Deep Learning Based Super Resolution of Gravity Data for Geophysical Exploration

Yi Zhang
Louisiana State University and Agricultural and Mechanical College

Follow this and additional works at: https://digitalcommons.lsu.edu/gradschool_theses

Recommended Citation
https://digitalcommons.lsu.edu/gradschool_theses/5476

This Thesis is brought to you for free and open access by the Graduate School at LSU Digital Commons. It has been accepted for inclusion in LSU Master's Theses by an authorized graduate school editor of LSU Digital Commons. For more information, please contact gradetd@lsu.edu.
DEEP LEARNING BASED SUPER RESOLUTION OF GRAVITY DATA FOR GEOPHYSICAL EXPLORATION

A Thesis

Submitted to the Graduate Faculty of the
Louisiana State University and
Agricultural and Mechanical College
in partial fulfillment
requirement for the degree of
Master of Science

in

Division of Computer Science and Engineering

by

Yi Zhang
B.S., Xi’an University of Architecture and Technology, 2005
M.S., Donghua University, 2008
Ph.D., Louisiana State University, 2017
December, 2021
ACKNOWLEDGEMENTS

There are many people that I would like to acknowledge for their help and support, I would not have been able to complete this thesis and earn my master degree without their help.

I wish to express my deepest gratitude to my advisor, Dr. Xin (Shane) Li, for his constant guidance, patience, and encouragement throughout my graduate studies. And for the long hours he devoted to this thesis through his guidance, review, and advice. His knowledge, invaluable assistance, and greatest patience during each step of this process have been essential towards a timely completion. Without his abundant insight, feedback, and encouragement, I would never have completed this project. I feel very blessed to have had Dr. Li as my supervising professor. I would also like to express gratitude to the co-advisor Dr. Jyotsna Sharma for the long time she dedicated to this thesis through her priceless guidance, constant assistance, and molding this thesis into what it is. Also, my gratitude extends to my committee Dr. Jinwen Ye for her advice and cooperation.

Particular gratitude goes to Dr. Matthew Korban for his help on the method selection and pipeline completion. Special thanks go to Dr. Zewei He for helping on data visualization and my countless questions. I also gratefully acknowledge colleague Yongqing Liang for solving any research problem I met during each step.

Further, I wish to express my love and gratitude to my beloved families for their understanding and endless love through the duration of my studies. I am grateful for my husband, Dr. Chengjun Zhou and children Jason and Jared who are always my source of motivation and encouraged me to study. I would like to sincerely thank my mother-in-law, Runying Zong for her support, without her help to take care family, I would not have made it to the finish line.
## TABLE OF CONTENTS

ACKNOWLEDGEMENTS............................................................................................................ ii

ABSTRACT.................................................................................................................................... iv

LIST OF TABLES.......................................................................................................................... v

LIST OF FIGURES....................................................................................................................... vi

CHAPTER 1. INTRODUCTION...................................................................................................... 1
  1.1. Background.......................................................................................................................... 1
  1.2. Objective of Present Research .......................................................................................... 2

CHAPTER 2. LITERATURE REVIEW.......................................................................................... 4
  2.1. Super Resolution Overview ............................................................................................. 4
  2.2. Single Image Super Resolution ......................................................................................... 5
  2.3. Deep learning .................................................................................................................... 8
  2.4. Deep Learning for Single Image Super Resolution ........................................................... 9

CHAPTER 3. METHODOLOGY................................................................................................... 18
  3.1. Introduction........................................................................................................................ 18
  3.2. Pre-processing .................................................................................................................... 19
  3.3. Method .............................................................................................................................. 26
  3.4. Evaluation ......................................................................................................................... 28

CHAPTER 4. RESULTS AND DISCUSSION........................................................................... 32
  4.1. Results of Bouguer Gravity ............................................................................................... 32
  4.2. Results of Isostatic Gravity ............................................................................................. 43

CHAPTER 5. CONCLUSION ..................................................................................................... 48

Reference ..................................................................................................................................... 50

VITA.............................................................................................................................................. 57
ABSTRACT

Gravity measurement from ground can be obtained by precision gyro-stabilized gravimeters. However, land-based gravity measurements can be challenging in many hard-to-reach regions. By contrast, satellite-based gravity measurement is considered to be a very promising for large-scale exploration since it opens up the possibilities for geophysical study of remote regions. However, comparing with the gravity from ground, the resolution and precision of satellite-based gravity measurements is lower, which limits its application for geological explorations. The novel and effective methods in improving the low spatial resolution and precision of satellite-based gravity data are active quest.

Super resolution is used for naming any technique that exploit the knowledge contained in several low-resolution image to form a high resolution. There are many applications of super-resolution, which successfully improved medical imaging systems, satellite imaging, astronomical imaging. In our works, deep learning based super resolution (SR) was adopted to improving the low resolution and precision of satellite-based gravity data. Satellite-based gravity data and land-based gravity were visualized to low-resolution image and high-resolution images, respectively. Based on the different cropping methods, HR and LR were cropped into small patches with different size and different overlap ratio. And then modified super-resolution residual network (SRResNet) were trained by paired HR and LR small patches; and then evaluated using average absolute difference gravity $Avg(\Delta H_{avg})$, improvement ($Imp$), Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM). The effectiveness of modified SRResNet of trained models on small patches of different size and different overlap ratio was also investigated and confirmed.
LIST OF TABLES

Table 4. 1. Patches number in training set and testing set for 16 different model............... 33
Table 4. 2. Result for $16 \times 16$ US patches trained models .............................................. 35
Table 4. 3. Results for $32 \times 32$ US patches trained models............................................. 36
Table 4. 4. Results for $48 \times 48$ US patches trained models............................................. 36
Table 4. 5. Results for $64 \times 64$ US patches trained models............................................. 37
Table 4. 6. Bouguer gravity results of US from different methods. ....................................... 39
Table 4. 7. Average PSNR/SSIM results of different method on testing set...................... 40
Table 4. 8. Results for $16 \times 16$ additional CA patches ..................................................... 41
Table 4. 9. Results for $32 \times 32$ additional CA patches ..................................................... 41
Table 4. 10. Results for $48 \times 48$ additional CA patches .................................................... 42
Table 4. 11. Results for $64 \times 64$ additional CA patches .................................................... 42
Table 4. 12. Bouguer gravity results of CA from different models....................................... 43
Table 4. 13. Average PSNR/SSIM results of different method on additional CA testing set ...... 43
Table 4. 14. Patches number in training set and testing set for M1 and M2......................... 44
Table 4. 15. Results of M1 and M2 models ............................................................................ 46
Table 4. 16. $Avg\Delta H_{avg}$ of SR from US testing and CA testing set for different methods...... 47
Table 4. 17. Average PSNR/SSIM results of different method on US testing set................. 47
Table 4. 18. Average PSNR/SSIM results of different method on CA testing set.................. 47
LIST OF FIGURES

Figure 2. 1. A framework of super-resolution imaging ................................................................. 4
Figure 2. 2. (a): Images of Mike; (b): enlarged image; (c): SR image ........................................ 5
Figure 2. 3. Sketch of the overall framework of SISR ................................................................... 6
Figure 2. 4. Sketch of the SRCNN architecture ............................................................................ 10
Figure 2. 5. Sketch of very deep super resolution network (VDSR) ........................................ 11
Figure 2. 6. Sketch of deeply-recursive convolutional network (DRCN) ............................. 12
Figure 2. 7. Sketch of SR deep residual networks (SRResNet) .................................................. 12
Figure 2. 8. Sketch of Deep Recursive Residual Network (DRRN) ........................................... 13
Figure 2. 9. Sketch of enhanced deep super-resolution network (EDSR) ................................ 14
Figure 2. 10. Sketch of DenseNet work ...................................................................................... 14
Figure 3. 1. (a) HR from ground bouguer gravity data (b) Interpolated LR .............................. 20
Figure 3. 2. Processed bouguer gravity map (a) HR image (b) LR image .................................. 21
Figure 3. 3. Bouguer gravity map for CA (a) HR (b) Interpolated LR ....................................... 22
Figure 3. 4. Isostatic gravity for US (a) HR (b) Interpolated LR ............................................... 24
Figure 3. 5. Processed map (a) HR isostatic gravity image (b) LR free-air gravity image ....... 25
Figure 3. 6. Isostatic gravity for CA (a) HR (b) Interpolated LR .............................................. 26
Figure 3. 7. Schematic diagram of the SISR pipeline ................................................................. 27
Figure 3. 8. Removed the BN layers in residual block in SRResNet ........................................ 28
Figure 4. 1. Chosen area for additional test in CA (a) HR image and (b) LR image .............. 33
Figure 4. 2. HR, LR, SR, $\Delta H L$ and $\Delta H S$ in one $48 \times 48$ small patch ......................... 35
Figure 4. 3. Results from different patch size and different overlap ratio ................................. 38
Figure 4. 4. The performance comparison of different methods ........................................... 40

Figure 4. 5. Results from CA SR patches based on different trained models......................... 42

Figure 4. 6. Chosen training area and testing area in CA (a) HR image and (b) LR image ........ 45
CHAPTER 1. INTRODUCTION

1.1. Background

Among many methods applied in survey and exploration for petroleum resources, geophysical methods are the most successful and effective one, which deal with basins, boundaries of concealed or partially concealed basins, and the main structure within the basin in the early stage of exploration. Gravimetry is an important geophysical exploration method, which uses the local changes in acceleration due to gravity caused by non-uniform density distribution of crustal materials. Regional geological structures, basin boundaries, oil and gas trap structures, metallogenic belt and deposit structures can be identified by analyzing the gravity anomaly data[1].

Gravity measurement from ground can be obtained by precision gyrostabilized gravimeters[2]. However, land-based gravity measurements can be challenging in hard-to-reach regions such as mountains, bogs, boundary areas of the sea and land, polar caps of the Earth. Measuring gravity from satellite (or satellite gravimetry) is an emerging geophysical method which is considered to be a very promising for large-scale exploration [3][4]. Its main advantage is that it opens up the possibilities for geophysical study of remote regions. However, comparing with the gravity from ground, the resolution and precision of satellite-based gravity measurements is lower, which limits its application for geological explorations. The goal of this study is to improve the low spatial resolution and precision of satellite-based gravity data through the application of machine learning.

Deep learning is part of a broader family of machine learning algorithms to learn the hierarchical representation of data, which is based on the development of efficient computing hardware and the advancement of sophisticated algorithms [5]. Deep learning architectures such as deep neural networks, convolutional neural network, and deep reinforcement learning have been applied to many fields including computer vision, natural language processing, speech recognition,
medical image analysis and smart industry, where they have produced prominent superiority over other machine learning algorithms[6]–[10]. Image super resolution (SR), with the goal of estimating the clean high-resolution (HR) counterpart x of a given low-resolution (LR) image y, is a classical problem with highly academic and practical values[11]. In literature, a variety of classical SR methods have been proposed, including prediction-based methods [12]–[14], edge-based methods[15], [16], statistical methods[17], [18], patch-based methods[19]–[21], and sparse representation methods[22], [23], and so on. With the rapid development of deep learning techniques in recent years, deep learning-based SR model have been actively explored and often achieve the state-of-the-art performance on various benchmark of SR. Many deep learning methods have been applied to obtain SR models, ranging from the early Convolution Neural Networks (CNN) based methods (e.g., SRCNN[24], FSRCNN[25] ) to recent promising Generative Adversarial Nets (GAN) based methods (e.g., SRGAN[26], ESRGAN[27] )

In this work, ground-based gravity anomaly measurements represent the high-resolution (HR) image, while satellite-based gravity anomaly data represents the low-resolution (LR) image. For the image processing, SR was employed to enhance the low-resolution input to higher resolution. Therefore, for gravity data, the satellite gravity data was up to the level of the ground gravity data, which offer more information.

1.2. Objective of Present Research

High-accuracy ground-based gravity data and low-accuracy satellite-based gravity data are available. The objective of present research is to conduct appropriate method on the collected data to obtain new effective trained model, which will be able to improve the satellite-based gravity data to the accurate level of high-accuracy ground-based gravity data. Therefore, for the regions
that the ground-based gravity was missed such as hard-to-reach regions, trained model will help to obtain high accuracy data from available satellite gravity data.
2.1. Super Resolution Overview

The basic principle of super resolution is using one low resolution image or a sequence of low-resolution image of a scene to create an image with higher spatial resolution that contain finer detail or content with higher frequencies than the low-resolution images. Since recording a digital image, there is always a natural loss of spatial resolution and usually some kind of motion blur and noise due to limitation of the imaging system as illustrated in Figure 2.1. As the development of image processing, there is a strong demand for providing the viewer high-resolution image not only for providing better visualization but also for extracting more information details.

Figure 2.1. A framework of super-resolution imaging

For better understand the super resolution, few fundamental concepts should be clarified. Firstly, the image resolution is fundamentally different from its physical size. The objective of SR is to produce an image with a clearer content, which is rather than simply achieving a larger size of image. As shown in Figure 2.2, (a) is the image of Mike with size of $80 \times 80$; (b) is enlarged image ($320 \times 320$) by applying pixel duplication method on the image (a); (c) an enlarged image
(320 × 320) by applying a SR algorithm based on images (a), and they show clearly the difference. The first priority goal of super resolution imaging is using the contents of input low resolution to produce output image containing with more clear and detailed contents. The physical size of the output image (in terms pf total number of pixels) could be the same as input image or enlarged one using an image interpolation method. Second, the super-resolution has been used in other industry field. Such as in the field of smart grid, to cope with intermittency of renewable energy and ensure the security of the smart grid, state estimation served as a basic tool for understanding the true states of a smart grid should be performed with high frequency. Super resolution is used to improve data completeness by recovering high-frequency data from low-frequency data[28]. Super resolution techniques have revolutionized the field of optical microscopy, and it overcoming the diffraction limit[29]

Figure 2. 2. (a): Images of Mike; (b): enlarged image; (c): SR image

2.2. Single Image Super Resolution

According to the number of inputs LR images, SR algorithms can be classified into: single image super-resolution (SISR) and multiple images super-resolution (MISR). SISR is much more popular than MISR because of its high efficiency. In this work, we mainly introduce single image
super-resolution. Enhancing and recovering a high-resolution image from a low-resolution counterpart is a theme both of science fiction movies and of the scientific literature.

High Resolution (HR)

![High Resolution Image]

Low Resolution (LR)

**Figure 2.3. Sketch of the overall framework of SISR**

In the typical SISR framework, as shown in Figure 2.3, the LR image \( y \) is modeled as follows:

\[
y = (x \otimes k) \downarrow_s
\]  

(2.1)

Where \( x \otimes k \) is the convolution between the blurry kernel \( k \) and the unknown HR image \( x \), \( \downarrow_s \) is the downsampling operator with scale factor \( s \), and \( n \) is the independent noise term. Up to now, mainstream SISR algorithm are mainly divide into two categories: reconstruction-based methods and learning-based methods. Both types of methods involve training the algorithm with different types of training images.

### 2.2.1. Reconstruction-based single image SR methods

Reconstruction-based SISR methods have important scientific significance in image processing. Much work has been done with reconstruction based SR methods[16], [30]–[33]. Yan et al proposed a novel SR algorithm based on the edge sharpness metric of gradient profile sharpness (GPS), in which two gradient profile description models are proposed for representing gradient profiles with different lengths and different complicated shapes [30]. However, they often
required sophisticated prior knowledge, which restrict the possible solution space with an advantage of generating flexible and sharp details. In addition, the performance of many reconstruction-based methods degrades rapidly when the scale factor increases, and these methods are usually time-consuming. Therefore, more researchers adopted learning-based SISR methods.

2.2.2. Learning-based single image SR methods

The single image super resolution methods mostly employ learning-based method to generate the missing information of the SR images using the relationship between LR and HR image from a training database. Learning-based SISR methods alias example-based methods were first introduced by Mjolsness [34], where a neural network was adopted to improve the resolution of fingerprint images. They have been attracting the attention of researchers since their fast computation and outstanding performance. These methods contain a training step, which utilize machine learning algorithms to learn the relationship between the LR and its counterpart HR based on the training examples. The learned network is then incorporated into the reconstruction. The training database of learning-based SISR algorithms should have a proper generalization capability [35], which was measured by the two factors of sufficiency and predictability. However, the larger database does not guarantee better results, on the other hand, a larger number of irrelevant examples not only increase the computational time of searching for the best mapping, but also disturb this search [36]. To deal with this issue, classifying the image patches based on the content are necessary during the training.

Different types of learning-based SISR algorithms were adopted such as feature pyramids, belief network, projection-based network and neural network methods. The notable work in feature pyramids was developed by Baker and Kanade for face hallucination [11], [37], [38]. In the training step, each HR image is down-sampled and blurred several times to produce a Gaussian resolution
pyramid. Then, from Gaussian pyramids, Laplacian pyramids and then feature pyramids were generated. As one notable part of belief network, Markov network was developed by Freeman and Pasztor [39][40]. Both HR image and its LR counterparts are divided into patches, and then corresponding patches in the two images are associated to each other by an observation function, which defines how well a candidate HR patch matches a known LR patch[41]. After obtained the trained network, LR input images were improved by trained network to the super-resolution images. The projection-based SR methods for learning the mapping between HR and LR impose global constraints. To considering both local and global constraints, combing projection-based methods with other imposing local constraints was good idea. Such as Liu et al. combined projection-based methods with other imposing local constraints were and non-parametric Markov networks for face hallucination [42]. Different types of neural networks (NN) also have been employed in many different SR algorithms, and more details will be talked later.

2.3. Deep learning

Deep learning is a branch of machine learning algorithms based on learning diverse representations of data[43]. Traditional task-specific learning algorithms mostly select useful handcrafted features with expert domain knowledge. However, deep learning algorithms aim to learn informative hierarchical representation automatically, which will be used to achieve the final goal, and the whole learning process can be seen as an entirety[44].

Many modern deep learning models are based on artificial neural network (ANN) since it has high approximating capacity and hierarchical property[45]. Early ANNs originated from perceptron algorithms in the 1960s, and then the multilayer perceptron were trained with the backpropagation algorithm in the 1980s[46], [47]. Convolutional neural network (CNN) and recurrent neural network (RNN) are two representative derivatives of the traditional ANN, and
they were introduced to the computer vision and speech recognition fields, respectively[48], [49]. There still are many deficiencies handicapping ANNs, even ANNs achieved remarkable progress[50]. The rebirth of modern ANN was originated from pretraining the deep neural network (DNN) with the restricted Boltzmann machine (RBM) in the 2010s[51]. Models based on the DNN have achieved remarkable success in various field since the boom of computing power and the development of advanced algorithms[52]–[54]. Meanwhile, DNN-based unsupervised algorithms such as generative adversarial nets (GAN), variational autoencoder (VAE) have attracted much attention because of their potential dealing with challenging unlabeled data[55], [56].

2.4. Deep Learning for Single Image Super Resolution

With the rapid development of deep learning techniques in recent years, deep learning based SISR models have been actively explored and often achieve the state-of-the-art performance on various benchmarks of SR. Many deep learning methods have been applied to obtain SR models, ranging from the early Convolution Neural Networks (CNN) based methods to recent promising Generative Adversarial Nets (GAN) based methods. Deep learning SR methods differ from each other in the following aspects: different types of network architectures, different types of loss functions, different types of learning principles and strategies, etc.

2.4.1. Architecture for SISR

In this subsection, some different types of network architecture will be reviewed. SRCNN was the pioneering work of using a convolutional neural network in image super-resolution reconstruction development. SRCNN proposed by Dong et. al [24][57] utilizes three convolutional layers to predict the HR image and the overall architecture of SRCNN is shown in Figure 2.4.
As established in many traditional methods, SRCNN only implements simply the luminance components for training. In their works, SRCNN is a three-layer CNN, and the filter size of each layer are $64 \times 1 \times 9 \times 9$, $32 \times 64 \times 5 \times 5$, and $1 \times 32 \times 5 \times 5$. The function of these three nonlinear transformations are patch extraction, nonlinear mapping and reconstruction, and the loss function for optimizing SRCNN is the mean square error (MSE). We also notice that the formulation of SRCNN is relatively simple and very similar to an ordinary CNN that approximates the complex mapping between LR and corresponding HR in an end-to-end manner. SRCNN demonstrated vast superiority over concurrent tradition methods since CNN’s strong capability of learning valid representations from big data in an end-to-end manner. However, there still are problems occurred in SRCNN, the following question inspired more effective architecture: (a) This SRCNN is just three-layer architecture, could more complex CNN architecture with more depths, widths and topologies achieve better results? How shall we design greater models? (b) MSE loss function did not reflect properties of HR very well, could we integrate property of the SISR process into the CNN frame or other parts in the SISR algorithm? Based on the solution to these questions, more studies have been conducted.

Theoretical work in deep learning research shows that the solution space of a DNN expanded by increasing its depth or width[58]. To attain more hierarchical representations
effectively, many works mainly focus on improvements by increasing the depth. Various DL based applications have demonstrated the great power of deep neural networks even there are many training difficulties. Kim et al firstly conducted a highly accurate SISR method also called very deep super resolution (VDSR) inspired by VGG-net used for ImageNet classification[59]. Figure 2.5 shows a VDSR, which is 20-layer VGG-net[60][5]. Their contribution is a thorough evaluation of network of increasing depth by an architecture with a small (3 × 3) convolution filters, and the results show that a significant improvement on the prior-art configuration can be attained by pushing the depth to 16-19 weight layer.

The convolution kernel in the nonlinear mapping part of VDSR are very similar, Kim et al further proposed a deeply-recursive convolutional network (DRCN) to reduce parameters [61]. DRCN was the first algorithm that applied a recursive method for image super resolution. As shown in Figure 2.6, DRCN consisted of three major parts, namely embedding net, inference net, and reconstruction net. Figure 2.6 also shows that DRCN repeats the same convolution kernel in the nonlinear mapping part 16 times (16 recursions). Increasing recursion depth can improve performance without introducing new parameters for additional convolutions. The network efficiently reuses weight parameter while exploiting a large image context. To ease the difficulty
of training a deep recursive CNN, recursive-supervision and skip-connection two extensions were conducted.

Plain architecture such as VGG-net going deeper is hard, but various deep models based on skip-connection can be extremely deep and have obtained great performance. He et al proposed deep residual networks (ResNet) to ease the training of network, which reformulate the layers as learning residual functions with reference to the layer inputs, instead of learning unreferenced functions[62][63]. Leding et al proposed SRResNet with skip-connection and diverge from mean squared error, which composed of 16 residual units [64]. Each residual unit consists of two nonlinear convolutions with residual learning, and in there batch normalization is used to stabilize the training process[65]. The overall architecture of SRResNet is shown in Figure 2.7.
Two drawbacks were observed from DRCN: one was that DRCN requires supervision on every recursion, which was a burden for process; second was that there was only a single type of weight being shared among all convolutional layers in the inference net. To overcome these drawbacks, many other methods are proposed. Such as based on the original residual unit[63], Tai \textit{et al} proposed a very deep CNN model (up to 52 convolutional layers) named Deep Recursive Residual Network (DRNN) that strives for deep yet concise networks[66]. In DRRN, the basic residual units are rearranged in a recursive topology to form a recursive block, and Figure 2.8 shows the details. Each blocks shares the same parameters and is reused recursively to accommodate parameter reduction, which is as the single recursive convolution kernel in DRCN. Extensive benchmark evaluation shows that DRRN is a deep, concise, and superior model for SISR.

![Figure 2.8. Sketch of Deep Recursive Residual Network (DRRN)](image)

Inspired by the residual network in VDSR and the network in SRResNet, enhanced deep residual network was proposed to overcome the problem of heavy computation time and memory consumption. Lim \textit{et al} developed EDSR by removing unnecessary modules in conventional residual network and expanding the model size while they stabilize the training procedure, as shown in Figure 2.9[67]. Comparing with ResNet[63] and SRResNet[64], batch normalization layers were removed. Therefore, EDSR get rid of range flexibility from networks by normalizing
the feature since batch normalization layers normalize the features. Furthermore, EDSR also reducing 40% GPU memory usage during training since batch normalization layers consume the same amount of memory as preceding convolutional layers.

Figure 2.9. Sketch of enhanced deep super-resolution network (EDSR)

Huang et al proposed another effective architecture called DenseNet based on skip connection [68]. Tong et al conducted network, where the feature maps of each layer are propagated into all subsequent layers, provide an effective way to combine the low-level features and high-level features to boost the reconstruction performance[69]. The dense skip connection enables short paths to be built directly from the output to each layer as shown in Figure 2.10, which alleviate the vanishing gradient problem of very deep network.

Figure 2.10. Sketch of DenseNet work
2.4.2. Loss function for SISR

Loss function is a type of learning strategy used in machine learning to measure prediction error or reconstruction error, and it provides a guide for model optimization[5]. For deep learning-based image super-resolution methods, mean squared error (MSE) as known as $L_2$ loss is often adopted for training the network. $MSE$ can be expressed as in Equation 2.2

$$L_2 = \frac{1}{hwc} \sum_{i,j,k} (\hat{I}_{i,j,k} - I_{i,j,k})^2$$  \hspace{1cm} (2.2)

Where $h, w$ is the height of the image and the width of the image; and $c$ is the number of channels of the image; $\hat{I}_{i,j,k}$ is the constructed individual pixel value at row $i$, column $j$ and channel $k$; $I_{i,j,k}$ is the ground truth individual pixel value at row $i$, column $j$ and channel $k$. $L_2$ loss is good for a model to get a high peak-signal-to-noise ratio (PSNR), which is an indicator to evaluate model performance.

Mean absolute error ($MAE$) is another type of loss, also known as $L_1$ loss. Comparing with $L_2$ loss, may $L_1$ loss not help the model in achieving a better PSNR, but $L_1$ loss provides a powerful accuracy and convergence ability to the model[70]. $MAE$ can be expressed as in Equation 2.3

$$L_1 = \frac{1}{hwc} \sum_{i,j,k} |\hat{I}_{i,j,k} - I_{i,j,k}|$$  \hspace{1cm} (2.3)

In addition, there is a variant of the $L_1$ loss, namely Charbonnier loss, express by Equation 2.4[71]:

$$L_{Cha} = \frac{1}{hwc} \sum_{i,j,k} \sqrt{(\hat{I}_{i,j,k} - I_{i,j,k})^2 + \epsilon^2}$$  \hspace{1cm} (2.4)

Where $\epsilon$ is a constant (e.g., $10^{-3}$) for numerical stability.

Chu et al employed a flexible two-parameter loss function in a novel multiconnected convolutional network for super resolution (MCSR) to optimize the training process[72]. MCSR loss function
can be viewed as a generalization of many popular loss function in robust statistics. The MCSR loss function $L_{MCSR}$ was defined as in Equation 2.5

$$L_{MCSR}(r, \alpha, \beta) = \begin{cases} 
\log_4 \left[ \frac{1}{2} (\frac{r}{\beta})^2 + 1 \right] & \alpha = 0 \\
1 - \exp \left[ -\frac{1}{2} (\frac{r}{\beta})^2 \right] & \alpha = -\infty \\
\frac{\rho(\alpha)}{\alpha} \left[ \left( \frac{1}{\rho(\alpha)} (\frac{r}{\beta})^2 + 1 \right)^{\frac{\alpha}{2}} - 1 \right] & \text{otherwise}
\end{cases}$$

(2.5)

Where $r$ is the pixel wise error between predicted and ground truth HR image, $\rho(\alpha) = \max(1, 2 - \alpha)$, $\alpha$ is the shape parameter that controls the robustness of the loss, and $\beta$ is the scale parameter that controls the size of the loss’s quadratic bowl. Equation 2.4 is a generalized equation, which can be represented as $L_1$ loss, $L_2$ loss, Charbonnier loss ($L_1 - L_2$ loss) by changing the value of $\alpha$ [73]. For example, when $\alpha = 1$, Equation 2.5 will be similar to the $L_1$ loss, when $\alpha = 2$, Equation 2.5 will be similar to the $L_2$ loss. Therefore, the tunable parameters provided a flexibility to the model to minimize the loss value and optimize the training process without constraint in one single type of loss function. Such as in MCSR, $\alpha$ and $\beta$ were set as 1.11 and 0.05, respectively.

In recent years, due to the powerful learning ability, GAN receive more and more attention and are introduced to various vision tasks. In super solution, it is straightforward to adopt adversarial learning, in which case we only need to treat the SR model as a generator, and define an extra discriminator to judge whether the input image is generated or not. Ledig [64] first propose SRGAN using adversarial loss based on cross entropy, as Equation 2.6:

$$\begin{align*}
L_{gan.ce.g} &= -\log D(\hat{I}) \\
L_{gan.ce.d} &= -\log D(I_s) - \log(1 - D(\hat{I}))
\end{align*}$$

(2.6)
Where $L_{gan,ce,g}$ and $L_{gan,ce,d}$ denote the adversarial loss of the generator (i.e., the SR model) and the discriminator D (i.e., a binary classifier), respectively, and $I_s$ represents images randomly sampled from the ground truths.
CHAPTER 3. METHODOLOGY

3.1. Introduction

The acceleration due to gravity at different regional geological regions is slightly different due to the non-uniform density distribution of crustal materials. The regional geological structures, such as basin boundaries, oil and gas trap structures, metallogenic belt and deposit structures can be identified by gravity data[1]. Therefore, gravimetry or the study of gravity anomaly information is one of the most powerful geophysical methods for the exploration of oil and gas. High accuracy gravity data is acquired from ground-based measurements, however, it is hard to take land-based measurements in remote, hard-to-reach region, like mountains, bogs, boundary area of the ocean and land. Measuring gravity from satellite or satellite gravimetry, has opened up the possibilities for geophysical study of hard-to-reach regions. However, the precision and spatial resolution of satellite gravity is lower than the land-based gravity measurements. Thus, the motivation of our study is to obtain high accuracy gravity data that also covers more regions.

Single image super resolution (SISR), as a fundamental low level vision problem, attracted our attention; since SISR aims at recovering a clear high-resolution (HR) counterpart image X from a given low-resolution (LR) image Y. In our work, the ground-based gravity values and satellite-based gravity values were analogous to high resolution pixel value and low resolution pixel value in image processing, respectively. Meanwhile, the satellite gravity is visualized to low resolution image and ground gravity is visualized to high resolution image, which transform the gravity problem into image processing. For the image processing, SISR methods was employed to enhance low resolution to high resolution. Corresponding to gravity, the satellite gravity was up to the level of the high accuracy ground gravity.
3.2. Pre-processing

In this work, the raw gravity data covered United States (US) and Canada (CA) include two types of gravity anomalies: bouguer gravity anomaly and isostatic gravity anomaly. Both of them adopted the same methodology, so in this chapter, we just use gravity term to represent both unless specified otherwise. In the following chapters, the results for the bouguer gravity and isostatic gravity would be discussed separately.

Ground gravity data were download from https://www.usgs.gov, while satellite gravity data were download from http://icgem.gfz-potsdam.de/calcgrid

In the gravity data visualization, Mercator projection were used for gravity map. In order to find difference between ground gravity value and satellite gravity value, we have to get the difference between HR and LR pixel by pixel since one pixel value represented one gravity value in gravity map. Therefore, we upscale (× 4) original LR image to the HR size using nearest-neighbor interpolation. In our work, LR means the interpolated image.

Figure 3.1 shows the bouguer gravity map in Mercator Projection for US (a) is the HR image from the original ground bouguer gravity data, (b) is from interpolated LR image, which has the same size as the HR image of US. In comparison to the US map, HR image shows a big difference since ground-based gravity data in boundaries of ocean and land is hard to obtain. Therefore, pre-processing is necessary to keep accuracy of images and the consistency between HR and LR. The pre-processing consists of three operations:

1. Remove the ocean area in LR image (as we only have land-based gravity in conterminous US). Figure 3.1(b) shows clearly the difference between land and ocean.

2. Remove the boundary area between ocean and land in HR image according to LR image.
3. Removed the lake portion and trimmed in LR image according to HR. HR and LR were matched to represent contiguous US.

Figure 3. 1. (a) HR from ground bouguer gravity data (b) Interpolated LR
After pre-processing, the final processed HR and LR were obtained, as shown in Figure 3.2. For both of HR and LR, their size is $2333 \times 1225$ in pixel. According to the US map, each pixel represents 1.9327 km.

Figure 3.2. Processed bouguer gravity map (a) HR image (b) LR image
For the CA bouguer gravity data visualization, Mercator projection were also used for to obtain CA bouguer gravity map. Original LR image of CA was upscaled (× 4) to the size of HR image by nearest-neighbor interpolation. Figure 3.3 shows the bouguer gravity map in Mercator Projection for CA (a) High resolution image from ground bouguer gravity data (b) Interpolated LR from original LR, which is from satellite bouguer gravity data. The big difference between HR and LR image were observed. The available ground bouguer gravity is limited for CA, likely due to inaccessible terrains. However, the satellite bouguer gravity is available for the entire region, which further demonstrate the advantage of satellite-based gravimetry. The available HR and LR images were matched to represent consistent area.

Figure 3. 3. Bouguer gravity map for CA (a) HR (b) Interpolated LR

For the isostatic gravity data visualization, Mercator projection were also used for US and CA to obtain isostatic gravity map. We would note here that while isostatic gravity anomaly was available for land-based gravity or HR data, only Bouguer and free-air gravity anomalies were
available for the satellite data or LR dataset. So, for the LR data, free-air anomaly is used instead. Original LR image was also upsampled (× 4) to the size of HR image by nearest-neighbor interpolation. Figure 3.4 (a) is HR image of US from the original isostatic gravity data, (b) is the from interpolated LR (free-air anomaly) image of US. Similar to the US bouguer gravity image processing steps, the pre-processing consisted of three operations for US isostatic gravity image, and the more details about the pre-processing are described previously. These pre-processing guaranteed the images accuracy and consistence between HR and LR images. Figure 3.5 shows the final HR isostatic gravity anomaly and LR free-air anomaly images. We observe that while the data is similar in most regions, some areas show marked differences in the western US region. However, since isostatic gravity anomaly was not available for satellite data, we have proceeded with using the free-air anomaly for LR data.
Figure 3. 4. Isostatic gravity for US (a) HR (b) Interpolated LR
Figure 3.5. Processed map (a) HR isostatic gravity image (b) LR free-air gravity image

For CA isostatic HR gravity data visualization, Mercator projection were also used for CA isostatic gravity images. Similar to the US dataset, isostatic data from satellite was not readily available for Canada and therefore free-air gravity anomaly was used instead for LR dataset. Original LR image of CA was upscaled (×4) to the size of HR image by nearest-neighbor interpolation. Figure 3.6 shows the CA HR isostatic gravity and LR free-air gravity images, the big difference between HR and LR image in some regions was also observed.
3.3. Method

High-resolution image and low-resolution image were cropped into small patches, and the minimum patch size is $16 \times 16$. The schematic diagram of the pipeline is illustrated in Figure 3.7. First, a training set include HR and LR patch pairs is constructed; then a mapping model between the HR patches and the LR patches can be learned from the training set by using SISR methods; finally, new counterpart SR patches can be reconstructed from their LR patches using the model found in the learning stage of the process.

Figure 3. 6. Isostatic gravity for CA (a) HR (b) Interpolated LR
Since SRResNet [64] is a well-known DNN-based super resolver, in our work we used a modified SRResNet, namely SRResNet+ to improve the network performance. There were two differences between SRResNet+ and SRResNet. Firstly, SRResNet+ increase the number of feature maps from 64 to 96. Secondly, SRResNet+ removes the batch normalization (BN) layer as shown in Figure 3.8. As we know, increasing the number of feature maps enhance the network performance. Removing BN layers can increase performance and reduce computational complexity in SR and deblurring, which has been proved[74][75]. BN layers normalized the feature using mean variance in a batch during training step, and then it adopted the estimated man and various of the whole training dataset during test step. If the statistic value of the training datasets is much different from the value of testing dataset, BN layers would bring unpleasant...
artifacts and reduce the generation ability. Therefore, removing BN layers is for stable training and consistent performance. Additionally, it might result to improve generation ability and reduce computational complexity.

![Residual Block (RB)](image)

Figure 3. 8. Removed the BN layers in residual block in SRResNet

### 3.4. Evaluation

Most image quality evaluation methods in SISR adopted *Peak Signal-to-Noise Ratio* (PSNR) and *Structural Similarity Index* (SSIM)[76]. Both of them compare numerical criteria with the ground truth (HR), which is being the reference image. However, for our work, the difference value of gravity is also critical.

#### 3.4.1. Difference Value of Gravity

For a pair of small batch HR and LR, the difference value of gravity for each pixel is expressed in Equation 3.1

$$\Delta H_L = HR(gravity) - LR(gravity)$$  \hspace{1cm} (3.1)

And the average absolute difference value between HR and LR patch is

$$\Delta H_{L_{avg}} = \frac{1}{n} \sum_{1}^{n} |\Delta H_L|$$  \hspace{1cm} (3.2)

Where n is the pixels number in one small patch.
After LR patch reconstructed by learned SR model, patch SR was obtained, HR small patch gravity is still be reference.

\[ \Delta HS = HR(gravity) - SR(gravity) \] (3.3)

And the average absolute difference value of the whole patch is

\[ \Delta HS_{avg} = \frac{1}{n} \sum_{1}^{n} |\Delta HS| \] (3.4)

The average absolute difference value for the training set or testing set is expressed by the following equations:

\[ Avg(\Delta HL_{avg}) = \frac{1}{t} \sum_{1}^{t} \Delta HL_{avg} \] (3.5)

\[ Avg(\Delta HS_{avg}) = \frac{1}{t} \sum_{1}^{t} \Delta HS_{avg} \] (3.6)

And then, the improvement (Impr) was used to evaluate the trained modes

\[ Impr = \frac{Avg(\Delta HL_{avg}) - Avg(\Delta HS_{avg})}{Avg(\Delta HL_{avg})} \times 100\% \] (3.7)

Where t is the number of patches in training set or testing set. Impr is based on all patches in the training set/testing set instead of one small patch.

3.4.2. Peak Signal-to-Noise Ratio

For image super resolution, PSNR is defined via the maximum pixel value and the mean squared error (MSE) of processed image, \( h(x, y) \), and the high-resolution reference image, \( f(x, y) \),

\[ MSE = \frac{1}{mn} \sum_{x=0}^{m} \sum_{y=0}^{n} [f(x, y) - h(x, y)]^2 \] (3.8)

And, the PSNR is defined as
\[ PSNR(dB) = 10 \times \log_{10}\left( \frac{MAX_f^2}{MSE} \right) \] (3.9)

Where \( MAX_f \) is the maximum possible pixel value of image, and the value equals to 255. If the processed image is similar to the reference image, the \( MSE \) will be close to zero, and \( PSNR \) will be to infinity. A higher similarity between \( h \) and \( f \), the \( PSNR \) will be higher.

### 3.4.3. Structural Similarity Index

Human visual system is highly adapted to extract image structure, so the structural similarity index is proposed in SISR for measuring the structure similarity between processed image \( h \) and high-resolution reference image \( f \) \[76][77]. Instead of calculating the difference of the \( h \) and \( f \), the comparison between of them is performed on the basis of the following features: Luminance, contrast, and structure.

Luminance is measured by averaging over all the pixel value. It is denoted by \( \mu \) and the formula is given below

\[
\mu_f = \frac{1}{N} \sum_{i=1}^{N} f_i 
\] (3.10)

Luminance comparison function is defined by a function \( l(f, h) \), which is shown below. \( f, h \) are the processed image and the high-resolution reference image being compared.

\[
l(f, h) = \frac{(2\mu_f\mu_h + C_1)}{\mu_f^2 + \mu_h^2 + C_1}
\] (3.11)

Contrast is measured by taking the standard deviation (square root of variance) of all the pixel values. It is denoted by \( \sigma \) and represented by the formula below

\[
\sigma_f = \left( \frac{1}{N-1} \sum_{i=1}^{N} \left( f_i - \mu_f \right)^2 \right)^{1/2}
\] (3.12)
Contrast comparison function is defined by a function \( c(f, h) \), which is shown below:

\[
c(f, h) = \frac{(2\sigma_f \sigma_h + C_2)}{(\sigma_f^2 + \sigma_h^2 + C_2)}
\]  

(3.13)

Structure comparison function is defined by the function \( s(f, h) \), which is expressed as:

\[
s(f, h) = \frac{(2\sigma_{f,h} + C_3)}{(\sigma_f^2 \sigma_h^2 + C_3)}
\]  

(3.14)

Where \( \sigma_{f,h} \) is defined as:

\[
\sigma_{f,h} = \frac{1}{N-1} \sum_{i=1}^{N} (f_i - \mu_f)(h_i - \mu_h)
\]  

(3.15)

Finally, the SSIM score is:

\[
SSIM(f, h) = [l(f, h)]^\alpha [c(f, h)]^\beta [s(f, h)]^\gamma
\]  

(3.16)

Where \( \alpha > 0, \beta > 0, \gamma > 0 \) denote the relative importance of each of the metrics. To simplify the expression, if assume \( \alpha = \beta = \gamma = 1 \) and \( C_3 = \frac{C_2}{2} \):

\[
SSIM(f, h) = \frac{(2\mu_f \mu_h + C_1)(2\sigma_{f,h} + C_2)}{\left(\mu_f^2 + \mu_h^2 + C_1\right)\left(\sigma_f^2 + \sigma_h^2 + C_2\right)}
\]  

(3.17)

The SSIM value ranges from 0 to 1, where a value of 1 means that the processed image are identical to high resolution reference image; and a value of 0 indicates no structural similarity between them.
CHAPTER 4. RESULTS AND DISCUSSION

In this chapter, results of bouguer gravity and results of isostatic gravity will be discussed separately. In both bouguer gravity and isostatic gravity, modified SRResNet+ was adopted to obtain learned SR models. In bouguer gravity part, the learned SR models were investigated on the different cropped patch size with different overlap size. In isogravity part, two different models M1 and M2 were studied. M1 was learned from only US patches, and M2 was learned from mixture of US patches and CA patches. Then, super-resolved (SR) patches from learned SR models were evaluated by different methods. Finally, the results after comparing with other methods demonstrate our method is comparable.

4.1. Results of Bouguer Gravity

4.1.1. Experiment Details

After obtained processed US bouguer gravity HR images and LR image as shown in Figure 3.2, HR and LR images are cropped into patches orderly. Four different patch sizes 16 × 16, 32 × 32, 48 × 48, 64 × 64 pixels were chosen to investigate the effect of patch size. Besides patch size, we also studied the effect of the horizontal overlap of patches. The overlap ratio (overlap pixel in length) was set as lp0, 0.125, 0.25, and 0.375. Take 16 × 16 patch size as an example: lp0 means no overlap between two adjacent patches; lp0.125 means there are 2 pixels overlap; And so on, lp 0.25 means there are 4 pixels overlap; lp 0.375 means there are 6 pixels overlap. Therefore, total 16 different patch set were obtained. For each patch set, 80% of them was chosen randomly for the training set, and the left 20% was for the testing set. Table 4.1 shows the patch number of training set and testing set for the different models. Rotated and flipped versions of the training patches are considered. The original patches were rotated by 90°, 180°, 270°, and flipped horizontally.
Table 4.1. Patches number in training set and testing set for 16 different models

<table>
<thead>
<tr>
<th></th>
<th>16 × 16</th>
<th>32 × 32</th>
<th>48 × 48</th>
<th>64 × 64</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Train Set</td>
<td>Test Set</td>
<td>Train Set</td>
<td>Test Set</td>
</tr>
<tr>
<td>𝑙𝑝0</td>
<td>5482</td>
<td>1370</td>
<td>1308</td>
<td>326</td>
</tr>
<tr>
<td>𝑙𝑝0.125</td>
<td>6278</td>
<td>1569</td>
<td>1501</td>
<td>375</td>
</tr>
<tr>
<td>𝑙𝑝0.25</td>
<td>7319</td>
<td>1829</td>
<td>1744</td>
<td>436</td>
</tr>
<tr>
<td>𝑙𝑝0.375</td>
<td>8774</td>
<td>2193</td>
<td>2096</td>
<td>524</td>
</tr>
</tbody>
</table>

Adam algorithm was adopted to optimize SRResNet+ by minimizing the $L_1$ mean absolute error (MAE) loss function[78]. The learning rate starts from $10^{-4}$, then decreases by half every $5 \times 10^5$ iterations and finally ends once it is smaller than $10^{-7}$. After SR models were trained, besides the US testing data, part of CA HR and LR images (450 × 450 pixels) as additional image were used to test the trained SR models. There were many missing data in CA HR image, so just part of area was chosen for the additional test (marked in dash), which was shown in Figure 4.1.

Figure 4.1. Chosen area for additional test in CA (a) HR image and (b) LR image
4.1.2. Results in Bouguer Gravity Value of US

The differences between HR and LR, HR and SR in bouguer gravity are objective to evaluate the obtained models. For a pair of small batch HR and LR, the difference value of gravity (\(\Delta HL\)) is expressed in Equation 3.1

\[\Delta HL = HR(\text{gravity}) - LR(\text{gravity})\]

After LR patch reconstructed by learned SR model, patch SR was obtained, HR small patch gravity is still be reference. The difference value of gravity between and HR and SR (\(\Delta HS\)) is expressed in Equation 3.3

\[\Delta HS = HR(\text{gravity}) - SR(\text{gravity})\]

In the bouguer gravity calculation, its unit is mGal. Take one 48 \times 48 patch as example, HR, LR, SR improved by learned SR model with 48 \times 48 patch size and \(lp0\), the difference value between HR and LR, and the difference value between HR and SR were shown in Figure 4.2. we can see clearly that the difference between HR and SR is smaller than the difference between HR and SR. It demonstrated SR trained model is effective. The improvement (\(Impr\)) was used to evaluate the trained model by Equation 3.7

\[Impr = \frac{Avg(\Delta HL_{avg}) - Avg(\Delta HS_{avg})}{Avg(\Delta HL_{avg})} \times 100\%\]
The results of training set and testing set for different patch size and different overlap size are listed in the following tables. The results of SR model trained by 16 \times 16 patches were listed in Table 4.2. It shows that the \textit{Impr} on testing set don’t drop much from that on training set for every SR model with different overlap size. It demonstrated that no overfitting in training for every model. About the effect of overlap size, the \textit{lp0.125} was better than the other three overlap size for testing set.

Table 4.2. Result for 16 \times 16 US patches trained models

<table>
<thead>
<tr>
<th>Model</th>
<th>$\text{Avg}(\Delta H_{avg})$ (Train set)</th>
<th>$\text{Avg}(\Delta H_{avg})$ (Train set)</th>
<th>$\text{Impr}(%)$ (Train set)</th>
<th>$\text{Avg}(\Delta H_{avg})$ (Train set)</th>
<th>$\text{Avg}(\Delta H_{avg})$ (Train set)</th>
<th>$\text{Impr}(%)$ (Train set)</th>
</tr>
</thead>
<tbody>
<tr>
<td>\textit{lp0}</td>
<td>6.2484</td>
<td>5.9422</td>
<td>4.90%</td>
<td>6.1751</td>
<td>5.9002</td>
<td>4.45%</td>
</tr>
<tr>
<td>\textit{lp0.125}</td>
<td>6.3275</td>
<td>6.0530</td>
<td>4.34%</td>
<td>5.9996</td>
<td>5.7162</td>
<td>4.72%</td>
</tr>
<tr>
<td>\textit{lp0.25}</td>
<td>6.2828</td>
<td>5.9800</td>
<td>4.82%</td>
<td>6.1420</td>
<td>5.8425</td>
<td>4.88%</td>
</tr>
<tr>
<td>\textit{lp0.375}</td>
<td>6.2208</td>
<td>5.9539</td>
<td>4.29%</td>
<td>6.3402</td>
<td>6.0596</td>
<td>4.23%</td>
</tr>
</tbody>
</table>

For 32 \times 32 patches, the results are listed in Table 4.3. The \textit{Impr} on testing set still don’t drop much from that on training set for every model with different overlap size. It demonstrated
that no overfitting in training for every SR model. About the effect of overlap size, the results are as well as $16 \times 16$, $lp$ 0.125 is better than the other three overlap size for testing set.

Table 4. 3. Results for $32 \times 32$ US patches trained models

<table>
<thead>
<tr>
<th>Model</th>
<th>$Avg(\Delta HL_{avg})$ (Train set)</th>
<th>$Avg(\Delta HS_{avg})$ (Train set)</th>
<th>$Impr(%)$ (Train set)</th>
<th>$Avg(\Delta HL_{avg})$ (Train set)</th>
<th>$Avg(\Delta HS_{avg})$ (Train set)</th>
<th>$Impr(%)$ (Train set)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$lp0$</td>
<td>5.6675</td>
<td>5.3569</td>
<td>5.48%</td>
<td>5.8549</td>
<td>5.5642</td>
<td>4.97%</td>
</tr>
<tr>
<td>$lp0.125$</td>
<td>5.9571</td>
<td>5.6487</td>
<td>5.18%</td>
<td>5.2213</td>
<td>4.9343</td>
<td>5.50%</td>
</tr>
<tr>
<td>$lp0.25$</td>
<td>5.6955</td>
<td>5.3598</td>
<td>5.89%</td>
<td>5.7020</td>
<td>5.3852</td>
<td>5.56%</td>
</tr>
<tr>
<td>$lp0.375$</td>
<td>5.7926</td>
<td>5.4970</td>
<td>5.10%</td>
<td>5.6818</td>
<td>5.4017</td>
<td>4.93%</td>
</tr>
</tbody>
</table>

The Table 4.4 show the results of model trained by $48 \times 48$ patches. As well as $16 \times 16$ and $32 \times 32$ patches, the $Impr$ on testing set still don’t drop much from that on training set for every trained SR model with different overlap size, which means no overfitting in training for every model. About the effect of overlap size, $lp$ 0.125 is better than the other three overlap size for testing set.

Table 4. 4. Results for $48 \times 48$ US patches trained models

<table>
<thead>
<tr>
<th>Model</th>
<th>$Avg(\Delta HL_{avg})$ (Train set)</th>
<th>$Avg(\Delta HS_{avg})$ (Train set)</th>
<th>$Impr(%)$ (Train set)</th>
<th>$Avg(\Delta HL_{avg})$ (Train set)</th>
<th>$Avg(\Delta HS_{avg})$ (Train set)</th>
<th>$Impr(%)$ (Train set)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$lp0$</td>
<td>5.7002</td>
<td>5.3148</td>
<td>6.76%</td>
<td>4.8732</td>
<td>4.5810</td>
<td>6.00%</td>
</tr>
<tr>
<td>$lp0.125$</td>
<td>5.6055</td>
<td>5.1486</td>
<td>8.15%</td>
<td>5.1918</td>
<td>4.8422</td>
<td>6.73%</td>
</tr>
<tr>
<td>$lp0.25$</td>
<td>5.5010</td>
<td>5.1034</td>
<td>7.23%</td>
<td>5.2166</td>
<td>4.8447</td>
<td>7.13%</td>
</tr>
<tr>
<td>$lp0.375$</td>
<td>5.5647</td>
<td>5.2768</td>
<td>5.17%</td>
<td>5.3078</td>
<td>5.0819</td>
<td>4.26%</td>
</tr>
</tbody>
</table>

The Table 4.5 show the results of model trained by $64 \times 64$ patches. For $Impr$, the result on testing set still don’t drop much from that on training set for every model with different overlap size, which means no overfitting in training for every model. About the effect of overlap size, the result is different from the other patch size. Instead of $lp0.25$, $lp0.125$ is better than the other three overlap size for testing set. Generally, overlapping patches have provide better result than adjacent patches. The reason is that the prediction of each pixel was affected by its neighborhoods[79]. For
the overlapping patches, more patches around the same pixel, more context would be captured by
network.

Table 4. 5. Results for 64 × 64 US patches trained models

<table>
<thead>
<tr>
<th>Model</th>
<th>$Avg(\Delta HL_{avg})$ (Train set)</th>
<th>$Avg(\Delta HS_{avg})$ (Train set)</th>
<th>$Impr$ (%) (Train set)</th>
<th>$Avg(\Delta HL_{avg})$ (Train set)</th>
<th>$Avg(\Delta HS_{avg})$ (Train set)</th>
<th>$Impr$ (%) (Train set)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$lp0$</td>
<td>5.3171</td>
<td>4.8361</td>
<td>9.05%</td>
<td>4.9355</td>
<td>4.6033</td>
<td>6.73%</td>
</tr>
<tr>
<td>$lp0.125$</td>
<td>5.1787</td>
<td>4.8199</td>
<td>6.93%</td>
<td>5.5327</td>
<td>5.1077</td>
<td>7.68%</td>
</tr>
<tr>
<td>$lp0.25$</td>
<td>5.2964</td>
<td>4.9165</td>
<td>7.17%</td>
<td>5.3305</td>
<td>4.9495</td>
<td>7.15%</td>
</tr>
<tr>
<td>$lp0.375$</td>
<td>5.3810</td>
<td>5.0249</td>
<td>6.62%</td>
<td>5.2426</td>
<td>4.9046</td>
<td>6.45%</td>
</tr>
</tbody>
</table>

To study the effect of patch size, $Impr$ values from each SR models trained by different patch
size and different overlap size were shown in Figure 4.3. For the testing set results, $Impr$ increase as the
increased patch size as shown in Figure 4.3 b. 64 × 64 patch size shows the better results than the other
three patch size since the network can capture more contextual information in larger patch[80]–[82].

About the effect of overlap ratio, from Figure 4.3b, we can observe that the $Impr$ increase as the
overlap ratio increase from $lp0$ to $lp0.25$, and then it decreases when $lp = 0.375$. Generally, a larger
overlap between patches will create a large number of redundant patches for the training process, which
can be much more computationally expensive. Thus, an appreciate selection of the overlap ratio to balance
performance and computational efficiency need to be carefully designed. However, in our work, the
64 × 64 patch size with $lp0.125$ has the largest $Impr$ value 7.68%, which do not have more overlap.
Therefore, larger patch size with an appropriate overlap ratio is a better cropping method to obtain the best model.
4.1.3. Compared Methods by US image

Quantitative and qualitative comparisons were compared between our model and the other five methods including SRCNN [57], VDSR[59], DRCN[61], DRRN[83], SRResNet[26]. Each model of different methods was trained by different patch size and different overlap ratio on the
training set, and then SR patches were enhanced by obtained trained SR model. Average difference value of bouguer gravity between LR and HR $\text{Avg} (\Delta H_{L_{avg}})$ and the average difference between SR and HR $\text{Avg} (\Delta H_{S_{avg}})$ are the average values of all different patch size and overlap SR, which were listed in Table 4.6. From Table 4.6, we can see clearly that our model has the lowest $\text{Avg} (\Delta H_{S_{avg}})$, which indicates our method works well. Red color indicates the best performance of our methods and blue color indicates the best performance of previous methods.

Table 4. 6. Bouguer gravity results of US from different methods.

<table>
<thead>
<tr>
<th>Method</th>
<th>LR $\text{Avg} (\Delta H_{L_{avg}})$</th>
<th>SRCNN $\text{Avg} (\Delta H_{S_{avg}})$</th>
<th>VDSR $\text{Avg} (\Delta H_{S_{avg}})$</th>
<th>DRCN $\text{Avg} (\Delta H_{S_{avg}})$</th>
<th>SRResNet $\text{Avg} (\Delta H_{S_{avg}})$</th>
<th>DRRN $\text{Avg} (\Delta H_{S_{avg}})$</th>
<th>Our $\text{Avg} (\Delta H_{S_{avg}})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>US (test set)</td>
<td>5.5467</td>
<td>5.5393</td>
<td>5.5097</td>
<td>5.3544</td>
<td>5.3516</td>
<td>5.2549</td>
<td>5.2324</td>
</tr>
</tbody>
</table>

PSNR and SSIM results are the average values from different patch size and overlap SR, which were listed in Table 4.7. Our method outperforms the other methods in both PSNR and SSIM, especially for SSIM. From Table 4.7, we have observations and analysis as follows. First, SRCNN, VDSR and DRCN did not perform very well. Second, DRRN produces better results than SRResNet, which also observed in other work [83], and our method is better than DRRN, which demonstrated that removing BN layers increase performance in SR [74][75].

As we discussed gravity value before, the best improvement is 7.68% for 64 × 64 patch size with $lp0.125$, which did not meet our expectation. There might be three reasons. One reason might be the small dataset size. Another reason might be the big scale factor × 4, such phenomenon has also been reported[84][85]. There might be a special reason. As we know, for the LR degraded from HR, the LR pixel value is same as the HR corresponding pixel value. But for our case, the original LR value is different from HR value.
Table 4. 7. Average PSNR/SSIM results of different method on testing set.

<table>
<thead>
<tr>
<th>Method</th>
<th>LR</th>
<th>SRCNN</th>
<th>VDSR</th>
<th>DRCN</th>
<th>SRResNet</th>
<th>DRRN</th>
<th>Ours</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSNR</td>
<td>27.27</td>
<td>27.48</td>
<td>27.60</td>
<td>27.82</td>
<td>27.84</td>
<td>27.97</td>
<td>28.19</td>
</tr>
<tr>
<td>SSIM</td>
<td>0.9159</td>
<td>0.9229</td>
<td>0.9252</td>
<td>0.9353</td>
<td>0.9378</td>
<td>0.9398</td>
<td>0.9456</td>
</tr>
</tbody>
</table>

Figure 4.4 illustrated the SISR results of different methods for scale factor 4 on 64 × 64 patche with $l_p0.125$. It can be observed from the visual results that our method produced more visually similar HR patch than competing methods. The performance of SRCNN and VDSR is severely affected by the “compression artifacts”. DRCN and SRResNet can successfully remove the “compression artifacts”, but they failed to recover sharp edge. In comparison, our method can not only remove the unsatisfying artifacts but also produce sharp edges. It also can be observed that DRRN tend to produce over-smoothed results, whereas our method can recover sharp image with better intensity.

Figure 4. 4. The performance comparison of different methods
4.1.4. Additional Testing Results for CA image

After SR models were trained by US patches, besides testing data of US patches, part of CA HR and LR images (450 × 450 pixels) as additional image are used to test the trained SR models. For the CA patches, they were cropped from HR and LR of CA image orderly, and there is no overlap between two adjacent patches. The patches sizes are 16 × 16, 32 × 32, 48 × 48, 64 × 64 as well as US patch size. In the Table 4.8-4.11 and Figure 4.5, the overlap size refers to the trained SR model, which are from overlapped training patches. All of CA patches are adjacent patches.

After obtained SR models trained by different patch size and different overlap size, CA additional testing set were enhanced and the results were listed in Table 4.8-4.11. Comparing with Table 4.2-4.5, we found that Impr of US testing set is much better than CA addition testing set for the same trained SR model. Meanwhile, it’s also noticed that the difference \( \text{Avg}(\Delta H_{\text{avg}}) \) between HR and LR for CA testing set is about 4.7 mGal, which is much smaller than the \( \text{Avg}(\Delta H_{\text{avg}}) \) in US testing set. It means that the consistency between CA data and US data is low, which might be explanation of the CA addition testing results is lower than US testing results.

Table 4. 8. Results for 16 × 16 additional CA patches

<table>
<thead>
<tr>
<th>Model</th>
<th>( \text{Avg}(\Delta H_{\text{avg}}) )</th>
<th>( \text{Avg}(\Delta H_{S_{\text{avg}}}) )</th>
<th>Impr(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( lp0 )</td>
<td>4.7276</td>
<td>4.6214</td>
<td>2.24%</td>
</tr>
<tr>
<td>( lp0.125 )</td>
<td>4.7276</td>
<td>4.6061</td>
<td>2.57%</td>
</tr>
<tr>
<td>( lp0.25 )</td>
<td>4.7276</td>
<td>4.6313</td>
<td>2.04%</td>
</tr>
<tr>
<td>( lp0.375 )</td>
<td>4.7276</td>
<td>4.6081</td>
<td>2.53%</td>
</tr>
</tbody>
</table>

Table 4. 9. Results for 32 × 32 additional CA patches

<table>
<thead>
<tr>
<th>Model</th>
<th>( \text{Avg}(\Delta H_{\text{avg}}) )</th>
<th>( \text{Avg}(\Delta H_{S_{\text{avg}}}) )</th>
<th>Impr(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( lp0 )</td>
<td>4.7276</td>
<td>4.5764</td>
<td>3.20%</td>
</tr>
<tr>
<td>( lp0.125 )</td>
<td>4.7276</td>
<td>4.6113</td>
<td>2.46%</td>
</tr>
<tr>
<td>( lp0.25 )</td>
<td>4.7276</td>
<td>4.6633</td>
<td>1.36%</td>
</tr>
<tr>
<td>( lp0.375 )</td>
<td>4.7276</td>
<td>4.5903</td>
<td>2.90%</td>
</tr>
</tbody>
</table>
Table 4.10. Results for 48 × 48 additional CA patches

<table>
<thead>
<tr>
<th>Model</th>
<th>Avg(ΔH_{avg})</th>
<th>Avg(ΔHS_{avg})</th>
<th>Impr(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>l0</td>
<td>4.6996</td>
<td>4.6072</td>
<td>1.97%</td>
</tr>
<tr>
<td>l0.125</td>
<td>4.6996</td>
<td>4.6018</td>
<td>2.08%</td>
</tr>
<tr>
<td>l0.25</td>
<td>4.6996</td>
<td>4.6008</td>
<td>2.10%</td>
</tr>
<tr>
<td>l0.375</td>
<td>4.6996</td>
<td>4.5532</td>
<td>3.12%</td>
</tr>
</tbody>
</table>

Table 4.11. Results for 64 × 64 additional CA patches

<table>
<thead>
<tr>
<th>Model</th>
<th>Avg(ΔH_{avg})</th>
<th>Avg(ΔHS_{avg})</th>
<th>Impr(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>l0</td>
<td>4.6305</td>
<td>4.5227</td>
<td>2.33%</td>
</tr>
<tr>
<td>l0.125</td>
<td>4.6305</td>
<td>4.5416</td>
<td>1.92%</td>
</tr>
<tr>
<td>l0.25</td>
<td>4.6305</td>
<td>4.5229</td>
<td>2.32%</td>
</tr>
<tr>
<td>l0.375</td>
<td>4.6305</td>
<td>4.5487</td>
<td>1.77%</td>
</tr>
</tbody>
</table>

Figure 4.5 shows the Impr results of CA additional testing on different trained models.

It can be found that the Impr result varied from the patch size and overlap ratio, and the best result 3.20% is from 32 × 32 patch size without overlap.

Figure 4.5. Results from CA SR patches based on different trained models
4.1.5. Compared Methods by CA Image

For CA testing set, gravity value and patch quantitative comparisons were compared between our model and the other five methods as well as US testing set. The $\text{Avg}(\Delta H_{\text{avg}})$, $\text{Avg}(\Delta H_{\text{avg}})$ results were average values based on different trained SR models for CA testing set, which were calculated based on different methods and listed in Table 4.12. Our method obtained the best SR patches. Table 4.13 shows the PSNR and SSIM results of different methods on additional CA testing set. Compared with Table 4.7, the observation shows that PSNR and SSIM of CA LR image is much better than US LR image, which also indicate that the difference CA HR and CA LR image is smaller than the difference between US HR and US LR image. As we discussed before, the gravity value also shows the same trend. From Table 4.12, we have observed that our method is better than the other methods in both PSNR and SSIM, which is same as US testing set. It further demonstrated that our method is better.

Table 4.12. Bouguer gravity results of CA from different models

<table>
<thead>
<tr>
<th>Method</th>
<th>LR</th>
<th>SRCNN</th>
<th>VDSR</th>
<th>DRCN</th>
<th>SRResNet</th>
<th>DRRN</th>
<th>Our</th>
</tr>
</thead>
</table>

Table 4.13. Average PSNR/SSIM results of different method on additional CA testing set.

<table>
<thead>
<tr>
<th>Method</th>
<th>LR</th>
<th>SRCNN</th>
<th>VDSR</th>
<th>DRCN</th>
<th>SRResNet</th>
<th>DRRN</th>
<th>Our</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSNR</td>
<td>30.03</td>
<td>30.07</td>
<td>30.08</td>
<td>30.13</td>
<td>30.26</td>
<td>30.29</td>
<td>30.32</td>
</tr>
<tr>
<td>SSIM</td>
<td>0.9499</td>
<td>0.9526</td>
<td>0.9513</td>
<td>0.9499</td>
<td>0.9561</td>
<td>0.9565</td>
<td>0.9583</td>
</tr>
</tbody>
</table>

4.2. Results of Isostatic Gravity

From previous part 4.1 Results of Bouguer Gravity, we know that the model learned by patch $64 \times 64$ with $lp0.125$ is the best. Therefore, isostatic gravity images were cropped into the patch $64 \times 64$ with $lp0.125$ for the training set in this part. Two different models M1 and M2
were learned by only US patches and mixture of US patches and CA patches, respectively. And then the results of these two models were displayed.

4.2.1. Experiment Details

After obtained processed US isostatic gravity HR image and LR image as shown in Figure 3.4, HR and LR images are cropped into patches \((64 \times 64)\). 80% of US patches was chosen randomly as US training set, and the left 20% was for US testing set. For the processed CA isostatic gravity HR image and LR image, the lower cropped area \((320 \times 448)\) was cropped into \((64 \times 64)\) to obtain CA training set. The upper cropped area \((260 \times 320)\) was cropped into \((64 \times 64)\) patches to obtain CA testing set. The mixture of US and CA training set was obtained from mix US training set and CA training set together. Detail information about the patch numbers in training set and testing set for M1 and M2 was listed in Table 4.14. US training set include 233 patches, and there are 35 patches in CA training set. Therefore, there are 268 patches in mixture training set, and US patches were much more than CA patches. From Table 4.14, we find that there were 59 patches and 20 patches in US testing set and CA testing set, respectively.

Table 4.14. Patches number in training set and testing set for M1 and M2

<table>
<thead>
<tr>
<th>Model</th>
<th>US (train set)</th>
<th>CA (train set)</th>
<th>US (test set)</th>
<th>CA (test set)</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td>233</td>
<td>0</td>
<td>58</td>
<td>20</td>
</tr>
<tr>
<td>M2</td>
<td>233</td>
<td>35</td>
<td>58</td>
<td>20</td>
</tr>
</tbody>
</table>
For the M1 and M2 models, the training process parameters are same. Adam algorithm was also adopted to optimize SRResNet+ by minimizing the $L_1$ mean absolute error (MAE) loss function. The learning rate starts from $10^{-4}$, then decreases by half every $5 \times 10^5$ iterations and finally ends once it is smaller than $10^{-7}$.

4.2.2. Results in Isostatic Gravity of US and CA

There are two different models M1 and M2 learned by only US patches, and mixture of US patches and CA patches, respectively. $Avg(\Delta H_{avg}), Avg(\Delta HS_{avg})$ and $Impr$ were also used to evaluate the model M1 and M2, and the results are listed in Table 4.15. For M1, it shows that the $Impr$ on US testing set doesn’t drop from that on training set, which demonstrated that no overfitting in training for but $Impr$ on CA testing set drop half from that on training set. For M2,
Impr on US testing set also doesn’t drop from that on training set, but Impr on CA testing set drop half from that on training set as well as M1. The reason for lower value of CA testing set might be mapping between CA HR and CA LR is different from that mapping between US HR and US LR. In other words, isostatic gravity US data and CA data are inconsistency. Compared M1 and M2, Impr value on the training set for M1 is 15.41%, which is larger than that for M2 11.55%. Impr value on the US testing set for M1 is 16.7%, which also larger than that of M2 13.85%. Generally, increasing the training set size, the result will be better. However, our result is opposite, which further prove that the isostatic gravity US data and CA data are inconsistency. For CA testing set, Impr value for M2 is almost same as that of M1. Therefore, the conclusion can be drawn that M1 model is better than M2.

Table 4.15. Results of M1 and M2 models

<table>
<thead>
<tr>
<th>Model</th>
<th>$\text{Avg}(\Delta H_{\text{avg}})$ (train set)</th>
<th>$\text{Avg}(\Delta R_{\text{avg}})$ (train set)</th>
<th>Impr(%) (train set)</th>
<th>$\text{Avg}(\Delta H_{\text{avg}})$ (US test set)</th>
<th>$\text{Avg}(\Delta R_{\text{avg}})$ (US test set)</th>
<th>Impr(%) (US test set)</th>
<th>$\text{Avg}(\Delta H_{\text{avg}})$ (CA test set)</th>
<th>$\text{Avg}(\Delta R_{\text{avg}})$ (CA test set)</th>
<th>Impr(%) (CA test set)</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td>11.3641</td>
<td>9.6130</td>
<td>15.41%</td>
<td>12.0415</td>
<td>10.0303</td>
<td>16.70%</td>
<td>15.8945</td>
<td>14.5335</td>
<td>8.56%</td>
</tr>
<tr>
<td>M2</td>
<td>11.4329</td>
<td>10.1123</td>
<td>11.55%</td>
<td>12.0415</td>
<td>10.3734</td>
<td>13.85%</td>
<td>15.8945</td>
<td>14.5127</td>
<td>8.69%</td>
</tr>
</tbody>
</table>

4.2.3. Compared methods

Isostatic gravity, PSNR and SSIM were compared between our model M1 and the other five methods including SRCNN [57], VDSR [59], DRCN [61], DRRN [83], SRResNet [26]. Each model of different methods was trained by US training set and then LR patches in US and CA testing set were enhanced by obtained trained SR model. Average isostatic gravity difference between LR and HR: $\text{Avg}(\Delta H_{\text{avg}})$, difference between SR and HR: $\text{Avg}(\Delta H_{\text{avg}})$ for US testing and CA testing set obtained from different methods were listed in Table 4.16. From Table 4.16, we can see clearly that our model M1 and M2 outperform the other five methods for both of US testing set and CA testing set.
PSNR and SSIM results are the average values of SR for US testing set, which are listed in Table 4.17. PSNR and SSIM results for CA testing set are listed in Table 4.18. From both of Table 4.17 and Table 4.18, Our method M1 and M2 outperforms the other methods. We also have observations and analysis as follows. First, SRCNN, VDSR and DRCN did not perform very well. Second, DRRN produces better results than SRResNet, which also observed in other work [83], and our method is better than DRRN, which demonstrated that removing BN layers increase performance in SR [74][75]. All average isostatic gravity, PSNR, and SSIM results demonstrated that our method is comparable method.

Table 4.16. $Avg(\Delta HS_{avg})$ of SR from US testing and CA testing set for different methods

<table>
<thead>
<tr>
<th>Method</th>
<th>LR</th>
<th>SRCNN</th>
<th>VDSR</th>
<th>DRCN</th>
<th>SRResNet</th>
<th>DRRN</th>
<th>M1</th>
<th>M2</th>
</tr>
</thead>
</table>

Table 4.17. Average PSNR/SSIM results of different method on US testing set.

<table>
<thead>
<tr>
<th>Method</th>
<th>LR</th>
<th>SRCNN</th>
<th>VDSR</th>
<th>DRCN</th>
<th>SRResNet</th>
<th>DRRN</th>
<th>M1</th>
<th>M2</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSIM</td>
<td>0.6967</td>
<td>0.7528</td>
<td>0.7551</td>
<td>0.7622</td>
<td>0.7766</td>
<td>0.7769</td>
<td>0.7787</td>
<td>0.7788</td>
</tr>
</tbody>
</table>

Table 4.18. Average PSNR/SSIM results of different method on CA testing set.

<table>
<thead>
<tr>
<th>Method</th>
<th>LR</th>
<th>SRCNN</th>
<th>VDSR</th>
<th>DRCN</th>
<th>SRResNet</th>
<th>DRRN</th>
<th>M1</th>
<th>M2</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSIM</td>
<td>0.6522</td>
<td>0.7078</td>
<td>0.7136</td>
<td>0.7217</td>
<td>0.7254</td>
<td>0.7320</td>
<td>0.7364</td>
<td>0.7328</td>
</tr>
</tbody>
</table>
CHAPTER 5. CONCLUSION

In this work, ground gravity data and satellite gravity data were visualized to high-resolution image and low-resolution image, respectively. The modified ResNet super resolution method was adopted to obtain learned super resolution model, which reconstructed low resolution image into super resolution image.

Based on the bouguer gravity image, SR models learned from US patches with different patch size and overlap ration were studied. The SR learned models showed different Impr value and the one from patch size $64 \times 64$ with overlap $lp0.125$ has the largest Impr value 7.68%. Compared our modified SRResNet method with other five methods, our method outperforms the other methods in gravity value, PSNR and SSIM, which demonstrated our modified SRResNet is an effective method. Even the Impr value 7.68% did not meet our expection, the reason might be the small training set. Compared the result of addition CA testing set with that of US testing set, the SR learned model did not improve the CA testing set very well. The reason might be the inconsistency between US image and CA image.

For the isostatic gravity image, two different SR models M1 and M2 learned from US only patches and mixture of US patches and CA patches were studied, respectively. The results from US testing set and CA testing set were also analyzed. For M1, the Impr value from CA testing set is about 8%, which is much lower than 16% that from US testing set. The M1 obtained higher Impr value 16.7% than 13.85% that from M2. Both above two results demonstrated that the US isostatic gravity and CA isostatic gravity is inconsistency. Compared with other five methods, our M1 also showed the best results in isogravity, PSNR, and SSIM.
The results of bouguer gravity and isostatic gravity turn out that we can fully exploit the advances of DNN-based SISR methods to design and train the super-resolver. Appropriate cropping method is helpful to obtain better SR model.
Reference


Acoust., vol. 29, no. 6, pp. 1153--1160, 1981.


Proceedings of the European conference on computer vision (ECCV) workshops, 2018, pp. 0--0.


[40] E. Freeman, William T and Pasztor, “Markov networks for low-level vision,” *Mitsubishi*


[54] J. CireAan, Dan and Meier, Ueli and Masci, Jonathan and Schmidhuber, “Multi-column


M. Ben naceur, M. Akil, R. Saouli, and R. Kachouri, Fully automatic brain tumor segmentation with deep learning-based selective attention using overlapping patches and


VITA

Name: Yi Zhang, Email: yzhan84@lsu.edu

Profile
My background is in Computer Science and Mechanical Engineering with an emphasis in machine learning and applications. As my passion is centered in this area, I am keen to gain more experience of working in it. My goal is to grow as a person and as a professional, every day.

Education

Master in Computer Science 2018-2021
Louisiana State University, Baton Rouge, LA

Relevant Coursework: Data Structure and Algorithm, Computational Theory, Operating System, Machine Learning, Data Mining and Knowledge Discovery from Datasets.

Thesis: Deep Learning Based Super Resolution of Gravity Data for Geophysical Exploration

Project:
• Visualized satellite gravity data and ground gravity data to obtain low-resolution image and high-resolution image, separately.
• Developed generative adversarial network (GAN) in python for image super-resolution
• Improved low-resolution image to counterpart high-resolution image by GAN
• Performed image quality evaluation by Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM)

PhD in Mechanical Engineering 2012-2017
Louisiana State University, Baton Rouge, LA

Simulation Project:
• Improved sub-modeling procedures on a series of 2D and 3D modeling in computational software ANSYS APDL and Workbench.
• Constructed verification method tuned test problem (TTP) for stress concentration.
• Optimized the design of shoulder fillets of shaft under the bending and torsion.

Material Project:
• Designed and developed novel multi elements alloys via mechanical alloying and arc melting.
• Developed fillers/metal matrix nano-composites by powder metallurgy for enhancing friction coefficient.
• Designed, prepared and characterized metal/polymer composites, and developed their application, including electrical/thermal conductivity.

Skills

Languages: Python, C/C++, Matlab
Operating Systems: HPC, Unix/Linux, MacOS,
Tools: Visual Studio Code, Jupyter Notebook
Software: MATLAB, ANSYS, SolidWorks, Paraview