that a registration algorithm can result in a lower SSD for a shift of 1 voxel, which is clearly not the correct result leading to an undesirable behavior.

B.4.2 Resolution measure

We use the measure in Sec. B.3 to estimate the image resolution. The HR data we use are BrainWeb images [18]. The LR data is the Gaussian blurred results from the HR data, with blur kernel being $1.5 \times 0.5 \times 0$mm. All the image intensities are normalized between [0, 255]. The additive Gaussian noise has a standard deviation $= 1$, which makes the SNR$\approx$ 38. The measured resolutions are listed in Table B.4(a). The ground truth of r_{HR} should be $1 \times 1 \times 1$mm, whereas the ground truth of r_{LR} should be $1.80 \times 1.12 \times 1$mm. The ground truth of $\sqrt{r_{LR}^2 - r_{HR}^2}$ should be $1.5 \times 0.5 \times 0$mm. We can see that the measured resolutions are close to the ground truth.

We then repeat the SSD experiment presented in Sec. B.4.1 with $v = 1$ on three directions. To obtain blurred HR images, we apply a lowpass filter on the HR noisy images using a Gaussian blur kernel with std of $\sqrt{r_{\mathrm{LR}}^2 - r_{\mathrm{HR}}^2}$. The result is shown in Table B.4(b). It can seen that $d'_{blurHR,LR} > d'_{HR,LR}$. Therefore, matching the resolution of two subject images can give a better image registration result.

B.5 Conclusion

In this work, we analyzed the effect of resolution on image registration. Our theoretical analysis and experiments show that 1) images with the same

Table B.4: Effects of matching resolution on image registration: (a) Measured resolution of the subject images from BrainWeb. The ground truth of r_{HR} should be $1 \times 1 \times 1$mm. The ground truth of r_{LR} should be $1.80 \times 1.12 \times 1$mm. The ground truth of $\sqrt{r_{LR}^2 - r_{HR}^2}$ should be $1.5 \times 0.5 \times 0$mm. **(b)** d' for SSD. The blurHR image is blurred HR image with blur kernel std being $1.55 \times 0.45 \times 0$mm.

(a) Measured resolution [mm]

v	x	y	z
r_{HR}	1.1	1.0	1.2
r_{LR}	1.9	1.1	1.2
$\sqrt{r_{LR}^2 - r_{HR}^2}$	1.55	0.45	0

(b)SSD

v	x	y	z
$d'_{HR,HR}$	42.1	36.7	33.9
$d'_{LR,LR}$	38.1	29.7	30.2
$d'_{HR,LR}$	29.6	22.6	22.6
$d'_{blurHR,LR}$	36.7	28.8	29.3

resolution can be registered accurately with more confidence; and 2) matching the resolution of two subject images can give better guarantees for an image registration result.

This work did theoretical analysis only for registration using SSD. The experiment was performed only on image translation. Future work will include a theoretical analysis of other cost functions and other transforms. Also the resolution measure is sensitive to parameter tuning and does not perform well on real clinical data. Future work will include a more robust resolution measure for real data.

Bibliography

[1] N. Burgos, M. J. Cardoso, K. Thielemans, M. Modat, S. Pedemonte, J. Dickson, A. Barnes, R. Ahmed, C. J. Mahoney, J. M. Schott, J. S. Duncan, D. Atkinson, S. R. Arridge, B. F. Hutton, and S. Ourselin, "Attenuation correction synthesis for hybrid PET-MR scanners: application to brain studies," *IEEE Transactions on Medical Imaging*, vol. 33, no. 12, pp. 2332–2341, 2014.

[2] B. E. Dewey, C. Zhao, A. Carass, J. Oh, P. A. Calabresi, P. C. van Zijl, and J. L. Prince, "Deep harmonization of inconsistent MR data for consistent volume segmentation," in *International Workshop on Simulation and Synthesis in Medical Imaging*. Springer, 2018, pp. 20–30.

[3] S. Roy, A. Carass, and J. Prince, "A compressed sensing approach for MR tissue contrast synthesis," in *Biennial International Conference on Information Processing in Medical Imaging*. Springer, 2011, pp. 371–383.

[4] R. Bitar, G. Leung, R. Perng, S. Tadros, A. R. Moody, J. Sarrazin, C. McGregor, M. Christakis, S. Symons, A. Nelson *et al.*, "MR pulse sequences: what every radiologist wants to know but is afraid to ask," *Radiographics*, vol. 26, no. 2, pp. 513–537, 2006.

[5] M. Chen, A. Carass, A. Jog, J. Lee, S. Roy, and J. L. Prince, "Cross contrast multi-channel image registration using image synthesis for MR brain images," *Medical Image Analysis*, vol. 36, pp. 2–14, 2017.

[6] J. Lee, A. Carass, A. Jog, C. Zhao, and J. L. Prince, "Multi-atlas-based CT synthesis from conventional MRI with patch-based refinement for mri-based radiotherapy planning," in *Medical Imaging 2017: Image Processing*, vol. 10133. International Society for Optics and Photonics, 2017, p. 101331I.

[7] N. Burgos, F. Guerreiro, J. McClelland, B. Presles, M. Modat, S. Nill, D. Dearnaley, N. Desouza, U. Oelfke, A.-C. Knopf, S. Ourselin, and M. J. Cardoso, "Iterative framework for the joint segmentation and CT synthesis of MR images: application to mri-only radiotherapy treatment planning," *Physics in Medicine & Biology*, vol. 62, no. 11, p. 4237, 2017.

[8] D. Andreasen, J. M. Edmund, V. Zografos, B. H. Menze, and K. Van Leemput, "Computed tomography synthesis from magnetic resonance images in the pelvis using multiple random forests and auto-context features," in *Medical Imaging 2016: Image Processing*, vol. 9784. International Society for Optics and Photonics, 2016, p. 978417.

[9] T. Huynh, Y. Gao, J. Kang, L. Wang, P. Zhang, J. Lian, and D. Shen, "Estimating CT image from MRI data using structured random forest and auto-context model," *IEEE Transactions on Medical Imaging*, vol. 35, no. 1, pp. 174–183, 2015.

[10] A. Jog, A. Carass, S. Roy, D. L. Pham, and J. L. Prince, "Random forest regression for magnetic resonance image synthesis," *Medical Image Analysis*, vol. 35, pp. 475–488, 2017.

[11] H. Van Nguyen, K. Zhou, and R. Vemulapalli, "Cross-domain synthesis of medical images using efficient location-sensitive deep network," in *International Conference on Medical Image Computing and Computer-assisted Intervention*. Springer, 2015, pp. 677–684.

[12] W. Wein, B. Röper, and N. Navab, "Automatic registration and fusion of ultrasound with CT for radiotherapy," in *International Conference on Medical Image Computing and Computer-assisted Intervention*. Springer, 2005, pp. 303–311.

[13] L. Mercier, V. Fonov, C. Haegelen, R. F. Del Maestro, K. Petrecca, and D. L. Collins, "Comparing two approaches to rigid registration of three-dimensional ultrasound and magnetic resonance images for neurosurgery," *International Journal of Computer Assisted Radiology and Surgery*, vol. 7, no. 1, pp. 125–136, 2012.

[14] V. Sa-Ing, K. Wangkaoom, and S. S. Thongvigitmanee, "Automatic dental arch detection and panoramic image synthesis from CT images," in *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2013, pp. 6099–6102.

[15] C. Won Kim and J. H. Kim, "Realistic simulation of reduced-dose CT with noise modeling and sinogram synthesis using dicom CT images," *Medical Physics*, vol. 41, no. 1, p. 011901, 2014.

[16] S. J. Riederer, S. Suddarth, S. Bobman, J. Lee, H. Wang, and J. R. MacFall, "Automated MR image synthesis: feasibility studies." *Radiology*, vol. 153, no. 1, pp. 203–206, 1984.

[17] R. Shams, R. Hartley, and N. Navab, "Real-time simulation of medical ultrasound from CT images," in *International Conference on Medical Image Computing and Computer-assisted Intervention*. Springer, 2008, pp. 734–741.

[18] C. A. Cocosco, V. Kollokian, R. K.-S. Kwan, G. B. Pike, and A. C. Evans, "Brainweb: Online interface to a 3D MRI simulated brain database," in *NeuroImage*, vol. 5. Elsevier, 1997, p. 425.

[19] A. Jog, S. Roy, A. Carass, and J. L. Prince, "Pulse sequence based multi-acquisition MR intensity normalization," in *Medical Imaging 2013: Image Processing*, vol. 8669. International Society for Optics and Photonics, 2013, p. 86692H.

[20] T. Sekine, A. Buck, G. Delso, E. E. Ter Voert, M. Huellner, P. Veit-Haibach, and G. Warnock, "Evaluation of atlas-based attenuation correction for integrated PET/MR in human brain: application of a head atlas and comparison to true CT-based attenuation correction," *Journal of Nuclear Medicine*, vol. 57, no. 2, pp. 215–220, 2016.

[21] H. Arabi, N. Koutsouvelis, M. Rouzaud, R. Miralbell, and H. Zaidi, "Atlas-guided generation of pseudo-CT images for MRI-only and hybrid PET-MRI-guided radiotherapy treatment planning," *Physics in Medicine & Biology*, vol. 61, no. 17, p. 6531, 2016.

[22] F. Guerreiro, N. Burgos, A. Dunlop, K. Wong, I. Petkar, C. Nutting, K. Harrington, S. Bhide, K. Newbold, D. Dearnaley, N. Souza, V. Morgan, J. McClelland, S. Nill, M. Cardoso, S. Ourselin, U. Oelfke, and A. Knopf, "Evaluation of a multi-atlas CT synthesis approach for MRI-only radiotherapy treatment planning," *Physica Medica*, vol. 35, pp. 7–17, 2017.

[23] X. Cao, J. Yang, Y. Gao, Y. Guo, G. Wu, and D. Shen, "Dual-core steered non-rigid registration for multi-modal images via bi-directional image synthesis," *Medical Image Analysis*, vol. 41, pp. 18–31, 2017.

[24] S. Roy, A. Carass, and J. L. Prince, "Magnetic resonance image example-based contrast synthesis," *IEEE Transactions on Medical Imaging*, vol. 32, no. 12, pp. 2348–2363, 2013.

[25] D. Nie, R. Trullo, J. Lian, C. Petitjean, S. Ruan, Q. Wang, and D. Shen, "Medical image synthesis with context-aware generative adversarial networks," in *International Conference on Medical Image Computing and Computer-assisted Intervention*. Springer, 2017, pp. 417–425.

[26] X. Han, "MR-based synthetic CT generation using a deep convolutional neural network method," *Medical Physics*, vol. 44, no. 4, pp. 1408–1419, 2017.

[27] S. U. Dar, M. Yurt, L. Karacan, A. Erdem, E. Erdem, and T. Çukur, "Image synthesis in multi-contrast MRI with conditional generative adversarial networks," *IEEE Transactions on Medical Imaging*, 2019.

[28] H. Greenspan, "Super-resolution in medical imaging," *The Computer Journal*, vol. 52, no. 1, pp. 43–63, 2008.

[29] F. Lüsebrink, A. Wollrab, and O. Speck, "Cortical thickness determination of the human brain using high resolution 3T and 7T MRI data," *NeuroImage*, vol. 70, pp. 122–131, 2013.

[30] C. Zhao, A. Carass, A. Jog, and J. L. Prince, "Effects of spatial resolution on image registration," in *Medical Imaging 2016: Image Processing*, vol. 9784. International Society for Optics and Photonics, 2016, p. 97840Y.

[31] J. Woo, E. Z. Murano, M. Stone, and J. L. Prince, "Reconstruction of high-resolution tongue volumes from mri," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 12, pp. 3511–3524, 2012.

[32] M. Ebner, M. Chouhan, P. A. Patel, D. Atkinson, Z. Amin, S. Read, S. Punwani, S. Taylor, T. Vercauteren, and S. Ourselin, "Point-spread-function-aware slice-to-volume registration: application to upper abdominal MRI super-resolution," in *Reconstruction, Segmentation, and Analysis of Medical Images*. Springer, 2016, pp. 3–13.

[33] G. S. V. Chilla, C. H. Tan, and C. L. Poh, "Deformable registration-based super-resolution for isotropic reconstruction of 4-D MRI volumes," *IEEE Journal of Biomedical and Health Informatics*, vol. 21, no. 6, pp. 1617–1624, 2017.

[34] J. Jurek, M. Kociński, A. Materka, M. Elgalal, and A. Majos, "CNN-based superresolution reconstruction of 3D MR images using thick-slice scans," *Biocybernetics and Biomedical Engineering*, vol. 40, no. 1, pp. 111–125, 2020.

[35] F. Shi, J. Cheng, L. Wang, P.-T. Yap, and D. Shen, "LRTV: MR image super-resolution with low-rank and total variation regularizations," *IEEE Transactions on Medical Imaging*, vol. 34, no. 12, pp. 2459–2466, 2015.

[36] R. Timofte, E. Agustsson, L. Van Gool, M.-H. Yang, and L. Zhang, "NTIRE 2017 challenge on single image super-resolution: methods and results," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017, pp. 114–125.

[37] R. Timofte, S. Gu, J. Wu, and L. Van Gool, "NTIRE 2018 challenge on single image super-resolution: methods and results," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 852–863.

[38] J. Cai, S. Gu, R. Timofte, and L. Zhang, "NTIRE 2019 challenge on real image super-resolution: methods and results," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019.

[39] J.-B. Huang, A. Singh, and N. Ahuja, "Single image super-resolution from transformed self-exemplars," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5197–5206.

[40] F. Rousseau, "Brain hallucination," in *European Conference on Computer Vision*. Springer, 2008, pp. 497–508.

[41] J. V. Manjón, P. Coupé, A. Buades, D. L. Collins, and M. Robles, "MRI superresolution using self-similarity and image priors," *Journal of Biomedical Imaging*, vol. 2010, p. 17, 2010.

[42] C. Zhao, A. Carass, J. Lee, Y. He, and J. L. Prince, "Whole brain segmentation and labeling from CT using synthetic MR images," in *International Workshop on Machine Learning in Medical Imaging*. Springer, 2017, pp. 291–298.

[43] C. Zhao, A. Carass, B. E. Dewey, J. Woo, J. Oh, P. A. Calabresi, D. S. Reich, P. Sati, D. L. Pham, and J. L. Prince, "A deep learning based anti-aliasing self super-resolution algorithm for MRI," in *International Conference on Medical Image Computing and Computer-assisted Intervention*. Springer, 2018, pp. 100–108.

[44] C. Zhao, A. Carass, B. E. Dewey, and J. L. Prince, "Self super-resolution for magnetic resonance images using deep networks," in *Biomedical Imaging (ISBI 2018), 2018 IEEE 15th International Symposium on*. IEEE, 2018, pp. 365–368.

[45] C. Zhao, M. Shao, A. Carass, H. Li, B. E. Dewey, L. M. Ellingsen, J. Woo, M. A. Guttman, A. M. Blitz, M. Stone, P. A. Calabresi, H. Halperin, and J. L. Prince, "Applications of a deep learning method for anti-aliasing and super-resolution in MRI," *Magnetic Resonance Imaging*, 2019.

[46] C. Zhao, S. Son, Y. Kim, and J. L. Prince, "iSMORE: An Iterative Self Super-Resolution Algorithm," in *International Workshop on Simulation and Synthesis in Medical Imaging*. Springer, 2019, pp. 130–139.

[47] C. Zhao, A. Carass, J. Lee, A. Jog, and J. L. Prince, "A supervoxel based random forest synthesis framework for bidirectional MR/CT synthesis,"

in *International Workshop on Simulation and Synthesis in Medical Imaging*. Springer, 2017, pp. 33–40.

[48] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics*, 2011, pp. 315–323.

[49] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1026–1034.

[50] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[51] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *International Conference on Machine Learning*, vol. 37, pp. 448–456, 2015.

[52] S. Santurkar, D. Tsipras, A. Ilyas, and A. Madry, "How does batch normalization help optimization?" in *Advances in Neural Information Processing Systems*, 2018, pp. 2483–2493.

[53] Y. Nesterov, "Introductory lectures on convex programming volume I: Basic course," *Lecture Notes*, vol. 3, no. 4, p. 5, 1998.

[54] X. Li, S. Chen, X. Hu, and J. Yang, "Understanding the disharmony between dropout and batch normalization by variance shift," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2682–2690.

[55] I. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio, "Maxout networks," in *International Conference on Machine Learning*, 2013, pp. 1319–1327.

[56] G. F. Montufar, R. Pascanu, K. Cho, and Y. Bengio, "On the number of linear regions of deep neural networks," in *Advances in Neural Information Processing Systems*, 2014, pp. 2924–2932.

[57] M. Raghu, B. Poole, J. Kleinberg, S. Ganguli, and J. S. Dickstein, "On the expressive power of deep neural networks," in *Proceedings of the 34th International Conference on Machine Learning*, vol. 70, 2017, pp. 2847–2854.

[58] N. Akhtar and A. Mian, "Threat of adversarial attacks on deep learning in computer vision: A survey," *IEEE Access*, vol. 6, pp. 14 410–14 430, 2018.

[59] G. Ghiasi, T.-Y. Lin, and Q. V. Le, "Dropblock: A regularization method for convolutional networks," in *Advances in Neural Information Processing Systems*, 2018, pp. 10 727–10 737.

[60] J. Tompson, R. Goroshin, A. Jain, Y. LeCun, and C. Bregler, "Efficient object localization using convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 648–656.

[61] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *arXiv preprint arXiv:1607.06450*, 2016.

[62] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Instance normalization: The missing ingredient for fast stylization," *arXiv preprint arXiv:1607.08022*, 2016.

[63] Y. Wu and K. He, "Group normalization," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 3–19.

[64] R. Girshick, "Fast R-CNN," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1440–1448.

[65] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1874–1883.

[66] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[67] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-assisted Intervention*. Springer, 2015, pp. 234–241.

[68] P. Burt and E. Adelson, "The Laplacian pyramid as a compact image code," *IEEE Transactions on Communications*, vol. 31, no. 4, pp. 532–540, 1983.

[69] S. G. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 11, no. 7, pp. 674–693, 1989.

[70] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, "3D U-Net: learning dense volumetric segmentation from sparse annotation," in *International Conference on Medical Image Computing and Computer-assisted Intervention*. Springer, 2016, pp. 424–432.

[71] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *2016 Fourth International Conference on 3D Vision (3DV)*. IEEE, 2016, pp. 565–571.

[72] S. K. Zhou, H. Greenspan, and D. Shen, *Deep learning for medical image analysis*. Academic Press, 2017.

[73] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. Van Der Laak, B. Van Ginneken, and C. I. Sánchez, "A survey on deep learning in medical image analysis," *Medical Image Analysis*, vol. 42, pp. 60–88, 2017.

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

[75] 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*, 2017, pp. 136–144.

[76] A. Veit, M. J. Wilber, and S. Belongie, "Residual networks behave like ensembles of relatively shallow networks," in *Advances in Neural Information Processing Systems*, 2016, pp. 550–558.

[77] Z. Wu, C. Shen, and A. Van Den Hengel, "Wider or deeper: Revisiting the resnet model for visual recognition," *Pattern Recognition*, vol. 90, pp. 119–133, 2019.

[78] 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*, 2016, pp. 1646–1654.

[79] 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*, 2017, pp. 3147–3155.

[80] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4681–4690.

[81] 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)*, 2018, pp. 286–301.

[82] J. Yu, Y. Fan, J. Yang, N. Xu, Z. Wang, X. Wang, and T. Huang, "Wide activation for efficient and accurate image super-resolution," *arXiv preprint arXiv:1808.08718*, 2018.

[83] R. Li, W. Zhang, H.-I. Suk, L. Wang, J. Li, D. Shen, and S. Ji, "Deep learning based imaging data completion for improved brain disease diagnosis," in *International Conference on Medical Image Computing and Computer-assisted Intervention*. Springer, 2014, pp. 305–312.

[84] H. Wang, J. W. Suh, S. R. Das, J. B. Pluta, C. Craige, and P. A. Yushkevich, "Multi-atlas segmentation with joint label fusion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 3, pp. 611–623, 2012.

[85] K. Kamnitsas, C. Ledig, V. F. Newcombe, J. P. Simpson, A. D. Kane, D. K. Menon, D. Rueckert, and B. Glocker, "Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation," *Medical Image Analysis*, vol. 36, pp. 61–78, 2017.

[86] B. Fischl, "Freesurfer," *NeuroImage*, vol. 62, no. 2, pp. 774–781, 2012.

[87] P. Moeskops, M. A. Viergever, A. M. Mendrik, L. S. de Vries, M. J. Benders, and I. Išgum, "Automatic segmentation of MR brain images with a convolutional neural network," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1252–1261, 2016.

[88] R. Manniesing, M. T. Oei, L. J. Oostveen, J. Melendez, E. J. Smit, B. Platel, C. I. Sánchez, F. J. Meijer, M. Prokop, and B. van Ginneken, "White matter and gray matter segmentation in 4d computed tomography," *Scientific Reports*, vol. 7, 2017.

[89] Q. Hu, G. Qian, A. Aziz, and W. L. Nowinski, "Segmentation of brain from computed tomography head images," in *Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the*. IEEE, 2006, pp. 3375–3378.

[90] C. R. Ng, J. C. M. Than, N. M. Noor, and O. M. Rijal, "Preliminary brain region segmentation using FCM and graph cut for CT scan images," in *BioSignal Analysis, Processing and Systems (ICBAPS), 2015 International Conference on*. IEEE, 2015, pp. 52–56.

[91] V. Gupta, W. Ambrosius, G. Qian, A. Blazejewska, R. Kazmierski, A. Urbanik, and W. L. Nowinski, "Automatic segmentation of cerebrospinal fluid, white and gray matter in unenhanced computed tomography images," *Academic Radiology*, vol. 17, no. 11, pp. 1350–1358, 2010.

[92] A. Kemmling, H. Wersching, K. Berger, S. Knecht, C. Groden, and I. Nölte, "Decomposing the Hounsfield unit," *Clinical Neuroradiology*, vol. 22, no. 1, pp. 79–91, 2012.

[93] S. Dodge and L. Karam, "Understanding how image quality affects deep neural networks," in *Quality of Multimedia Experience (QoMEX), 2016 Eighth International Conference on*. IEEE, 2016, pp. 1–6.

[94] N. J. Tustison, B. B. Avants, P. A. Cook, Y. Zheng, A. Egan, P. A. Yushkevich, and J. C. Gee, "N4ITK: improved N3 bias correction," *IEEE Transactions on Medical Imaging*, vol. 29, no. 6, p. 1310, 2010.

[95] D. S. Marcus, T. H. Wang, J. Parker, J. G. Csernansky, J. C. Morris, and R. L. Buckner, "Open access series of imaging studies (oasis): cross-sectional MRI data in young, middle aged, nondemented, and demented older adults," *Journal of Cognitive Neuroscience*, vol. 19, no. 9, pp. 1498–1507, 2007.

[96] T. Rohlfing, R. Brandt, R. Menzel, and C. R. Maurer Jr, "Segmentation of three-dimensional images using non-rigid registration: Methods and validation with application to confocal microscopy images of bee brains," in *Medical Imaging 2003: Image Processing*, vol. 5032. International Society for Optics and Photonics, 2003, pp. 363–374.

[97] T. Yousaf, G. Dervenoulas, and M. Politis, "Advances in MRI methodology." *International review of neurobiology*, vol. 141, pp. 31–76, 2018.

[98] 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, pp. 946–953, 2014.

[99] D. C. Alexander, D. Zikic, J. Zhang, H. Zhang, and A. Criminisi, "Image quality transfer via random forest regression: applications in diffusion MRI," in *International Conference on Medical Image Computing and Computer-assisted Intervention*. Springer, 2014, pp. 225–232.

[100] D. Mahapatra, B. Bozorgtabar, S. Hewavitharanage, and R. Garnavi, "Image super resolution using generative adversarial networks and local saliency maps for retinal image analysis," in *International Conference on Medical Image Computing and Computer-assisted Intervention*. Springer, 2017, pp. 382–390.

[101] 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 *15th International Symposium on Biomedical Imaging*. IEEE, 2018.

[102] A. Jog, A. Carass, and J. L. Prince, "Self super-resolution for magnetic resonance images," in *International Conference on Medical Image Computing and Computer-assisted Intervention*. Springer, 2016, pp. 553–560.

[103] R. Timofte, V. D. Smet, and L. V. Gool, "Anchored neighborhood regression for fast example-based super-resolution," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1920–1927.

[104] M. Delbracio and G. Sapiro, "Hand-held video deblurring via efficient fourier aggregation," *IEEE Transactions on Computational Imaging*, vol. 1, no. 4, pp. 270–283, 2015.

[105] M. Weigert, U. Schmidt, T. Boothe, A. Müller, A. Dibrov, A. Jain, B. Wilhelm, D. Schmidt, C. Broaddus, S. Culley, M. Rocha-Martins, F. Segovia-Miranda, C. Norden, R. Henriques, M. Zerial, M. Solimena, J. Rink,

P. Tomancak, L. Royer, F. Jug, and E. W. Myers, "Content-aware image restoration: pushing the limits of fluorescence microscopy," *Nature Methods*, vol. 15, no. 12, p. 1090, 2018.

[106] M. A. Bernstein, S. B. Fain, and S. J. Riederer, "Effect of windowing and zero-filled reconstruction of MRI data on spatial resolution and acquisition strategy," *Journal of Magnetic Resonance Imaging: An Official Journal of the International Society for Magnetic Resonance in Medicine*, vol. 14, no. 3, pp. 270–280, 2001.

[107] Y. Blau, R. Mechrez, R. Timofte, T. Michaeli, and L. Zelnik-Manor, "2018 PIRM challenge on perceptual image super-resolution," *arXiv preprint arXiv:1809.07517*, 2018.

[108] H. Zhao, O. Gallo, I. Frosio, and J. Kautz, "Loss functions for image restoration with neural networks," *IEEE Transactions on Computational Imaging*, vol. 3, no. 1, pp. 47–57, 2016.

[109] W. Luo, Y. Li, R. Urtasun, and R. Zemel, "Understanding the effective receptive field in deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2016, pp. 4898–4906.

[110] A. Jog and B. Fischl, "Pulse sequence resilient fast brain segmentation," in *International Conference on Medical Image Computing and Computer-assisted Intervention*. Springer, 2018, pp. 654–662.

[111] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli *et al.*, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[112] C. T. Vu and D. M. Chandler, "S3: A spectral and spatial sharpness measure," in *Advances in Multimedia, 2009. MMEDIA'09. First International Conference on*. IEEE, 2009, pp. 37–43.

[113] A. Mittal, R. Soundararajan, and A. C. Bovik, "Making a "completely blind" image quality analyzer," *IEEE Signal Processing Letters*, vol. 20, no. 3, pp. 209–212, 2013.

[114] A. Mittal, A. K. Moorthy, and A. C. Bovik, "No-reference image quality assessment in the spatial domain," *IEEE Transactions on Image Processing*, vol. 21, no. 12, pp. 4695–4708, 2012.

[115] O. Bruder, A. Wagner, C. J. Jensen, S. Schneider, P. Ong, E.-M. Kispert, K. Nassenstein, T. Schlosser, G. V. Sabin, U. Sechtem, and H. Mahrholdt, "Myocardial scar visualized by cardiovascular magnetic resonance imaging predicts major adverse events in patients with hypertrophic cardiomyopathy," *Journal of the American College of Cardiology*, vol. 56, no. 11, pp. 875–887, 2010.

[116] S. Yamada, K. Tsuchiya, W. Bradley, M. Law, M. Winkler, M. Borzage, M. Miyazaki, E. Kelly, and J. McComb, "Current and emerging MR imaging techniques for the diagnosis and management of CSF flow disorders: a review of phase-contrast and time–spatial labeling inversion pulse," *American Journal of Neuroradiology*, vol. 36, no. 4, pp. 623–630, 2015.

[117] C. Zhao, B. E. Dewey, D. L. Pham, P. A. Calabresi, D. S. Reich, and J. L. Prince, "SMORE: A Self-supervised Anti-aliasing and Super-resolution

Algorithm for MRI Using Deep Learning," *IEEE Transactions on Medical Imaging*, 2020.

[118] B. Avants, N. Tustison, and H. J. Johnson, "Advanced normalization tools," http://stnava.github.io/ANTs/, 2011.

[119] B. B. Avants, C. L. Epstein, M. Grossman, and J. C. Gee, "Symmetric diffeomorphic image registration with cross-correlation: evaluating automated labeling of elderly and neurodegenerative brain," *Medical Image Analysis*, vol. 12, no. 1, pp. 26–41, 2008.

[120] M. Shao, S. Han, A. Carass, X. Li, A. M. Blitz, J. L. Prince, and L. M. Ellingsen, "Shortcomings of Ventricle Segmentation Using Deep Convolutional Networks," in *Understanding and Interpreting Machine Learning in Medical Image Computing Applications*. Springer, 2018, pp. 79–86.

[121] V. S. Fonov, A. C. Evans, R. C. McKinstry, C. Almli, and D. Collins, "Unbiased nonlinear average age-appropriate brain templates from birth to adulthood," *NeuroImage*, no. 47, p. S102, 2009.

[122] S. Roy, J. A. Butman, D. L. Pham, and Alzheimers Disease Neuroimaging Initiative, "Robust skull stripping using multiple MR image contrasts insensitive to pathology," *NeuroImage*, vol. 146, pp. 132–147, 2017.

[123] L. R. Dice, "Measures of the amount of ecologic association between species," *Ecology*, vol. 26, no. 3, pp. 297–302, 1945.

[124] Y. Bei, A. Damian, S. Hu, S. Menon, N. Ravi, and C. Rudin, "New techniques for preserving global structure and denoising with low information loss in single-image super-resolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 874–881.

[125] F. Helmchen and W. Denk, "Deep tissue two-photon microscopy," *Nature methods*, vol. 2, no. 12, p. 932, 2005.

[126] Y. Zhang, Y. Zhu, E. Nichols, Q. Wang, S. Zhang, C. Smith, and S. Howard, "A poisson-gaussian denoising dataset with real fluorescence microscopy images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 11 710–11 718.

[127] S. Yoon, M. Kim, M. Jang, Y. Choi, W. Choi, S. Kang, and W. Choi, "Deep optical imaging within complex scattering media," *Nature Reviews Physics*, pp. 1–18, 2020.

[128] T. Ragan, L. R. Kadiri, K. U. Venkataraju, K. Bahlmann, J. Sutin, J. Taranda, I. Arganda-Carreras, Y. Kim, H. S. Seung, and P. Osten, "Serial two-photon tomography for automated ex vivo mouse brain imaging," *Nature methods*, vol. 9, no. 3, pp. 255–258, 2012.

[129] S. Roy, W.-T. Wang, A. Carass, J. L. Prince, J. A. Butman, and D. L. Pham, "PET attenuation correction using synthetic CT from ultrashort echo-time MR imaging," *Journal of Nuclear Medicine*, vol. 55, no. 12, pp. 2071–2077, 2014.

[130] N. Burgos, M. J. Cardoso, F. Guerreiro, C. Veiga, M. Modat, J. McClelland, A.-C. Knopf, S. Punwani, D. Atkinson, S. R. Arridge *et al.*, "Robust CT synthesis for radiotherapy planning: application to the head and neck region," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2015, pp. 476–484.

[131] C. Zhao, A. Carass, J. Lee, Y. He, and J. L. Prince, "Whole brain segmentation and labeling from CT using synthetic MR images," in *International Workshop on Machine Learning in Medical Imaging*. Springer, 2017, pp. 291–298.

[132] M. Chen, A. Carass, A. Jog, J. Lee, S. Roy, and J. L. Prince, "Cross contrast multi-channel image registration using image synthesis for MR brain images," *Medical image analysis*, vol. 36, pp. 2–14, 2017.

[133] J. E. Iglesias, E. Konukoglu, D. Zikic, B. Glocker, K. Van Leemput, and B. Fischl, "Is synthesizing MRI contrast useful for inter-modality analysis?" in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2013, pp. 631–638.

[134] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2274–2282, 2012.

[135] D. L. Pham, C. Xu, and J. L. Prince, "Current methods in medical image segmentation," *Annual review of biomedical engineering*, vol. 2, no. 1, pp. 315–337, 2000.

[136] W. Bai, W. Shi, C. Ledig, and D. Rueckert, "Multi-atlas segmentation with augmented features for cardiac MR images," *Medical image analysis*, vol. 19, no. 1, pp. 98–109, 2015.

[137] N. Komodakis, "Image completion using global optimization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1. IEEE, 2006, pp. 442–452.

[138] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 23, no. 11, pp. 1222–1239, 2001.

[139] J. Lee, A. Carass, A. Jog, C. Zhao, and J. L. Prince, "Multi-atlas-based CT synthesis from conventional MRI with patch-based refinement for MRI-based radiotherapy planning," in *Medical Imaging 2017: Image Processing*, vol. 10133. International Society for Optics and Photonics, 2017, p. 101331I.

[140] G. K. Rohde, A. Aldroubi, and B. M. Dawant, "The Adaptive Bases Algorithm for intensity based nonrigid Image Registration," *IEEE Transactions on Medical Imaging*, vol. 22, no. 11, pp. 1470–1479, 2003.

[141] G. P. Penney, J. M. Blackall, M. S. Hamady, T. Sabharwal, A. Adam, and D. J. Hawkes, "Registration of freehand 3D ultrasound and magnetic resonance liver images," *Medical Image Analysis*, vol. 8, no. 1, pp. 81–91, 2004.

[142] B. B. Avants, C. L. Epstein, M. Grossman, and J. C. Gee, "Symmetric diffeomorphic image registration with cross-correlation: evaluating

automated labeling of elderly and neurodegenerative brain," *Medical Image Analysis*, vol. 12, no. 1, pp. 26–41, 2008.

[143] M. P. Heinrich, M. Jenkinson, M. Bhushan, T. Matin, F. V. Gleeson, M. Brady, and J. A. Schnabel, "MIND: Modality independent neighbourhood descriptor for multi-modal deformable registration," *Medical Image Analysis*, vol. 16, no. 7, pp. 1423–1435, 2012.

[144] C. Wachinger and N. Navab, "Entropy and laplacian images: Structural representations for multi-modal registration," *Medical Image Analysis*, vol. 16, no. 1, pp. 1–17, 2012.

[145] W. R. Crum, T. Hartkens, and D. L. G. Hill, "Non-rigid image registration: theory and practice," *The British Journal of Radiology*, vol. 77, no. S2, 2014.

[146] A. Sotiras, C. Davatzikos, and N. Paragios, "Deformable medical image registration: A survey," *IEEE Transactions on Medical Imaging*, vol. 32, no. 7, pp. 1153–1190, 2013.

[147] M. Chen, A. Jog, A. Carass, and J. L. Prince, "Using image synthesis for multi-channel registration of different image modalities," in *SPIE Medical Imaging*. International Society for Optics and Photonics, 2015, pp. 94 131Q–94 131Q.

[148] M. Chen, A. Lang, H. S. Ying, P. A. Calabresi, J. L. Prince, and A. Carass, "Analysis of macular oct images using deformable registration," *Biomedical optics express*, vol. 5, no. 7, pp. 2196–2214, 2014.

[149] M. Bilgel, A. Carass, S. M. Resnick, D. F. Wong, and J. L. Prince, "Deformation field correction for spatial normalization of PET images," *Neuroimage*, vol. 119, pp. 152–163, 2015.

[150] A. Rosenfeld, *Multiresolution image processing and analysis*. Springer Science & Business Media, 2013, vol. 12.

[151] J. Sun, Z. Xu, and H.-Y. Shum, "Image super-resolution using gradient profile prior," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2008, pp. 1–8.

[152] W. M. Wells III, P. Viola, H. Atsumi, S. Nakajima, and R. Kikinis, "Multi-modal volume registration by maximization of mutual information," *Medical Image Analysis*, vol. 1, no. 1, pp. 35–51, 1996.

[153] P. Viola and W. M. Wells III, "Alignment by maximization of mutual information," *International Journal of Computer Vision*, vol. 24, no. 2, pp. 137–154, 1997.

[154] C. Studholme, D. L. Hill, and D. J. Hawkes, "An overlap invariant entropy measure of 3D medical image alignment," *Pattern recognition*, vol. 32, no. 1, pp. 71–86, 1999.

[155] B. A. Landman, A. J. Huang, A. Gifford, D. S. Vikram, I. A. L. Lim, J. A. Farrell, J. A. Bogovic, J. Hua, M. Chen, S. Jarso *et al.*, "Multi-parametric neuroimaging reproducibility: a 3-T resource study," *Neuroimage*, vol. 54, no. 4, pp. 2854–2866, 2011.

Vita

Can Zhao earned her Bachelor degree in Electrical Engineering from Tsinghua University, Beijing in 2013. She received her M.S.E. degree in Electrical and Computer Engineering from the Johns Hopkins University in 2017. Currently she is a research scientist in NVIDIA. She is also a PhD candidate in Electrical and Computer Engineering at the Johns Hopkins University and is advised by Dr. Jerry L. Prince. Her research focuses on image synthesis and super-resolution. Her general interests include medical image analysis, computer vision, and deep learning.