2018

Master's thesis

Three-Dimensional Super-Resolution for Magnetic Resonance Imaging using Convolutional Neural Networks

1215088 Churong ZHUO

Advisor Associate Professor. Shinichi Yoshida

Informatic course

Graduate School of Engineering, Kochi University of Technology

Abstract

Three-Dimensional Super-Resolution for Magnetic Resonance Imaging using Convolutional Neural Networks

Churong ZHUO

High resolution magnetic resonance imaging (MRI) is becoming indispensable for accurate quantitative medical diagnosis. However, it is difficult to obtain high-resolution MRI images due to the medical device and other limitation. Single image super-resolution (SISR) method can generate a high-resolution (HR) image from a single low-resolution (LR) image. Recently, SISR methods has made a major breakthrough in deep learning. In this paper, we first used two-dimensional (2D) super resolution convolutional neural network (SRCNN) to generate the HR image in MRI images, then we introduced a new neural network architecture, three-dimensional (3D) enhanced deep super-resolution network (EDSR) to generate HR images of structural brain magenetic resonance images. The proposed 3D networks show superior performance over the 2D SRCNN networks and interpolation methods. The results suggest that 3D convolutional neural network could be promising and helpful in medical imaging super-resolution.

key words Convolutional nerual network, Super-resolution, Deep learning, 3D nerual network, High-resolution medical imaging, Magnetic resonance imaging

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This chapter presents the background and significance for the research presented in this thesis. We start with the research background in section 1.1. Then we review super-resolution in section 1.2. In section 1.3, we discuss about prior work on super-resolution method, such as, interpolation-based method, reconstruction-based method and example-based method. In section 1.4, we review about deep-learning algorithms for super-resolution. In the section 1.5, we show significance of our research.

1.1 Background

Medical imageing is the result of interactions between a tissue and a physical phenomenon such as a wave (in Ultrasound, MRI) or an ionizing particle (in X-ray/CT, PET)[1]. It is becoming indispensable for medical diagnosis and other medical application, but due to medical imaging device and other limitations, it is always difficult to obtain high-resolution (HR) images. High-resolution medical images can help doctor to have clear sight of patient and make better decision about the treatment options. It is also important to have HR images in medical imaging, since it contains abundant structural details which can help to facilitate accurate diagnosis and quantitative measurements. However, due to technological and economical limitations, it is difficult to obtain HR medical images.

Super-resolution has gained increasing research attention for decades. However,

most of the researches are based on natural images instead of medical images. In this thesis, we first applied a benchmark method-super-resolution convolutional neural networks (CNN) in MRI, then we proposed a new three-dimensional CNN for MRI. We showed that CNN and the proposed method for super-resolution can be useful in medical images domain.

1.2 Super-resolution

In computer vision fields[2], high-resolution (HR) images are required for better performance in pattern recognition and analysis of images. Super-resolution[3] refers to construct a HR images from one or more obtained LR images. There are two kinds of super resolution methods, one is using multiple images to reconstruct the high resolution (HR) image (MISR), and the other is obtaining HR images from single LR images (SISR). SISR is more efficient compared with MISR. In MRI super resolution, multiple image super resolution techniques have been proposed[4, 5] because the accuracy to the true image based on the statistical estimation is important in the medical scene. However, the scan time of MRI is long, and it is difficult to obtain multiple images because the scan time is too long for patients to wait without any movement. Therefore, SISR method is a promising approach to address SISR problem.

In SISR methods, one specific LR input can correspond to many possible HR outputs, and it is usually intractable for us to map the LR input into the HR space[6]. So SISR is a notoriously challenging ill-posed problems. Till now, mainstream algorithms of SISR methods can be broadly categorized into three categories: interpolation-based methods, reconstruction-based methods and example-based methods.

1.3 Approaches for super-resolution

1.3.1 Interpolation-based methods

Interpolation method[7] is widely used in image processing, it is the process of estimating unknown values from known sample values and estimating continuous samples from discrete samples. It is intuitive and computationally effective. There are three basic interpolation methods, nearest neighbor, bilinear, and bicubic interpolation methods.

Nearest neighbor method is a local interpolator so the computing load is relatively light, it is also the simplest and fastest implementation way in image scaling. Nearest neighbor simply selects the value of the closest pixel to which the interpolation resampling maps. And it have the worst results since magnified image have mosaic effect and the zoom out results isn't quite true to the original input.

Bilinear methods is a resampling method that uses the distance 足 weighted average of the four nearest pixel values to estimate a new pixel value. The four cell centers from the input raster are closest to the cell center for the output processing cell will be weighted and based on distance and then averaged.

Bicubic method is a bit difficult to use. Bicubic method considers the closest 4x4 neighborhood of known pixels. Since these are at various distances from the unknown pixel, closer pixels are given a higher weighting in the calculation. Bicubic produces noticeably sharper images than the nearest neighbor method.

Interpolation methods is the simplest and fastest way to construct HR images, but it often generate over-smoothed images which loss the details of fine edges and introduce additional artifacts. In our thesis, we use two interpolation methods to construct HR images and then compare with our proposed methods.

1.3.2 Reconstruction-based methods

Reconstruction based super-resolution methods[8, 9, 10, 11] often adopt sophisticated prior knowledge to restrict the possible solution space with an advantage of generating flexible and sharp details.

However, the choice of magnification factors would affects the reconstruction results[12]. And researcher also found that with the increment of the magnification factor and reconstruction-based methods can only generate an overly smoothed output[13]. In general, the reconstruction performance degrades rapidly when the scale factor increases, and these methods are also time-consuming[14].

1.3.3 Example-based methods

Example-based methods[15] are also known as learning-based methods,it use the image database or the image itself to the relationship between LR and HR image pairs. And use this relationship as a prior constraint to generate HR images. It obtains information from a large number of training sets, it can achieve better results than reconstruction-based methods when enlarging amplification factors.

Freeman et al.[16] first proposed Markov Random Field (MRF) in super-resolution domain. Compared with reconstruction based methods, it could obtain more abundant high frequency details to construct HR images under the condition of magnification 4 times. Chang et al.[17] proposed neighbor embedding methods by assuming the similar local geometrical feature space between LR and HR images to get high quality reconstructed images. It use less training samples and have low sensitivity to noise compared to MRF methods, but it is difficult to choose the block size of neighborhood. Inspired by the sparse representation theory[18] in signal recovery, many researchers[19, 20, 21] proposed sparse presentation methods in SISR domains. Sparse representation do not

need to select the neighborhood size, its shortcoming is how to choose a over complete dictionary.

1.4 Deep learning methods for super-resolution

In the past few years, example-based methods solution based on deep learning methods are playing a vital part to improve the image quality of magnified images in many computer vision tasks. Deep learning is an algorithms that directly learn diverse representations of data[22]. Influenced by the success of deep learning method which applied in computer vision field, Dong et al. proposed super-resolution convolutional neural network (SRCNN)[23, 24], which is the SISR using only 3-layers deep neural network to generate HR image, and it demonstrated high quality image reconstruction by machine learning in natural images. We will explicit more details of SRCNN in chapter 3. In medical image field, researchers have discovered that the SRCNN application for chest radiographs[25] and CT images[26] could be greatly improved medical image quality compared with conventional linear interpolation method. However, the reconstruction results of SRCNN is sensitive to little changes of the structure. When using different initialization and training method, it will obtain different performance even in same model. Therefore, it is important to have a well-designed CNN architecture and optimization methods when training the neural networks.

Recently, residual neural network (ResNet)[27] gains researcher's attention because of its promising approach for image reconstruction. And different kinds of architecture that try to obtain HR images from LR images using ResNet have been published. Unfortunately, there are limitation about the aforementioned deep learning methods. First, the previous approach of deep learning is proposed for 2D natural images, but many medical images are 3D volumes. Second, directly convert 2D deep-learning net-

works into 3D networks might result in enormous number of network parameters and thus suffer problems in memory allocation. Finally, the architecture could be further improved. Hence, 3D deep-learning networks for MRI images is desirable.

1.5 Significance

In this paper, we first applied and evaluated the application which using 2D SRCNN to MRI in order to obtain high quality medical images. The reconstruction results of SRCNN showed that it achieved great results compared with interpolation methods. However, 2D SRCNN seems to generate over-smoothed HR MRI in visual effects. Then we proposed an architecture for MRI that could using 3D convolution to process the volumetric information contained in MRI scans, and taking advantage of ResNet to improved the quality of reconstructed images. The model is based on enhanced deep super-resolution network (EDSR)[28]. We use perceptual loss based on the first three convolution layers of VGG16[29] instead of mean square error (MSE) loss in order to improve the quality of reconstructed images. We evaluate our proposed 3D EDSR model by peak signal-to-noise ratio (PSNR)[30], structural similarity (SSIM)[31] and mean opinion score (MOS), the MOS showed that our network outperformed 2D SRCNN and interpolation methods.

Chapter 2

Convolutional Neural Network

In this chapter, we explicit some basic deatils of convolutional neural networks that would be used in our research.

2.1 Brief of CNNs

Machine learning is subfield of artificial intelligence (AI). Its goal is to make computers to learn and act like humans do, and improve their learning over time on their own, by feeding them data and information in the form of observations and real-world interactions. In the past decade, machine learning has been used in many fields of modern society: effective speech recognition[32], effective web search[33, 34], identify objects in images[35, 36], improve the quality of the images. The conventional machine learning methods were limited when it process raw form of the natural data. It needs lot of domain expertises to construct a traditional machine learning system, but they still need guidances. Human intervention is needed when it returns an inaccurate prediction. But with a deep learning model, it can determine on their own if a prediction is accurate or not.

Covolutional neural networks were inspired by the organization of the animal visual cortex. Research in the 1950s and 1960s by Hubel and Wiesel [37] on the brain of mammals suggested that how mammals perceive the world visually. They found out that visual cortex of cat and monkey include neurons that exclusively respond to neurons.

In 1984, inspired by the concept of receptive field, Fukushima proposed a a hierarchical neural network model[38] which was called the neocognitron. Later, Le cun et at. introduced a convolutional neural networks which was called LeNet-5[39]. LeNet-5 was able to classify hand-written numbers.

2.2 Structure of CNNs

CNNs have a different architecture than regular neural networks[40]. Regular neural networks take all the informations in original images since each neuron is fully connected to all neurons in the previous layer. It causes huge numbers of the parameters which is wasteful and may quickly lead to overfitting. Unlike regular neural networks, CNNs can reduce the original images into a single vector.

A CNN basically consists convolutional layer, pooling layer and fully connected layer[41]. First, the original input data will be transformed into raw pixel data. Next, convolutional layer will obtain the feature map by using different feature detectors (filters). Then, rectified linear unit layer (ReLU)[42] is used to increase the non-linearity in the images. Afterward, pooling layer will perform a down-sampling operation in the feature map. Finally, fully connected layer will combine all the features that is gained by previous layers into a wider variety of attributes. Figure 2.1 shows a simple architecture of CNN.

2.3 Convolutional layer

Convolutional layer is one of the important layer in the CNNs. The term convolution refers to the mathematical combination of two functions to produce a third function. It merges two sets of information. In the case of CNNs, the convolution refers to using a edge detectors (filter or kernel) to produce extract features from an input

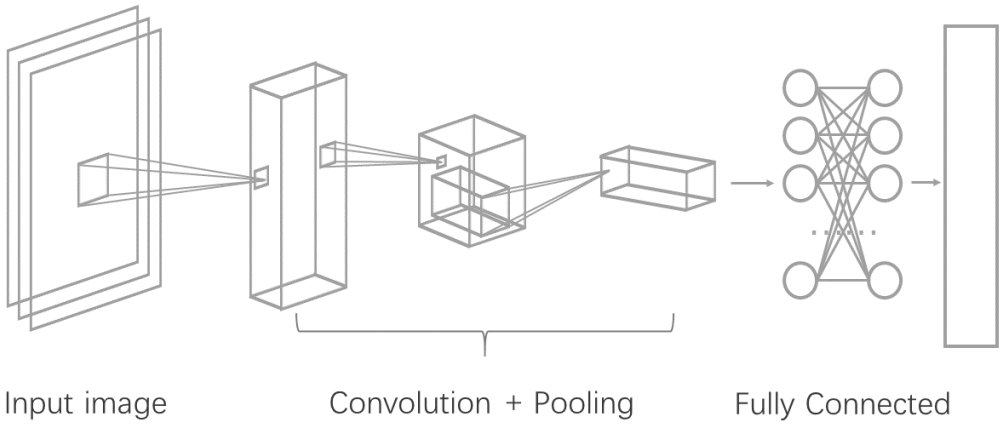


Fig. 2.1: A simple CNN

image and produce feature map. Convolution preserves the relationship between pixels by learning image features using small squares of input data.

2.3.1 Convolutional operator

In order to generate a feature map, an array of weights are taken and slided it over the image, then taking the dot product between the weights and small area of the pixels to generate a feature map. Stride is the number of pixels slide over the input weights.

As the figure 2.2 shows below, the input image has been defined as I_{ijk}, the filter array has been defined as W_{ijk}, and the feature maps is defined as O_{ijk}. The convolution computation is formally defined as follow:

$$O_{ij} = \sum_{k=1}^{N} I_{ijk} * W_{ijk} + bias \tag{2.1}$$

There are three feature maps after the convolutional computation since the picture has 3 channels, then combine all three feature maps into one feature maps.

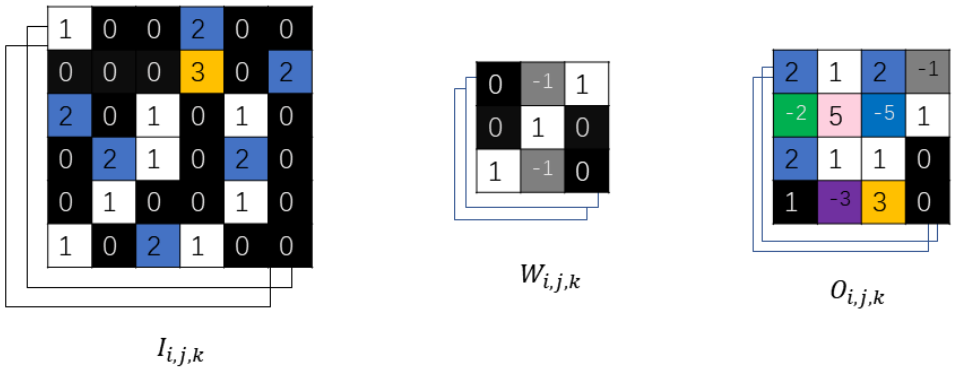


Fig. 2.2: Convolution computation

2.3.2 Padding

There are two problems when we apply convolutional operator. First, every time we apply a convolutional operator, the input images would become smaller and smaller. As figure 2.2 shows, the 9x9 image shrink into 4x4 image. Second, the pixel at the corner or the edge is used only once in the computation, but the pixel in the middle is used multiple times in the computation. The pixels on the corners are used much less in the outputs, that means we throw away a lot of information near the edge of the image.

In order to solve these problems, we pad the image with an additional border of one pixel all around the edges. As figure 2.3 shown, after padding an additional border, the original input size become 8x8, and we manage to preserve the original input size of six by six.

2.3.3 Activation function

Convolution is a linear operation-element wise matrix multiplication and addition, adding a activation layer can help to transform the input of the neurons. There are many activation method, such as sigmoid function, tanh function and ReLU function.

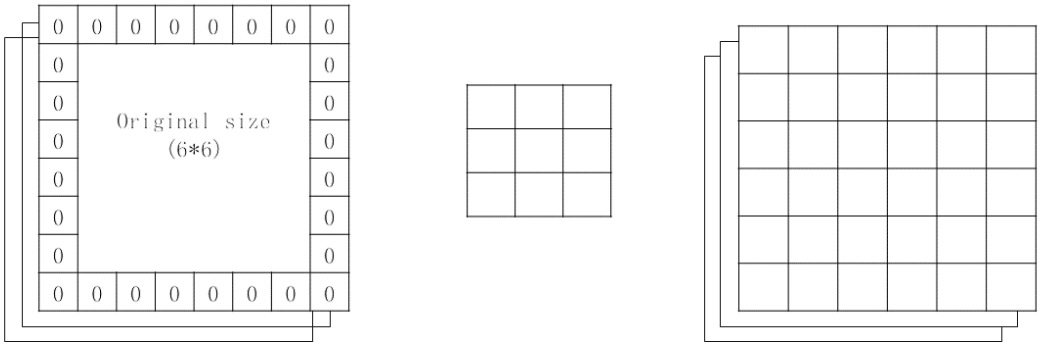


Fig. 2.3: Padding

Recently, ReLU becomes popular in CNNs since it is easier to calculate and can reduce the gradient vanishing problem. ReLU can be expressed like: f(x)=max(0,x). It will replace all the negative value by zero. ReLU is shown in figure 2.4.

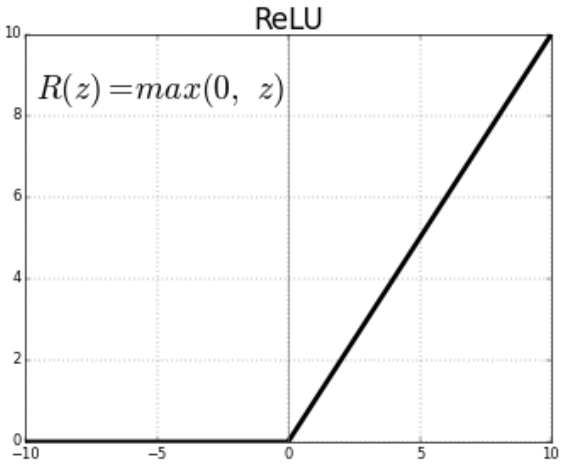


Fig. 2.4: Rectified Linear Unit

2.4 Pooling layer

Pooling layer can reduce the spatial amount of the presentation, dispose of unnecessary information or features and lessening the computation cost. Preventing overfit-

ting coming. There are two different function called max pooling[43], average pooling. However, researchers are mainly focus on max pooling layer because it extract the important features (the biggest pixel) of the input volume. Figure 2.5 shows max pooling and average pooling methods with 2x2 filters and stride of 2 (frequently used setting). Max pooling applied to the input feature map and output the maximum pixel in every subregion that filter slide around. Average pooling calculate the average pixel in the subregion that filter slide around. As we can see in the figure 2.5, a 4x4 features map becomes a 2x2 pooled feature map, but average pooling sometimes cannot extract good features.

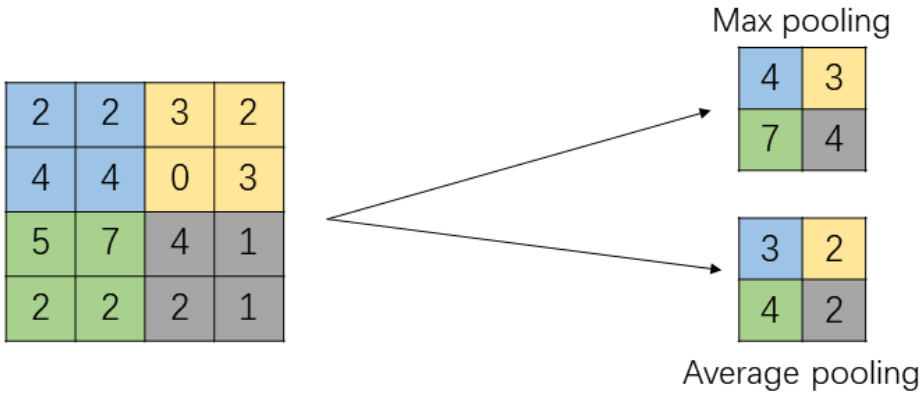


Fig. 2.5: Example of max pooling

2.5 Fully connected layer

After multiple convolutional layers and pooling layers, there are one or two fully connected layers to give the final classification results. Fully connected layer connect every neuron in the previous layer to every neuron in the next layer like regular neural networks. The fully connected layer in the CNN represents the feature vector for the input. The feature vector holds information that can present the input. During training,

the feature vector is being used to determine the loss, and help the network to get train.

The convolutional layers before fully connected layer extracts the pixel regarding local features (edges, shapes, etc.) in the input images. Each convolutional layer using serveral filters to extract the different local features. The fully connected layer composite and aggregated all features from all the previous convolutional layers then help to classify the results.

Chapter 3

Methods of Super-Resolution

In this chapter, we introduce CNNs method for super-resolution. Then we introduce our details of SRCNN configuration. Finally, we explain our proposed 3D-EDSR model configuration.

3.1 Super-Resolution Neural Network

With the development and success of deep learning method applied in computer vision fields, Dong et al. introduced super-resolution convolutional neural network (SRCNN). It uses only 3 layers network to generate HR images, and it demonstrates high quality image reconstruction in natural images. The SRCNN scheme is a feed-forward network which can be divided into three steps, patch extraction and representation, non-linear mapping, and reconstruction. The patch reconstruction step extracts patches from the LR image Y and represents each patch as a HR vector. The output of this steps is all the features of the input images which is expressed as $F_1(Y)$. The first step is expressed as follows:

$$F_1(Y) = \max(0, W_1 * Y + B_1) \tag{3.1}$$

Here, Y is the input image, * represents convolution operation, W_1 and B_1 represent the filters and biases respectively. W_1 has a size c $\times$ f_1 $\times$ f_1 $\times$ n_1, where c is the number of channels in the input image, f_1 is the spatial size of a filter, and n_1 is the number of filters. The output is composed of n_1 feature maps. B_1 is an n_1 -dimensional

vector. The feature maps obtained by convolution is processed by the activation function called ReLU. The next step is non-linear mapping step, the n_1-dimensional vectors are mapped non-linearly to another set of n_2-dimensional feature vectors. Each mapped vector represents a HR pixel block, and these vectors form another feature map set F_2(Y). The non-linear mapping operation is expressed as follows:

$$F_1(Y) = \max(0, W_2 * F_1(Y) + B_2) \tag{3.2}$$

Here, W_2 has a size of $n_1 \times f_1 \times f_1 \times n_2$. If the second convolution layer contains n_2 convolution kernels, after the convolution operation the algorithm will generate n_2-dimensional feature map. The output of this each vector of n_2-dimension represents a high- resolution pixel block. The last step is reconstruction step, in this step the algorithm aggregates the above HR representation to generate the final HR image. The operation of the last step is as follows:

$$F(Y) = W_3 * F_2(Y) + B_3 \tag{3.3}$$

Here W_3 has a size of $n_2 \times f_3 \times f_3$, corresponds to c filters, and B_3 is a c-dimensional vector. We adapted SRCNN for magnetic resonance images. We used the Adam optimizer instead of stochastic gradient descent (SGD) optimizer that Dong used in his paper. The SGD maintains a single learning rate for all weight updates and the learning rate does not change during training. Adam combines the best properties of the AdaGrad and RMSProp algorithms to provide an optimization algorithm that can handle sparse gradients on noisy problems. We used the typical and basic SRCNN configuration that proposed by Dong et al., which $f_1 = 9$, $f_2 = 1$, $f_3 = 5$, $n_1 = 64$, $n_2 = 32$. Figure 3.1 shows the architecture of the SRCNN for MRI.

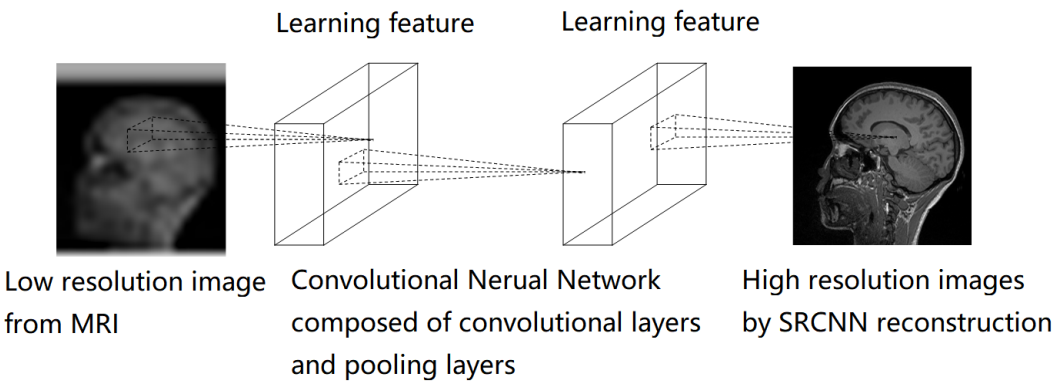


Fig. 3.1: SRCNN structure for MRI

3.2 Super-Resolution Generative Adversarial Network

3.2.1 Brief of SRGAN

In 2017, Lim, Bee, et al. proposed a SRGAN[44] based on Ian Goodfellow's Generative adversarial network (GAN) [45] to solve the super-resolution problems. GAN have two models, a generative model and a discriminative model. The task of the discriminative model is determining whether a given image looks natural or looks like it is artificially created. The generative model have the task of creating looks like natural images that are similar to the original data distribution. The analogy used in the original paper is that the generative model seems "a team of counterfeiters, trying to make and use the fake currency," in the same time, discriminative models is like "the police, trying to detect the counterfeit currency "[44]. As the model train through alternating optimization, both methods are improved until the discriminative model can't detect the counterfeit currency anymore. In SRGAN, generative model generate HR images, and discriminative model try to detect whether the images is generated by generative or the original image from the database. When discriminative model thinks the image

generated by generative model is the image from the database, then we think the HR images have been generated by the SRGAN.

3.2.2 Architecture of SRGAN

There are multiple residual block in generator network (SRResNet), each residual block contains two 3x3 convolutional layers, then batch normalization is followed by convolutional layer, and using PReLU as activation function. Two sub-pixel convolution layers are used to increase the feature size. In the discriminative network, it has 8 convolution layers, with the layers of the network getting deeper, the number of the features are increasing. Then used LeakyReLU as activation function. Finally use two fully connected layers and sigmoid activation function to achieve the the probability of predicting a natural image. Figure 3.2 shows the architecture of the SRGAN.

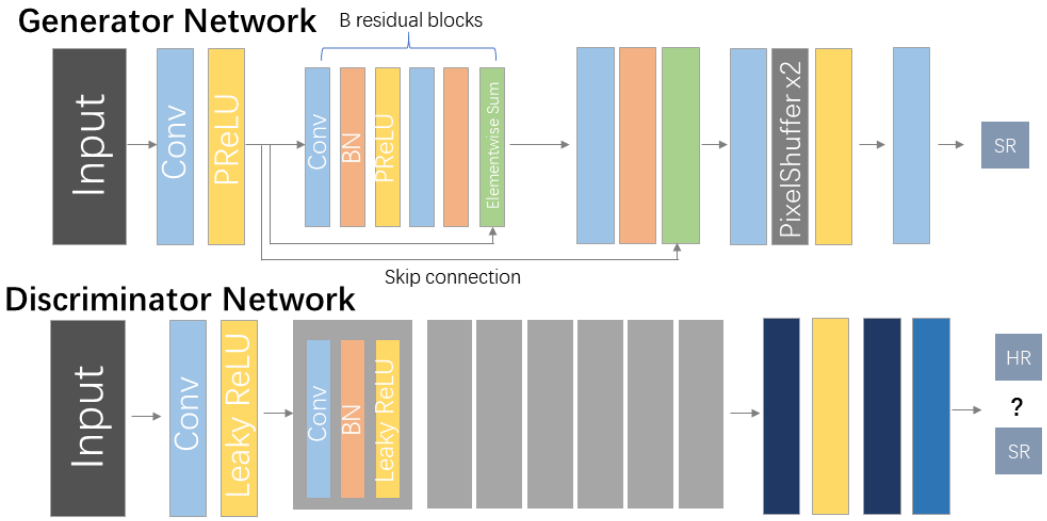


Fig. 3.2: Architecture of the SRGAN

3.2.3 Residual blocks

As more layers using certain activation functions are added to neural networks, the gradients of the loss function approaches zero, making the network difficult to train. This problem was solved by the batch normalization [46] Szegedy proposed. However, degradation problem caused by complex networks still exists. As the figure 3.2 shows, the generator network have multiple residual blocks. Residual blocks are the fundamental building block of residual networks [27] which is one of the solution for vanishing gradient problems. Figure 3.3 shows the residual block. Residual block not only have a directly convolutional output, but also have a branch which connect the input to the output. As the figure 3.3 shows, the residual connections directly adds the value at the beginning of the block, x, to the end of the block H(x)=F(x)+x.

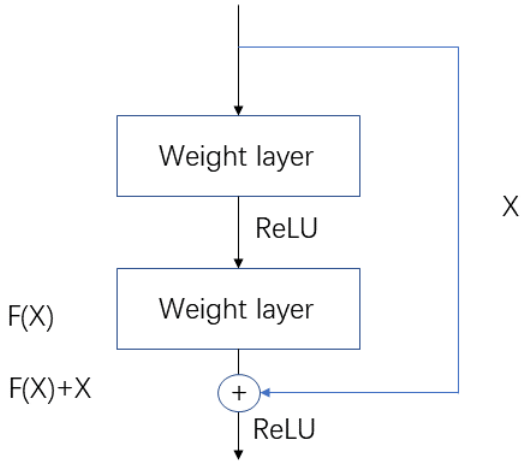


Fig. 3.3: Residual block

3.2.4 Perceptual loss

A new loss function - perceptual loss was proposed in the paper, this function enable the network to recover realistic textures and fine grained details from images. The loss

function have two parts, the content loss and the adversarial loss. The adversarial encourage images that look like the image from the database (more natural), and the content loss makes sure the generated image has similar features with the original LR images. Content loss is basically a Euclidean distance loss between the feature maps (pretrained VGG networks) of the new reconstructed image and actual HR training image. SRGAN uses a perceptual loss measuring the MSE of features extracted by VGG-19 networks:

$$l^{SR}_{VGG/i,j} = \frac{1}{W_{i,j}H_{i,j}} \sum_{x=1}^{W_{i,j}} \sum_{y=1}^{H_{i,j}} (\phi_{i,j}(I^H R)_{x,y} - \phi_{i,j}(G_{\theta G}(I^{LR}))_{x,y})^2 \tag{3.4}$$

3.3 Enhanced deep residual networks

Enhanced deep residual networks (EDSR)[28] is started from SRResNet and optimized it for higher accuracy. EDSR removed the batch-normalization of the residual blocks, because the input and output share similar distribution in super-resolution problem. It employs 32 residual blocks with 256 channels and pixel-wise L_1 loss instead of L_2. Figure 3.4 shows different kinds of residual block. Inspired by the EDSR, we propose

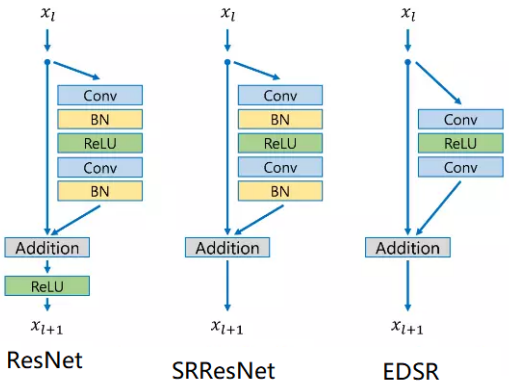


Fig. 3.4: Residual block

a three-dimensional ResNet model for Super-Resolution(3D-EDSR). The network gen-

erally followed the EDSR model. In our architecture, to be able to deal with volumetric information we use 3D convolutional layers. 3D convolutional kernels are necessary to take full advantage of the structure of the input data. We use eight residual blocks composed of a convolution with 64 filters of size 3x3x3, ReLu activation function and another convolution layers with the same parameters before. The original version of EDSR is using MSE loss, inspired by the SRGAN, we use perceptual loss based on the first three convolutional layers of VGG16. Perceptual loss compared the features from VGG16 and real features from the generated image, making the reconstructed images more similar. VGG Net is one of the most influential networks. VGG16 contains 13 convolution layers and 3 full-connected layers. The weight configuration of the VGGNet is publicly available and has been used in many other applications and challenges as a baseline feature extractor. Figure 3.5 shows our proposed network.

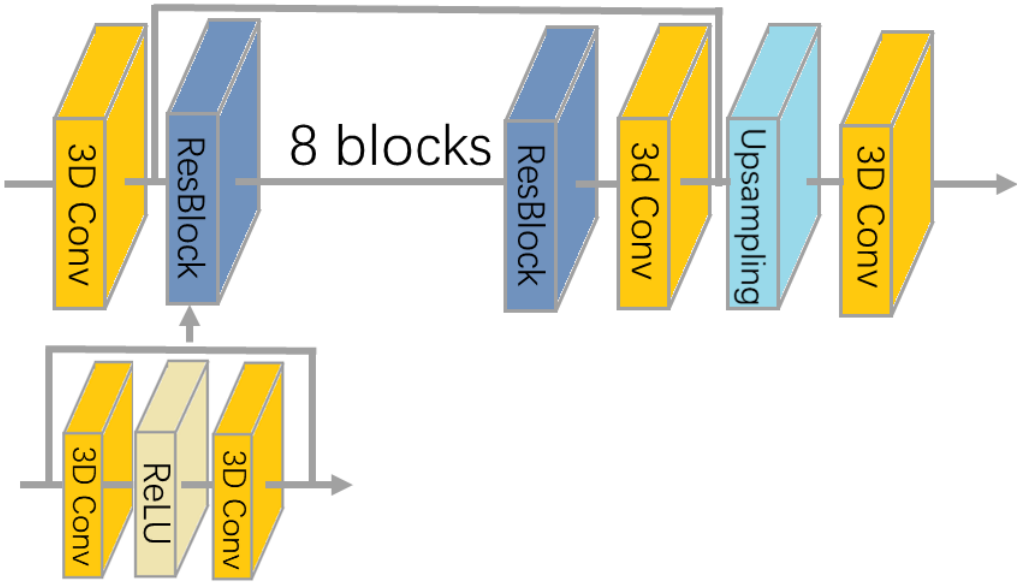


Fig. 3.5: Proposed network

Figure 3.6 shows the development from GAN to our proposed networks. In SRGAN, authors used residual blocks to build a generative model which can be called SRResNet.

3.3 Enhanced deep residual networks

Inspired by the SRResNet model, EDSR had been introduced which also used residual block but changed the architecture of residual block as figure 4.4 shows. Our proposed network is based on EDSR but adapted it for volumetric data.

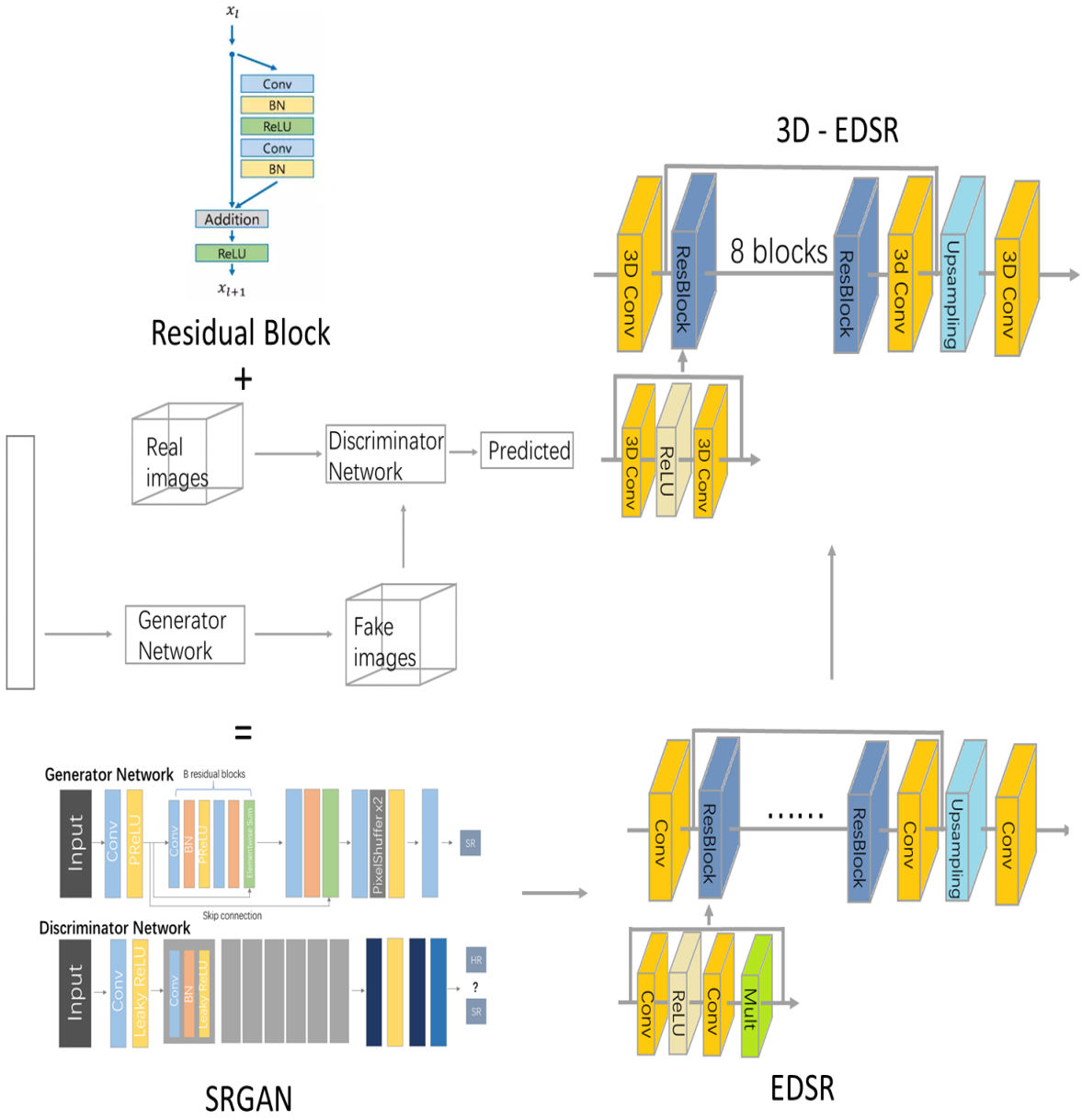


Fig. 3.6: From GAN to EDSR

Chapter 4

Experiments and Evaluation methods

In this chapter, we introduce our experiment details and our evaluation methods. First, we talk about the hardware and software environment where our experiment conducted. Then, we explain how we do the experiment by using SRCNN and 3D-EDSR. Finally we explain the details of evaluation methods.

4.1 Experiment environment

The 2D-SRCNN model is implemented in Keras (Tensorflow backend). Keras is a high-level neural networks API, written in Python and capable of running on top of TensorFlow, CNTK or Theano. It is widely used in the industry and research community, helping researchers start from scratch and turn the model into product.

3D-EDSR model is implemented in FastAI. FastAI simplifies training fast and accurate neural networks using modren best practise. FastAI is not only an educational resource, it also can be used in cutting-edge research and have achieved state of the art results.

They were performed in different hardware and software environment. Table 4.1 shows the experiment environments.

Table 4.1: Experiment Environment

/	SRCNN	3D-EDSR
CPU	Intel(R)Core(TM)i7	Intel(R) Xeon(R) CPU E5-2620 v4
GPU	GeForce GTX 1080 Ti	GeForce GTX 1080 Ti
Memory	32G	32G
OS	Ubuntu 16.04 LTS	Ubuntu 16.04 LTS
Deep Learning Frameworks	FastAI	Keras, Tensorflow

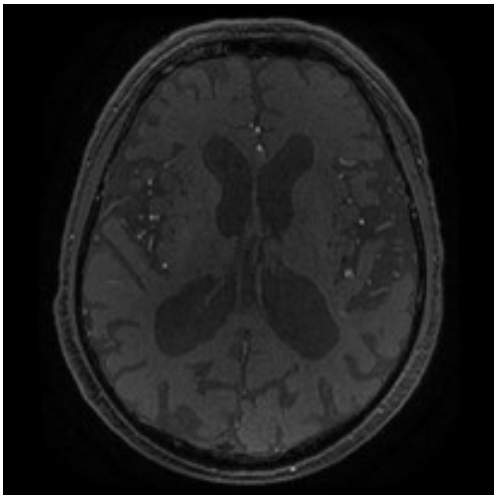
4.2 SRCNN

We use the MRI image provided by the hospital. We have 280 MRI dicom images, we randomly selected 224 as training set, 56 as test set.

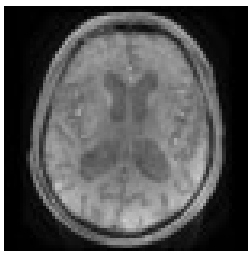
4.2.1 Preprocess

First we transformed the dicom data into three channels png files which size is 512 x 512. Next we downsampled it into 256 x 256. Then we use gaussian function to blur the image. Finally we upsample the image to original size 512 x 512. We called this image low-resolution image. Figure 4.1 shows the steps we descried before.

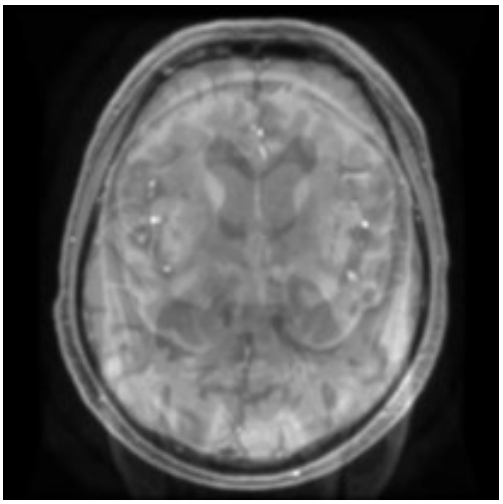
We use stride equals 14 to generate patches which size is 33x33, we can get 61,000 sub images, and we called it X, which is the high resolution image (Label). Figure 4.2 shows some of the patches the networks generates. Then we use the exact same process of patching in the low-resolutionimages. Finally we put this low-resolution patches into training phase and called it image Y.



(a) Orignal image



(b) Downsample image



(c) Upsample blur image

Fig. 4.1: Preprocessing

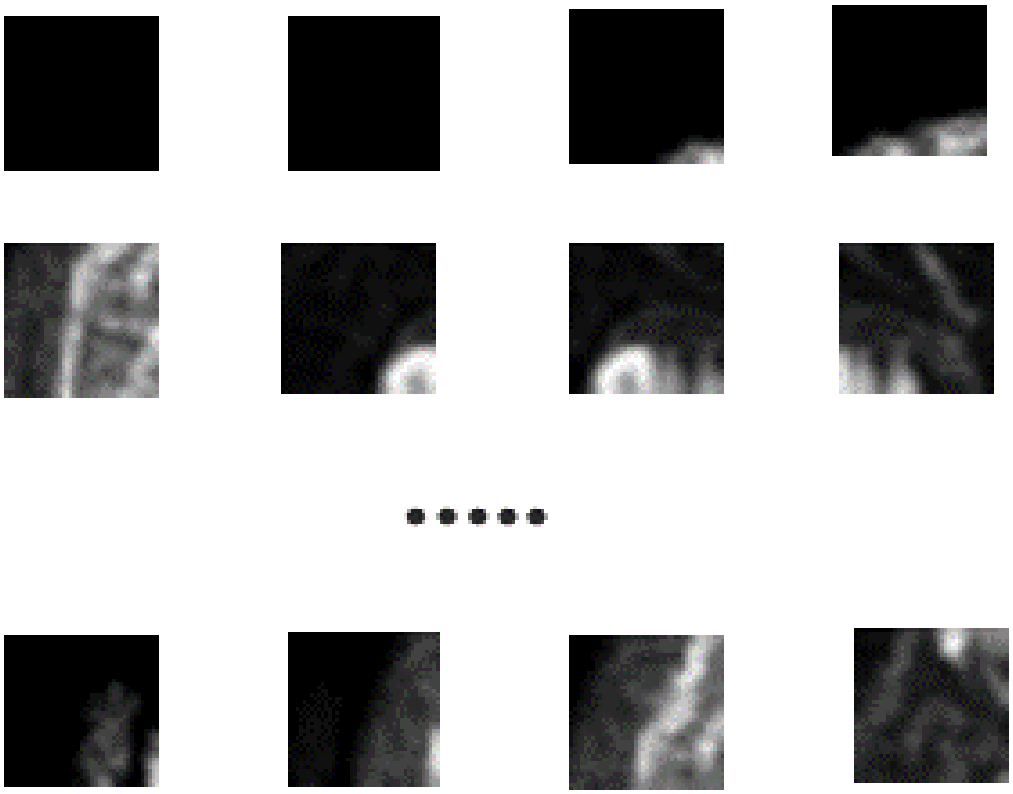


Fig. 4.2: Generated patches

4.2.2 Training

We first compared three different configuration of SRCNN. Table 4.2 shows different configuration we used for SRCNN. After comparing different configuration of SRCNN. We found out the typical and basic SRCNN configuration are the most useful one, which is $f_1 = 9$, $f_2 = 1$, $f_3 = 5$, $n_1 = 64$, $n_2 = 32$. In the original paper, authors only consider training in Y channel, they did bicubic upsampling to other channels in order to compared with the conventional way. And other channels would not be trained or tested. We didn't use any padding in all three layers in the training phase. Hence the output size of the SRCNN is 21*21. In order to compare the PSNR, we added padding

Table 4.2: Configuration of SRCNN

	9-1-5	9-3-5	9-5-5
layer 1	64 kernels, size 9x9	32 kernels, size 1x1	1 kernels, size 5x5
layer 2	64 kernels, size 9x9	32 kernels, size 3x3	1 kernels, size 5x5
layer 3	64 kernels, size 9x9	32 kernels, size 5x5	1 kernels, size 5x5

after every convolutional layer in the testing phase. The details of each layer is shown in table 4.3.

Table 4.3: Details of each layers

Conv1	Input	33*33*1
	Filters	9*9*64
	Output	25*25*64
Conv2	Input	25*25*64
	Filters	1*1*32
	Output	25*25*32
Conv3	Input	25*25*32
	Filters	5*5*1
	Output	21*21*1

4.3 3D-EDSR

In order to load DICOM MRI volume data, pydicom toolbox was used. The data in dicom had been read and the slices were ordered by the pydicom using method "Slice Location". The data is normalized to intensities between 0 and 1. Then blurred all three

dimension by using sigma gaussian filter. Figure 4.3 shows the ground truth image and figure 4.4 shows the blurred image by sigma gaussian filter. Then stored the data in

Fig. 4.3: Ground truth

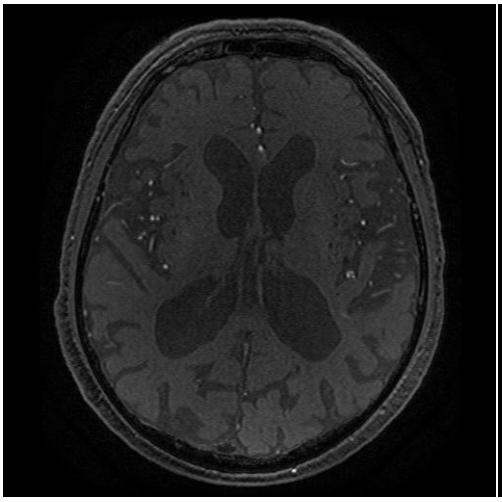
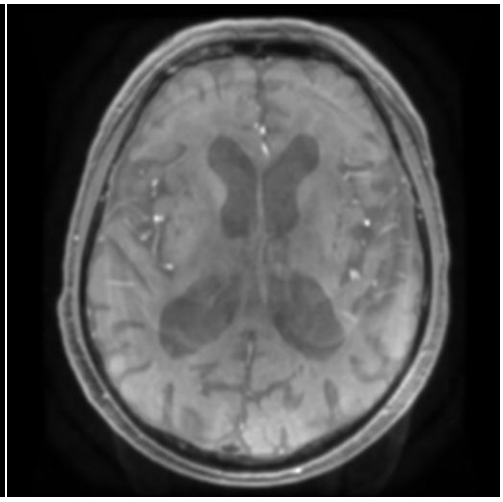


Fig. 4.4: Blurred image



32x32x32 volumes for training and save as h5 files. Figure 4.5 shows the blurred volumes and original volumes.

VGG16 is loaded in eval mode in order to get feature activations for both the HR training blocks as well as the network output for predicted HR blocks. The gradients for the super resolution network can be taken through the feature activation comparisons from the VGG network. FastAI code was written for 2D images, we expanded the perceptual loss function to calculate the feature activations for every possible 2D slice of the 32x32x32 training volume data. VGG16 expects a 3-channel RGB image, so here every 2D slice of the 32x32x32 volume data is transformed into a 3x32x32 image, and the appropriate means and stdevs are set for the RGB channels.The feature activations are calculated for each slice and the loss function is composed iteratively.

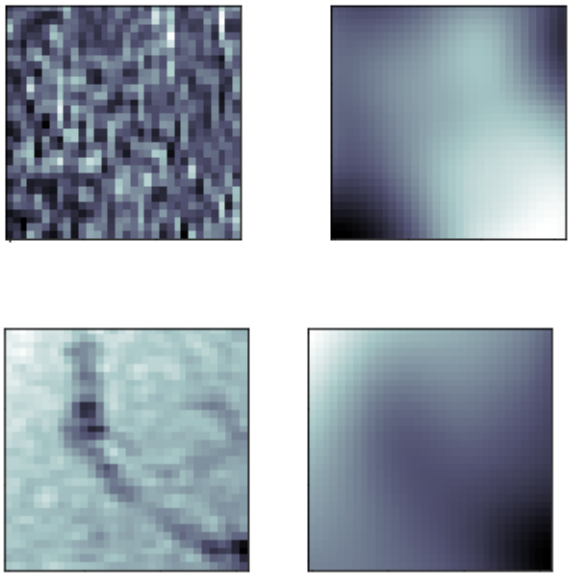


Fig. 4.5: Original volume and blurred volume

4.4 Evaluation methods

In order to evaluate the result high-resolution images quantitatively, we use peak signal-to-noise ratio (PSNR)[30] and structural similarity (SSIM)[31] to evaluate the results. The MSE represents the cumulative squared error between the compressed and the original image, whereas PSNR represents a measure of the peak error. The higher of the PSNR, the better degraded image has been reconstructed to match the original image and the better the reconstructive algorithm. PSNR is commonly used as a measure of the quality of noisy images.

The main limitation of the PSNR is that it relies strictly on numeric comparison and does not actually take into account any level of biological factors of the human vision system such as the structural similarity index.

SSIM actually measures the perceptual difference between two similar images. It cannot judge which of the two is better: that must be inferred from knowing which is the “original” and which has been subjected to additional processing such as data

compression. SSIM is designed to improve on traditional methods such as peak signal-to-noise ratio (PSNR) and mean squared error (MSE).

Accroding to the author of SRGAN, the ability of MSE (and PSNR) to capture perceptually relevant differences, such as high texture detail, is very limited as they are defined based on pixel-wise image differences [47, 48, 49]. So we also performed a Mean opinion score (MOS) test to quantify the ability of different approaches to reconstruct perceptually convincing images. Specifically, we asked 10 raters to rate from 1 (low quality) to 5 (high-quality) to the super-resolved images. The raters rated versions of each bicubic, nearest neighbor (NN), SRCNN, 3D-EDSR models and the original HR images. The raters were calibrated on the blurred images (score 1) and ground truth (score 5) versions of 10 images from the training set. We randomly select 20 images from the testing set, then presents all the images in a randomized fashion to the raters to evaluate (5 methods: bicubic, nearest neighbor, SRCNN, 3D-EDSR, Ground-truth, total 100 images). We will talk about more detail about the result in next chapter.

Chapter 5

Results and Discussion

In this chapter, we discuss about the results of the experiments and demonstrating visual results for the viewers.

5.1 Comparison of different SRCNN configuration

Table 5.1 shows the performance comparison between interpolation methods and different SRCNN configuration. As the table shown, SRCNN methods outperformed interpolation methods. Among SRCNN configuration, the 9-5-5 model has the best performance. However, 9-5-5 model is just slightly better than 9-1-5 model and 9-3-5 model but need longest time to train (Table 5.2). In the next section we mainly compared the performance of 9-1-5 model and our proposed 3D-EDSR methods.

5.2 Comparison of image quality

Quantitative results are summarized in Table 5.3. As the tables shows, 3D-EDSR achieve the better results in MOS testing, but in PSNR and SSIM, SRCNN achieve better results. Bicubic gain a bit higher SSIM value than 3D-EDSR. In SRGAN's paper, author achieve state-of-the-art result in SSIM and PSNR by a upscale factor 4. And we only use 8 residual block in our proposed 3D-EDSR model.

Table 5.1: The results of SSIM and PSNR for different methods

Method	Evaluation	mean	std
Nearest Neighbor	SSIM	0.7295	0.0086
	PSNR	27.78	0.91
Bilinear	SSIM	0.7756	0.0091
	PSNR	29.07	0.90
Bicubic	SSIM	0.8233	0.0088
	PSNR	29.84	0.89
9-1-5	SSIM	0.9214	0.0059
	PSNR	32.11	0.77
9-3-5	SSIM	0.9371	0.0053
	PSNR	32.72	0.79
9-5-5	SSIM	**0.9376**	0.0053
	PSNR	**32.88**	0.77

Table 5.2: Training time of different SRCNN model

Model	9-1-5	9-3-5	9-5-5
Training time (h)	17.4	23.9	27.3

5.3 Visual Results

Figure5.1 - 5.12 illustrates an example by using interpolation methods (nearest neighbor and bicubic), SRCNN and 3D-EDSR. In this example, SRCNN achieve 1.2 dB higher PSNR than our proposed 3D-EDSR. Accroding to the definition of PSNR, the higher of the PSNR, the better degraded image has been reconstructed to match the original image and the better the reconstructive algorithm. SRCNN gained higher

Table 5.3: Comparison of NN, Bicubic, SRCNN , 3D-EDSR.

	PSNR	SSIM	MOS
Nearest Neighbor	25.94	0.7124	1.76
Bicubic	26.14	0.7439	2.07
SRCNN	**28.12**	**0.7841**	2.51
3D-ResNet	26.43	0.7405	**3.69**
Ground Truth	/	1	4.32

PSNR value than 3D-EDSR means SRCNN outperformed 3D-EDSR. In fact, compared the image generated by SRCNN, the image reconstructed by 3D-EDSR have more sharp edges which seems more clear for human being.

Fig. 5.1: Blurred image

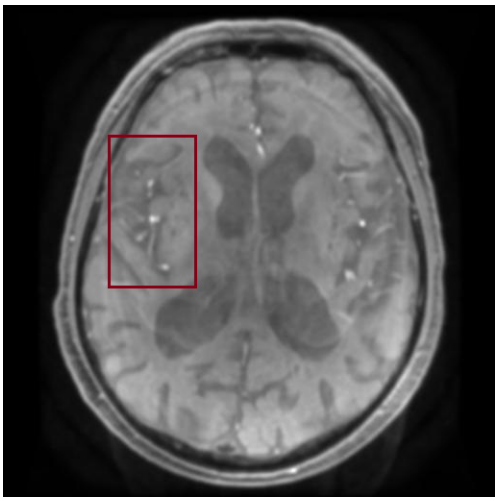


Fig. 5.2: Blurred image

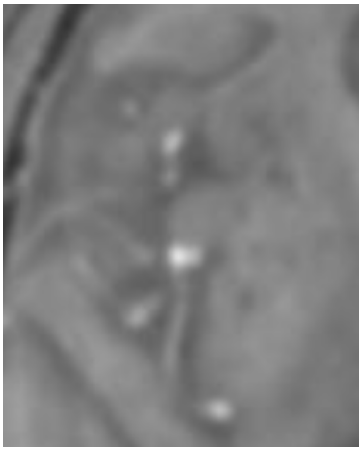


Fig. 5.3: nearest neighbor image

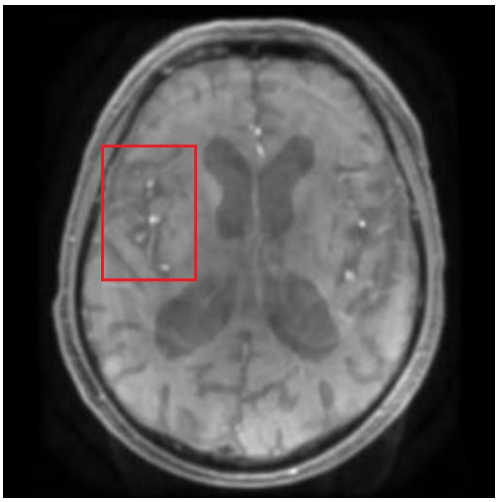


Fig. 5.4: enlarged NN image

Fig. 5.5: Bicubic image

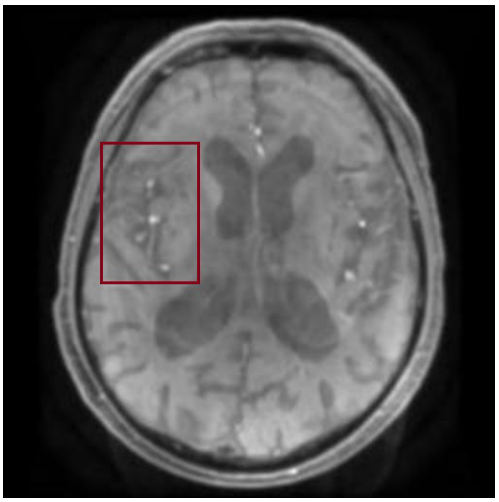


Fig. 5.6: enlarged bicubic image

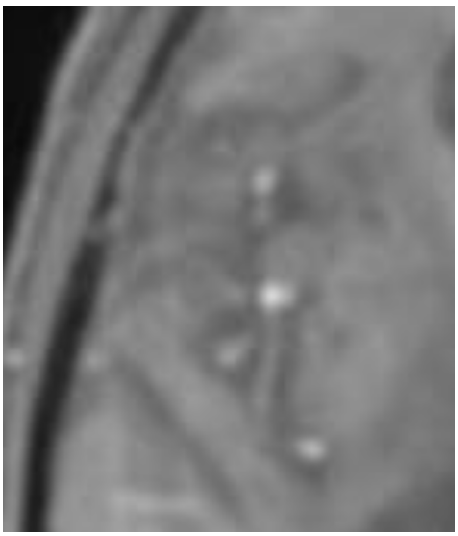


Fig. 5.7: SRCNN image

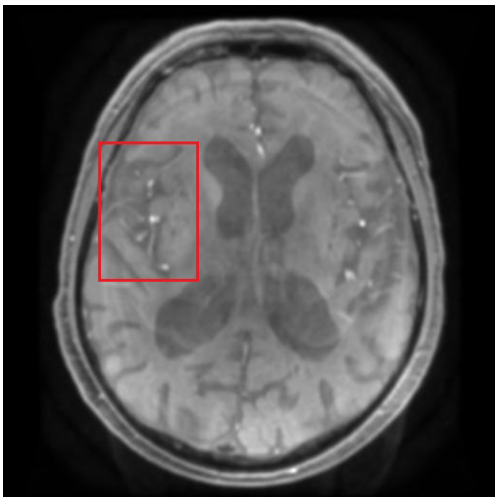


Fig. 5.8: enlarged SRCNN image

Fig. 5.9: 3D network image

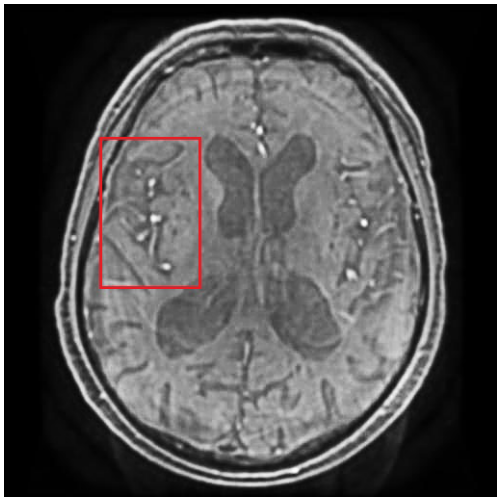


Fig. 5.10: enlarged image

Fig. 5.11: Ground Truth

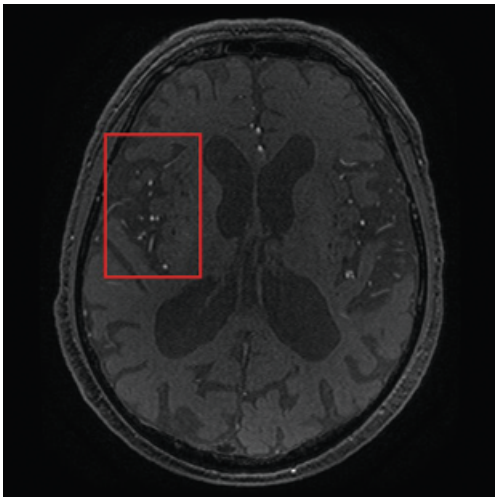
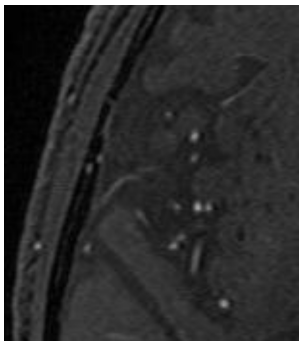


Fig. 5.12: Ground Truth



Chapter 6

Conclusion

This dissertation consisted of introduction of convolutional neural networks, methods for single image super resolution, a 2D-SRCNN model has been applied and evaluated, also a new 3 dimensional EDSR or MRI have been proposed.

We compared different SRCNN configurations, we found out that 9-5-5 model achieved best performance but it took too much time to train. 9-1-5 model can achieved similar results compared with 9-5-5 model but is 1.5x faster. Hence, in the following experiments, we use 9-1-5 model to represent SRCNN method. SRCNN achieved higher PSNR than interpolation methods, but it still generate visually over-smoothed images.

We showed that our proposed methods can recover local image textures and details more accurately, and our reconstruction images have more shaper edges than SRCNN and interpolation methods. However, our proposed methods didn't achieved higher PSNR and SSIM compared with SRCNN methods.

There are several reasons that might explain why our proposed method didn't achieve higher PSNR than SRCNN methods. First, our proposed network still not "deep" enough, we only use eight residual blocks in our network. Many researchers use more than 16 residual block in their network to improve the performance of their networks. In the future works, we should try more residual blocks to improve the performance of our proposed method.

Second, we generated images which upscale factor is 2. In SRGAN, authors used an upscale factor 4 to generate images and achieved highest PSNR and SSIM values

compare with other methods. We should try different upscale factor instead of scale factor 2 in the future.

Third, and maybe the most imporatant reasons. We use perceptual loss instead of MSE loss in our proposed model. As seen from the inverse relationship between the MSE and PSNR, to achieve higher PSNR is to minimize MSE loss. The easiest way to minimize the MSE loss is to blur the image. Maybe that is the reason why SRCNN generate over-smoothed images but still have higher PSNR value. Our proposed method has highest mean opinion score, but mean opinion score depends on subjective visual impression. We need to evaluate our methods in objective evaluation methods. Since PSNR and SSIM have it own limitaion, we should try another evaluation method instead of PSNR and SSIM to evaluate our proposed methods.

Acknowledgement

I would like to express my great appreciation to my supervisor, Associate Professor Sinichi Yoshida. His insights lead me to the original proposal to build a three-dimensional convolutional neural networks to reconstruct a high-resolution images. I am glad to have him as my supervisor, who provide his heartfelt support and guidance at all times and has given me invaluable guidance, inspiration and suggestions in my quest for knowledge. He has given me all support and freedom to pursue research, while silently and non-obtrusively ensuring that I stay on course and do not deviate from the core of my research. Without his able guidance, this thesis would not have been possible and I am grateful to him for his assistance.

I would like to thank everyone in the Yoshida laboratory for sharing all these happy and sad moments through my postgraduate years with me. This research would not be finished without the assistance of their technical supports. I would also like to extend my thanks to the secretaries of the Informatics courses and members of IRC in Kochi University of Technology for their selfless help when I was in troubles.

I also wish to acknowledge Kochi University of Technology and Japan Student Services Organization (JASSO) for giving me the great opportunity of financial supports which help me studying in Japan.

It would be inappropriate if I omit to mention the name of my dear friend Li Xiang, she kept me going on my path to success, assisting me as possible as she could, in whatever manner possible and for ensuring that good times keep flowing. She never let things get dull or boring, have all made a tremendous contribution in helping me reach this stage in my life. She putting up with me in difficult moments where I felt stumped and for goading me on to follow my dream of getting this degree.

Acknowledgement

I would like to dedicate this work to my parents whose dreams for me have resulted in this achievement and without their loving upbringing and nurturing, I would not have been where I am today and what I am today. This would not have been possible without their unwavering and unselfish love and support given to me at all time.

References

[1] Kouame, D., & Ploquin, M. (2009, June). Super-resolution in medical imaging: An illustrative approach through ultrasound. In Biomedical Imaging: From Nano to Macro, 2009. ISBI'09. IEEE International Symposium on (pp. 249-252). IEEE.

[2] Krizhevsky, A., Sutskever, I.,& Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).

[3] Park, S. C., Park, M. K., & Kang, M. G. (2003). Super-resolution image reconstruction: a technical overview. IEEE signal processing magazine, 20(3), 21-36.

[4] Greenspan, H., Oz, G., Kiryati, N., & Peled, S. (2002). MRI inter-slice reconstruction using super-resolution. Magnetic resonance imaging, 20(5), 437-446.

[5] Peled, S., & Yeshurun, Y. (2001). Superresolution in MRI: application to human white matter fiber tract visualization by diffusion tensor imaging. Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine, 45(1), 29-35.

[6] Yang, C. Y., Ma, C., & Yang, M. H. (2014, September). Single-image super-resolution: A benchmark. In European Conference on Computer Vision (pp. 372-386). Springer, Cham.

[7] Zhou, F., Yang, W., & Liao, Q. (2012). Interpolation-based image super-resolution using multisurface fitting. IEEE Transactions on Image Processing, 21(7), 3312-3318.

[8] Dai, S., Han, M., Xu, W., Wu, Y., Gong, Y., & Katsaggelos, A. K. (2009). Softcuts: a soft edge smoothness prior for color image super-resolution. IEEE Transactions on Image Processing, 18(5), 969-981.

[9] Sun, J., Xu, Z., & Shum, H. Y. (2008, June). Image super-resolution using gradient profile prior. In Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on (pp. 1-8). IEEE.

[10] Yan, Q., Xu, Y., Yang, X., & Nguyen, T. Q. (2015). Single image superresolution based on gradient profile sharpness. IEEE Transactions on Image Processing, 24(10), 3187-3202.

[11] Marquina, A., & Osher, S. J. (2008). Image super-resolution by TV-regularization and Bregman iteration. Journal of Scientific Computing, 37(3), 367-382.

[12] Baker, S., & Kanade, T. (2002). Limits on super-resolution and how to break them. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(9), 1167-1183.

[13] Elad, M., & Feuer, A. (1997). Restoration of a single superresolution image from several blurred, noisy, and undersampled measured images. IEEE transactions on image processing, 6(12), 1646-1658.

[14] Lin, Z., & Shum, H. Y. (2004). Fundamental limits of reconstruction-based superresolution algorithms under local translation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 26(1), 83-97.

[15] Timofte, R., De Smet, V., & Van Gool, L. (2013). Anchored neighborhood regression for fast example-based super-resolution. In Proceedings of the IEEE International Conference on Computer Vision (pp. 1920-1927).

[16] Freeman, W. T., Jones, T. R., & Pasztor, E. C. (2002). Example-based superresolution. IEEE Computer graphics and Applications, 22(2), 56-65.

[17] Chang, H., Yeung, D. Y., & Xiong, Y. (2004, June). Super-resolution through neighbor embedding. In Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on (Vol. 1, pp. I-I). IEEE.

[18] Aharon, M., Elad, M., & Bruckstein, A. (2006). K-SVD: An algorithm for designing

overcomplete dictionaries for sparse representation. IEEE Transactions on signal processing, 54(11), 4311.

[19] Yang, J., Wright, J., Huang, T. S., & Ma, Y. (2010). Image super-resolution via sparse representation. IEEE transactions on image processing, 19(11), 2861-2873.

[20] Dong, W., Zhang, L., & Shi, G. (2011, November). Centralized sparse representation for image restoration. In Computer Vision (ICCV), 2011 IEEE International Conference on (pp. 1259-1266). IEEE.

[21] Yang, S., Wang, M., Chen, Y., & Sun, Y. (2012). Single-image super-resolution reconstruction via learned geometric dictionaries and clustered sparse coding. IEEE Transactions on Image Processing, 21(9), 4016-4028.

[22] Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. IEEE transactions on pattern analysis and machine intelligence, 35(8), 1798-1828.

[23] Dong, C., Loy, C. C., He, K., & Tang, X. (2014, September). Learning a deep convolutional network for image super-resolution. In European conference on computer vision (pp. 184-199). Springer, Cham.

[24] Dong, C., Loy, C. C., He, K., & Tang, X. (2016). Image super-resolution using deep convolutional networks. IEEE transactions on pattern analysis and machine intelligence, 38(2), 295-307.

[25] Mathieson, J. R., Mayo, J. R., Staples, C. A., & Müller, N. L. (1989). Chronic diffuse infiltrative lung disease: comparison of diagnostic accuracy of CT and chest radiography. Radiology, 171(1), 111-116.

[26] Umehara K, Ota J, Ishimaru N, Ohno S, Okamoto K, Suzuki T, Shirai N, Ishida T: Super-resolution convolutional neural network for the improvement of the image quality of magnified images in chest radiographs. Proc SPIE 10133: 101331P-1–101331P-7, 2017

[27] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).

[28] Lim, B., Son, S., Kim, H., Nah, S., & Lee, K. M. (2017, July). Enhanced deep residual networks for single image super-resolution. In The IEEE conference on computer vision and pattern recognition (CVPR) workshops (Vol. 1, No. 2, p. 4).

[29] Han, S., Mao, H., & Dally, W. J. (2015). Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. arXiv preprint arXiv:1510.00149.

[30] Huynh-Thu, Q., & Ghanbari, M. (2008). Scope of validity of PSNR in image/video quality assessment. Electronics letters, 44(13), 800-801.

[31] Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing, 13(4), 600-612.

[32] Amodei, D., Ananthanarayanan, S., Anubhai, R., Bai, J., Battenberg, E., Case, C., ... & Chen, J. (2016, June). Deep speech 2: End-to-end speech recognition in english and mandarin. In International Conference on Machine Learning (pp. 173-182).

[33] Agichtein, E., Brill, E., & Dumais, S. (2006, August). Improving web search ranking by incorporating user behavior information. In Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval (pp. 19-26). ACM.

[34] Huang, P. S., He, X., Gao, J., Deng, L., Acero, A., & Heck, L. (2013, October). Learning deep structured semantic models for web search using clickthrough data. In Proceedings of the 22nd ACM international conference on Conference on information & knowledge management (pp. 2333-2338). ACM.

[35] Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., & Oliva, A. (2014). Learning deep features for scene recognition using places database. In Advances in neural information processing systems (pp. 487-495).

[36] Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., & Darrell, T. (2014, January). Decaf: A deep convolutional activation feature for generic visual recognition. In International conference on machine learning (pp. 647-655).

[37] Hubel, D. H., & Wiesel, T. N. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. The Journal of physiology, 160(1), 106-154.

[38] Fukushima, K., Hirota, M., Terasaki, P. I., Wakisaka, A., Togashi, H., Chia, D., ... & Hakomori, S. I. (1984). Characterization of sialosylated Lewisx as a new tumor-associated antigen. Cancer Research, 44(11), 5279-5285.

[39] Lenet, B. J., Komorowski, R., Wu, X. Y., Huang, J., Grad, H., Lawrence, H. P., & Friedman, S. (2000). Antimicrobial substantivity of bovine root dentin exposed to different chlorhexidine delivery vehicles. Journal of endodontics, 26(11), 652-655.

[40] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324.

[41] Schmidhuber, J. (2015). Deep learning in neural networks: An overview. Neural networks, 61, 85-117.

[42] Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In Proceedings of the 27th international conference on machine learning (ICML-10) (pp. 807-814).

[43] Tolias, G., Sicre, R., & Jégou, H. (2015). Particular object retrieval with integral max-pooling of CNN activations. arXiv preprint arXiv:1511.05879.

[44] Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., ... & Shi, W. (2017). Photo-realistic single image super-resolution using a generative

adversarial network. arXiv preprint.

[45] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial nets. In Advances in neural information processing systems (pp. 2672-2680).

[46] Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167.

[47] P. Gupta, P. Srivastava, S. Bhardwaj, and V. Bhateja. A modified psnr metric based on hvs for quality assessment of color images. In IEEE International Conference on Communication and Industrial Application (ICCIA), pages 1–4, 2011. 1

[48] Z. Wang, E. P. Simoncelli, and A. C. Bovik. Multi-scale structural similarity for image quality assessment. In IEEE Asilomar Conference on Signals, Systems and Computers, volume 2, pages 9–13, 2003

[49] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: From error visibility to structural similarity. IEEE Transactions on Image Processing, 13(4):600–612, 2004.