DATA-DRIVEN AND MACHINE LEARNING APPROACHES FOR SUPPRESSION OF MULTIPLY REFLECTED SEISMIC WAVES

by

Mert Sinan Recep Kiraz

A thesis submitted to the Faculty and the Board of Trustees of the Colorado School of Mines in partial fulfillment of the requirements for the degree of Doctor of Philosophy (Geophysics).

Golden, Colorado

Date ________________________

Signed: ________________________

Mert Sinan Recep Kiraz

Signed: ________________________

Dr. Roel Snieder
Thesis Advisor

Golden, Colorado

Date ________________________

Signed: ________________________

Dr. Paul Sava
Professor and Head
Department of Geophysics

ABSTRACT

Imaging the Earth's interior requires seismic data to be free from the events which reflect more than once in the subsurface (also referred to as *multiples*). For most seismic imaging algorithms, multiples are considered noise since they shadow the useful information about the subsurface, and they need to be removed from the seismic data. However, the removal of multiples remains one of the most challenging tasks in seismic data processing to this day.

The occurrence of multiples depends on a strong impedance contrast between two adjacent subsurface structures and the strength of the multiples increases as the contrast increases. The presence of complex overburden structures (e.g., salt bodies) in the subsurface is one of the reasons for a strong impedance contrast. The solution of the Marchenko equation retrieves the up- and down-going focusing functions which allow one to account for the effects of complex overburden structures and enables the retrieval of the Green's functions for virtual source locations in the subsurface. I propose a two-dimensional (2-D) Marchenko equation where the up- and down-going focusing functions are not needed for the Green's function retrieval. The proposed Marchenko equation not only effectively retrieves the Green's function for a virtual source location in the subsurface, but also forms the basis for obtaining multiple-free images of the subsurface. I also investigate the role of the background velocity model for a successful Green's function retrieval with the Marchenko equation for the reflected and refracted waves. I show that a constant background model retrieves a high-accuracy Green's function as long as the average slowness can be obtained between the virtual source and receiver locations. I also show that the refracted waves can only be incorporated into the retrieved Green's function with a background model that is detailed enough to model the refracted waves during the Marchenko scheme.

The presence of a free surface (or air-water interface) is another reason for a strong impedance contrast. The free-surface effects can be seen in two different ways in the seismic data; (1) free-surface multiples and (2) ghost reflections. Free-surface multiples have at least one downward reflection at the free surface and are the most dominant set of multiples in marine seismic data. Ghost reflections create notches in the frequency content of the seismic data causing the low-frequency content to deteriorate. I develop a Convolutional Neural Network (CNN) (which is an important building block of artificial intelligence) solution to attenuate the free-surface multiples and remove ghost reflections from seismic data, simultaneously. The developed CNN algorithm works on a single seismic trace and does not need to interpolate, extrapolate, or regularize data gaps and near offsets, avoiding the costs and time associated with such procedures. It is designed for use in irregular surveys, such as ocean-bottom node seismic data or time-lapse monitoring studies. I present synthetic examples to demonstrate that the CNN-based solution removes free-surface effects from seismic data. I then apply the developed CNN-based simultaneous free-surface multiple attenuation and ghost removal method on a 2-D field data set acquired in the North Viking Graben. Lastly, I apply the developed CNN-based solution to the simultaneously acquired seismic data (also referred to as *blended data*). I first demonstrate the efficacy of the proposed algorithm on a 2-D synthetic blended data set. Then, I apply the proposed algorithm to a 2-D field data set acquired in the North Sea. The numerical examples demonstrate that the CNN algorithm effectively eliminates free-surface effects on blended data.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

Automatic Gain Control	AGC
Carbon Capture, Utilization and Storage	CCUS
Common Depth Point	CDP
Convolutional Neural Network	CNN
Correlation Coefficient	CC
Deep Learning	DL
Interbed Multiple Prediction	IMP
Inverse Scattering Multiple Attenuation	ISMA
Machine Learning	ML
Mean-Squared Error	MSE
Non-Destructive Testing	NDT
Normal Moveout	NMO
Ocean-Bottom Node	OBN
Physics-Informed Neural Network	PINN
Rectified Linear Unit	ReLU
Surface-Related Multiple Elimination	SRME

ACKNOWLEDGMENTS

First and foremost, I would like to thank my Ph.D. advisor, Roel Snieder, for his continuous help and support during my journey at Mines. He has always encouraged me to go beyond my comfort zone and boldly try new things. His knowledge and wisdom about any scientific topic always amazed me and showed me how little I knew! He has been a dear friend and a great mentor to me and he has always shown his support and care in the most generous way possible. Roel, your speech at my wedding remains unforgettable for everybody! Thank you so much for the great memories you have given me!

I would like to thank my thesis committee members, Dr. Ebru Bozdag, Dr. Leslie Lamberson, Dr. Jeff Shragge, and Dr. Mahadevan Ganesh for kindly serving on my committee and for their insightful comments on my thesis. I give particular gratitude to Ebru for the Turkish coffee and chat sessions we had! In addition, I would like to thank Dr. Ilya Tsvankin for giving me the opportunity to be a TA for his class. Having that two-week experience in his class made it much more fun and productive!

During my Ph.D., I have been privileged to work with two wonderful scientists; Dr. Kees Wapenaar and Dr. Jon Sheiman. Kees, thank you so much for the conversations we had over coffee when you were in Golden, and thank you for your support during my Ph.D. You have helped me grow both personally and professionally. Jon, it's been a great pleasure working with you. You have been a dear friend and a colleague to me over the last 2 years of my Ph.D. I am grateful not only for the fruitful discussions we had about multiples and neural networks but also for the memorable times we spent in Galveston.

I am grateful to the CWP consortium sponsors for making this thesis research topic possible through their financial support. I thank TotalEnergies, one of the sponsor companies of CWP, for giving me the opportunity to spend 2 summers with their EP research and technology team. I also thank Shell, another sponsoring company of CWP, for offering me a great internship and a wonderful start to my full-time career. I am especially indebted to Tim Roden, Bert Michels, Paul Webster, Christian Theriot, and Christian Sutton for making my experience at Shell possible and wonderful.

I also thank Michelle Szobody, Noelle Vance, and Lynn Lundebrek for the behind-the-scenes work they did to help me focus solely on my Ph.D. without worrying about the administrative hiccups. My special thanks go to Diane Witters for her genuine personality and friendship. Diane, thank you for your support and the most delicious orange rolls! I also thank the members of CWP for the fruitful discussions and productive feedback they provided over the last four years of my Ph.D. journey.

Above all, I would like to thank my wife, my mom, my dad, my brother, and my dear friends for their unconditional love and support. My wife has been the greatest source of love and encouragement during my Ph.D. I have been more than lucky to have her on my side. My mom, my dad, and my brother have always shown their love and continuous support from the time I started my graduate school journey back in 2016 to this day. Nothing I have done would have been possible without their support. My dear friends, Erman, Eda, Atakan, Gizem, Efkan, and Güney have always cheered for me and shown me what true friendship looks like. Love you guys!

To my daughter.

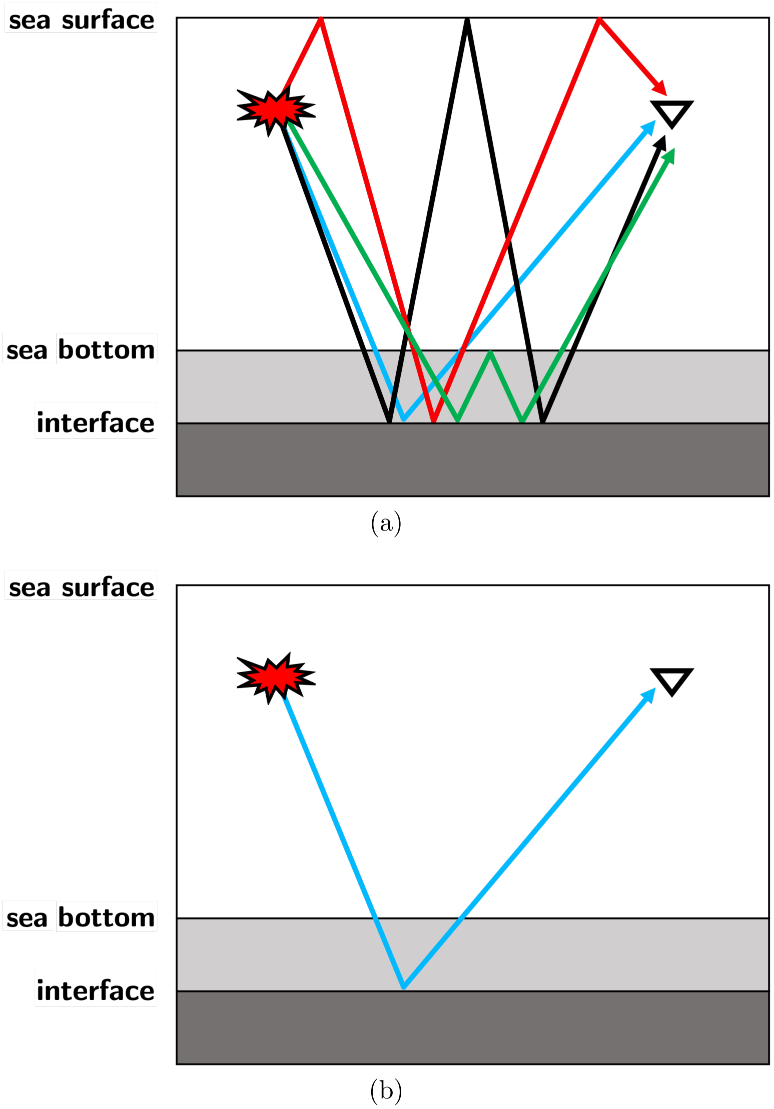
Mommy and Daddy love you so very much!

CHAPTER 1
INTRODUCTION

One of the most challenging tasks in seismic data processing is the removal of multiples from seismic data. A multiple is a wave that has at least one downward reflection somewhere along its ray path. Internal multiples are one of the most commonly encountered multiples in seismic data. Internal multiples have at least one downward reflection at a reflector below the subsurface (Berkhout & Verschuur, 1997); the green line in Figure 1.1(a) illustrates an example of an internal multiple for a marine seismic acquisition where the red star denotes a source located at depth near the sea surface, and the white triangle denotes a receiver also located at depth near the sea surface.

Marchenko redatuming and imaging, a novel approach for generating multiple-free seismic images using surface seismic data and a smooth velocity model, has the potential to resolve problems related to internal multiples (Broggini & Snieder, 2012; Wapenaar *et al.*, 2014). Marchenko redatuming removes the effects of complex overburden structure by creating virtual responses (also referred to as wavefield focusing) using the Marchenko equation (Marchenko, 1955). Marchenko imaging creates artifact-free images of the subsurface by applying an imaging condition. Marchenko redatuming and imaging eliminates the effects of internal multiples and produces more accurate images of the subsurface (Behura *et al.*, 2014; Jia *et al.*, 2018, 2021; Ravasi *et al.*, 2016) compared to the images obtained by the reverse-time migration (Baysal *et al.*, 1983; McMechan, 1983).

The relationship between wavefield focusing and the Marchenko equation was established by Rose (2001, 2002b) allowing for wavefield focusing in an unknown medium once the Marchenko equation is solved. Broggini & Snieder (2012) demonstrate the retrieval of the Green's function between any subsurface point and surface acquisition point without the need for a physical receiver at the virtual source location, using only one-sided illumination. Wapenaar *et al.* (2013) examine the retrieval of the three-dimensional Green's function and provide an example of the two-dimensional retrieval.



(a)

(b)

Figure 1.1 (a) Schematic illustration of the ray paths of primary reflection (blue line), internal multiple (green line), free-surface multiple (black line), and source and receiver side ghost reflections (red line). The white triangle denotes a receiver located at depth and the red shape denotes a source located at depth *before* multiple and ghost removal. (b) The ray paths in (a) *after* multiple and ghost removal.

A comprehensive explanation and numerical implementation of the Marchenko redatuming and imaging method is provided by Wapenaar *et al.* (2014), van der Neut *et al.* (2015), Thorbecke *et al.* (2017), and Lomas & Curtis (2019). Marchenko redatuming and imaging has been widely utilized for various purposes including internal multiple elimination (Meles *et al.*, 2015, 2016; Thorbecke *et al.*, 2021), elastic wave applications (da Costa Filho *et al.*, 2014, 2015; Wapenaar, 2014), subsurface imaging and for comparisons with the conventional imaging results (e.g., reverse time migration) (Behura *et al.*, 2014; Jia *et al.*, 2018; Ravasi *et al.*, 2016; Wapenaar *et al.*, 2014). Marchenko redatuming and imaging has been applied in various field data sets including imaging of a North Sea field data set (Ravasi *et al.*, 2016), target-oriented subsalt imaging of the Gulf of Mexico data set (Jia *et al.*, 2018), time-lapse monitoring of the Frio carbon sequestration data set (Kiraz & Nowack, 2018), multiple suppression on an Arabian Gulf field data set (Staring *et al.*, 2021), and an offshore Brazil data set for imaging a reservoir under basalt (Jia *et al.*, 2021).

Multiples are generated when waves encounter strong impedance contrasts. In marine seismic data, the main sources of impedance contrast are the free surface (air-water interface) and the water bottom. The water bottom may have weak impedance contrast (e.g., weak impedance contrast), but the free surface is always an almost-perfect reflector (e.g., strong impedance contrast), producing two main problems in recorded seismic data: (1) free-surface multiples and (2) ghost reflections.

Multiple reflections in recorded marine seismic data pose a significant challenge as they interfere with the primary reflections. Free-surface multiples, with at least one downward reflection at the sea surface (black line in Figure 1.1(a)), are particularly critical to address. Surface-Related Multiple Elimination (SRME) is a data-driven method that predicts multiples by utilizing the relationship between multiples and their sub-events (Verschuur *et al.*, 1992). The SRME technique has gained widespread use and has established itself as the primary and industry-accepted solution for eliminating free-surface multiples (Ma *et al.*, 2019). It requires a dense and regular source and receiver distribution, a known source wavelet, interpolated data, and near-offset data, but the method does not need subsurface information. The SRME method consists of data pre-processing, multiple prediction, wavelet estimation through matching filtering, and adaptive subtraction (Dragoset *et al.*, 2010; Verschuur *et al.*, 1992). Despite being widely used and one of the most effective tools in seismic processing, SRME is computationally intensive and does not always guarantee optimal results. During the subtraction process, there is a risk of removing both primary and multiple reflections, as they can overlap (Dragoset *et al.*, 2010). While two-dimensional (2-D) SRME can be more computationally efficient, it may not handle complex subsurface structures requiring three-dimensional (3-D) source and receiver distribution effectively, especially in the presence of strong cross-line data components, making this even harder to achieve in 3-D surveys.

Several efforts have been made to tackle free-surface multiples. One of them is presented by Riley & Claerbout (1976) who discusses the theory and application of a one-dimensional (1-D) free-surface multiple attenuation algorithm. Verschuur *et al.* (1992) highlight the technique of predicting and subtracting free-surface multiples from seismic data. Jakubowicz (1998) describes a method for eliminating interbed multiples, which is an extension of SRME. There are numerous examples of field data applications of the SRME method (Dragoset & MacKay, 1993; Dragoset & Jeričević, 1998; Kelamis & Verschuur, 2000; Verschuur *et al.*, 1995).

Ghost reflections cause notches in seismic data frequency due to destructive interference of the downward radiated wave with the wave reflected off the free surface, and alter the source signature (Amundsen & Zhou, 2013). Attenuating ghost reflections (or deghosting) is crucial for broadband processing and full-waveform inversion (Kragh *et al.*, 2010; Virieux & Operto, 2009). Ghost reflections in seismic data appear as delayed primary signals with opposite polarity (Dondurur, 2018). There are three types of ghosts recorded in seismic data: (1) source ghost, (2) receiver ghost, and (3) source and receiver ghost (shown as red line in Figure 1.1(a)). Deghosting can be done through acquisition-based solutions such as slanted streamer, over-under steamers, and dual-sensor streamer acquisitions (Carlson *et al.*, 2007; Moldoveanu *et al.*, 2007; Soubaras & Whiting, 2011) or processing-based solutions such as low-frequency deghosting filtering and joint deconvolution (Amundsen & Zhou, 2013; Soubaras, 2010). Recent advancements have also applied machine learning techniques to remove ghost reflections (Almuteri & Sava, 2023; Vrolijk & Blacquière, 2021).

Seismic migration methods (Baysal *et al.*, 1983; McMechan, 1983) presume that seismic data only consists of primary reflections (the blue line in Figure 1.1). It is, hence, crucial for the seismic data to be free of free-surface-generated events (represented by the red and black lines in Figure 1.1(a)) that are reflected at least twice off reflectors in the subsurface. A profound limitation of the existing algorithms that remove free-surface multiples and ghost reflections is that they require a dense source and receiver distribution over a seismic survey. These algorithms are sensitive to missing data gaps and it is computationally demanding to fill these gaps in the data (e.g., interpolation, extrapolation, regularization). For example, for a successful implementation of the SRME algorithm, shot and receiver spacings along inline and crossline directions should be around tens of meters (Dragoset *et al.*, 2010). These specifications are unfeasible in reality, so some form of interpolation, extrapolation, or regularization is required for the SRME method.

In recent years, the geophysics community has adapted a novel approach called Convolutional Neural Networks (CNNs) to address various problems like SRME. A CNN is a specific type of artificial intelligence commonly used for image recognition, natural language processing, and for tasks that require an understanding of visual data (Ekman, 2021; Géron, 2019). CNNs use convolutional layers to extract features from the input image. Network parameters are learned through training on a dataset, where the network is optimized to minimize the difference between the predicted and actual output. The resulting network can then be applied to new images to make predictions, which can be improved through further training (Bishop, 2006; Ekman, 2021; Géron, 2019; Murphy, 2012).

CNNs have been used to address various problems in the exploration seismology such as obtaining an elastic subsurface model using recorded normal-incidence seismic data (Das *et al.*, 2019), 3-D seismic fault segmentation (Wu *et al.*, 2019), seismic interpretation (Di *et al.*, 2020), seismic deghosting (Almuteri & Sava, 2023), deblending the simultaneous source seismic data (Cheng *et al.*, 2022), magnetotelluric noise suppression (Li *et al.*, 2023), full-waveform inversion (Wu & McMechan, 2019), microseismic waveform classification and arrival picking (Zhang *et al.*, 2020), velocity model building (Alzahrani & Shragge, 2022), random noise attenuation (Si *et al.*, 2019), sparse least-square migration (Liu *et al.*, 2020), and free-surface multiple attenuation (Ovcharenko *et al.*, 2021; Siahkoohi *et al.*, 2018, 2019).

The central objective of this thesis is to design algorithms that eliminate multiples and ghost reflections from seismic data while circumventing the need for a dense source and receiver distribution. To achieve this goal, I first develop a Marchenko equation that, unlike the previous algorithms, does not require decomposing the focusing function into upgoing and downgoing components to accurately retrieve the Green's functions for virtual source locations in the subsurface. The algorithm that encodes the new Marchenko equation forms the basis for obtaining images that are free of multiples caused by the complex overburden structures in the subsurface (e.g., the green line in Figure 1.1(a)). I then perform a sensitivity analysis to background models of the Marchenko algorithm for the reflected and refracted waves. To tackle free-surface multiples and ghost reflections in seismic data (e.g., the red and black lines in Figure 1.1(a)), I next develop a CNN-based algorithm. In contrast to SRME, the proposed algorithm operates on a single trace at a time and unlike the existing methods, can thus be used for surveys with sparse source and receiver distributions. The CNN-based algorithm, therefore, does not require any interpolation/extrapolation/regularization step and can be applied to surveys with irregular acquisition such as ocean-bottom node seismic data or time-lapse monitoring studies. I also apply the proposed CNN-based methodology to a 2-D data set acquired in the North Sea. Finally, I extend the proposed CNN method for the simultaneous source acquisition surveys. I present synthetic and field data applications to show that the

CNN-based free-surface multiple attenuation and deghosting algorithm effectively removes free-surface effects on blended data, and generalizes well for the field data.

1.1 Thesis Outline

The chapters in this thesis are based on publications. One chapter has been published in a peer-reviewed journal, three chapters have been submitted for publication, and one chapter will be submitted for publication. As the first author of the following technical chapters, I developed the theory and performed numerical computations. As the thesis advisor and coauthor of the following technical chapters, Dr. Roel Snieder supervised the analytical and numerical findings. As a coauthor for the first two chapters, Dr. Kees Wapenaar supervised the analytical and numerical findings. As a coauthor for the last three chapters, Dr. Jon Sheiman supervised the analytical and numerical findings. The authors discussed the results and contributed to writing the manuscripts and documenting the research work.

In Chapter 2, entitled "Focusing waves in an unknown medium without wavefield decomposition", I present a novel Marchenko equation that does not require the up- and down-going components of the focusing function to effectively retrieve the Green's function for a virtual source location in the subsurface. The outcomes of this chapter were presented at a Society of Exploration Geophysicists (SEG) Annual Meeting, and were published in *JASA Express Letters*:

- Kiraz, M. S. R., R. Snieder, and K. Wapenaar, 2020, Marchenko focusing without up/down decomposition: 90th Annual International Meeting, SEG, Expanded Abstracts, 3593-3597, doi: 10.1190/segam2020-3423471.1.

- Kiraz, M. S. R., R. Snieder, and K. Wapenaar, 2021, Focusing waves in an unknown medium without wavefield decomposition: JASA Express Letters, 1, 055602, doi: 10.1121/10.0004962.

In Chapter 3, entitled "The role of the background velocity model for the Marchenko focusing of reflected and refracted waves", I describe the role of the background model for an accurate Green's function retrieval using the Marchenko equation. I present different background models and investigate the accuracy of the retrieved Green's functions using these models. I also show that the background velocity model, which is used to model the direct wave for the Marchenko algorithm, must be sufficiently detailed to produce refracted waves if these waves are to be recovered. The outcomes of this chapter were presented at the First International Meeting for Applied Geoscience & Energy and have been submitted for peer review in *Geophysical Journal International*:

- Kiraz, M. S. R., R. Snieder, and K. Wapenaar, 2021, Marchenko without up/down decomposition on the Marmousi model and retrieval of the refracted waves: Are they caused by the Marchenko algorithm?: First International Meeting for Applied Geoscience & Energy, SEG/AAPG, Expanded Abstracts, 3280-3284, doi: 10.1190/segam2021-3583530.1.
- Kiraz, M. S. R., R. Snieder, and K. Wapenaar, 2023, The role of the background velocity model for the Marchenko focusing of reflected and refracted waves: Geophysical Journal International (under review).

In Chapter 4, entitled "Removing free-surface effects from seismic data using convolutional neural networks – Part 1: Theory and sensitivity analysis", I propose a novel CNN-based approach to simultaneously attenuate free-surface multiples and remove ghost reflections from seismic data. The proposed algorithm works on a trace-by-trace basis and makes the algorithm applicable for surveys with sparse source and receiver distributions (such as OBN data or time-lapse monitoring data). I analyze the sensitivity of the CNN-based free-surface multiple attenuation and ghost removal by presenting different numerical examples. The outcomes of this chapter have been submitted to *Geophysical Prospecting*:

- Kiraz, M. S. R., J. Sheiman, and R. Snieder, 2023, Removing free-surface effects from seismic data using convolutional neural networks – Part 1: Theory and sensitivity analysis: Geophysical Prospecting (under review).

In Chapter 5, entitled "Removing free-surface effects from seismic data using convolutional neural networks – Part 2: An application to the Mobil AVO Viking Graben data set", I illustrate the efficacy of the CNN-based free-surface multiple attenuation and ghost removal presented in Chapter 4 on the Mobil AVO Viking Graben field data set acquired in the North Sea. The outcomes of this chapter have been submitted for publication in *Geophysical Prospecting*:

- Kiraz, M. S. R., J. Sheiman, and R. Snieder, 2023, Removing free-surface effects from seismic data using convolutional neural networks – Part 2: An application to the Mobil AVO Viking Graben data set: Geophysical Prospecting (under review).

In Chapter 6, entitled "Free-surface multiple attenuation and seismic deghosting for blended data using convolutional neural networks", I extend the CNN-based free-surface multiple attenuation and ghost removal presented in Chapter 4 to the simultaneously acquired seismic data (or blended data). I present synthetic examples and show that the CNN-based algorithm removes free-surface effects for blended data. Additionally, I blend the Mobil AVO Viking Graben data set and present a blended field data set example for free-surface multiple attenuation and ghost removal. The outcomes of this chapter will be submitted for publication in *Geophysics*:

- Kiraz, M. S. R., J. Sheiman, and R. Snieder, 2023, Free-surface multiple attenuation and seismic deghosting for blended data using convolutional neural networks: Geophysics (to be submitted).

I also propose a CNN-based algorithm for the Green's function retrieval in 1-D. I illustrate the efficacy of the CNN-based method on 1-D synthetic data set. This method shows that once the CNN is trained to retrieve the Green's function for a certain virtual source location, we do not need to solve the Marchenko equation for wavefield focusing. The outcomes of this work were presented at the Second International Meeting for Applied Geoscience & Energy:

- Kiraz, M. S. R. and R. Snieder, 2022, Marchenko focusing using convolutional neural networks: Second International Meeting for Applied Geoscience & Energy, SEG/AAPG, Expanded Abstracts, 1930-1934, doi: 10.1190/image2022-3738167.1.

In Chapter 7, I present the general conclusions and propose future recommendations on various applications of the presented methods in this thesis.

CHAPTER 2
FOCUSING WAVES IN AN UNKNOWN MEDIUM WITHOUT WAVEFIELD DECOMPOSITION

A paper published[1] in *JASA Express Letters*

Mert Sinan Recep Kiraz[2,3,4], Roel Snieder[3] & Kees Wapenaar[5]

The Gel'fand-Levitan equation, the Gopinath-Sondhi equation, and the Marchenko equation are developed for one-dimensional inverse scattering problems. Recently, a version of the Marchenko equation based on wavefield decomposition has been introduced for focusing waves in multi-dimensions. However, wavefield decomposition is a limitation when waves propagate horizontally at the focusing level. Here, the Marchenko equation for focusing without wavefield decomposition is derived, and by iteratively solving the Marchenko equation, the Green's function for an arbitrary location in the medium is retrieved from the scattered waves recorded on a closed receiver array and an estimate of the direct-wave without wavefield decomposition.

2.1 Introduction

Inverse scattering (Chadan & Sabatier, 1989; Colton & Kress, 1998; Gladwell, 1993) uses scattered waves to determine the scattering properties of a medium. Burridge (1980) shows that the Gel'fand-Levitan equation and the Gopinath-Sondhi equation have the same structure as the Marchenko equation, and shows that the Marchenko equation can be used for medium reconstruction (Burridge, 1980; Newton, 1980a). The solution of the

[1] Reproduced from *JASA Express Lett.* **1**, 055602 (2021); https://asa.scitation.org/doi/10.1121/10.0004962 with the permission of *The Acoustical Society of America.*
[2] Primary researcher and author.
[3] Author for correspondence. Direct correspondence to mertkiraz@gmail.com.
[4] Center for Wave Phenomena, Colorado School of Mines, 1500 Illinois St., Golden, CO 80401, USA.
[5] Department of Geoscience and Engineering, Delft University of Technology, P.O. Box 5048, Delft, GA 2600, The Netherlands.

one-dimensional (1-D) Marchenko equation is an exact integral equation to make the connection between the scattered data and the scatterer potential. Rose (2001, 2002a) defines focusing as finding an incident wave that becomes a delta function at a prescribed focus location and time inside the medium. He shows that this incident wave follows from the scattered data and uses the Marchenko equation for 1-D inverse scattering problems. Broggini & Snieder (2012) utilize Rose's approach and introduce a scheme in 1-D to retrieve the Green's function containing single-scattered and multiply-scattered waves of the inhomogeneous medium. Wapenaar *et al.* (2012) show the virtual source creation in two dimensions using the recorded data but the proposed method excludes horizontally propagating energy at the virtual source level. Wapenaar *et al.* (2013, 2014) derive the three-dimensional Marchenko equation for wavefield focusing and, therefore, for the Green's function retrieval; however, their solution requires up/down decomposition of the wavefield, which also excludes horizontally propagating energy at the focusing level. This is a limitation when the medium has steeply dipping structures because the horizontally scattered waves and refracted waves cannot be fully represented with the up/down decomposition. Recently, there have been several studies to address the limitation of the Marchenko method due to the up/down separation of the Marchenko equation. Kiraz *et al.* (2020) show wavefield focusing for an arbitrary point inside an unknown highly scattering inhomogeneous medium using the data acquired on a closed boundary. Diekmann & Vasconcelos (2021) and Wapenaar *et al.* (2021) present alternative approaches to Green's function retrieval without up/down decomposition each with their own pros and cons.

The Marchenko schemes proposed in 1-D provide an exact solution for focusing, and for Green's function retrieval in the medium. Green's function retrieval is of importance for imaging applications in many fields. The ability to focus waves opens up applications ranging from scattering kidney stones to performing imaging, monitoring in seismology, and non-destructive testing.

In this paper, we propose a two-dimensional (2-D) Marchenko equation for focusing waves in a highly scattering inhomogeneous medium. We show that by iteratively solving the Marchenko equation, the Green's function for an arbitrary point in a strongly scattering inhomogeneous medium can be retrieved without wavefield decomposition at the focal point. As opposed to the current Marchenko algorithms that use single-sided acquisition methods, we include waves propagating in all directions at the focal point using the contributions from a closed array. Our scheme is an extension of the 1-D Marchenko algorithms proposed by Newton (1980a), Rose (2001, 2002a), and Broggini & Snieder (2012) into 2-D.

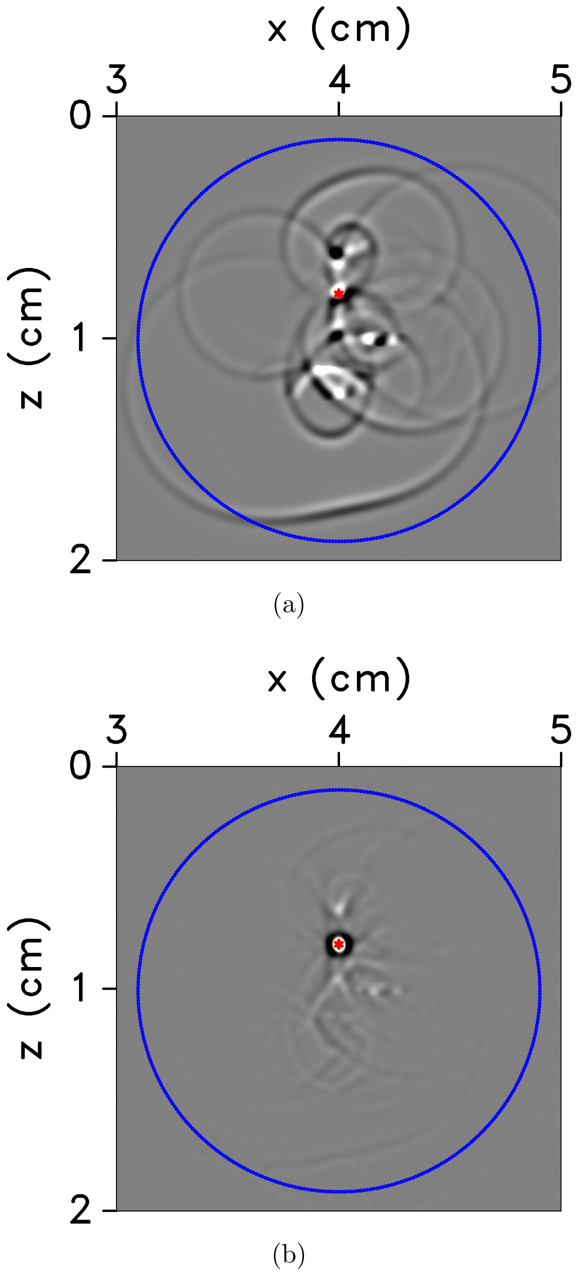
2.2 Theory

Consider the acoustic wave equation

$$\rho\nabla\cdot\left(\frac{1}{\rho}\nabla p\right)+\frac{\omega^2}{c^2}p=f\ , \tag{2.1}$$

where ρ is density, ω is the angular frequency, f is the source term, p is pressure, and c is the velocity. We use the acoustic wave equation for a constant velocity and variable density in the numerical examples in this paper.

We define the Green's function, $G(\mathbf{x},\mathbf{x}_s,t)$, as the solution to the wave equation $LG=\delta(\mathbf{x}-\mathbf{x}_s)\delta(t)$, with the differential operator $L=\rho\nabla\cdot(\rho^{-1}\nabla)-c^{-2}\partial^2/\partial t^2$. Here, $\mathbf{x}_s$ is the source location and the Green's function is the response to a source at $\mathbf{x}_s$ recorded at the receiver location $\mathbf{x}$. We use the following convention for the Fourier transform: $f(t)=\frac{1}{2\pi}\int F(\omega)exp(-i\omega t)d\omega$, where i is the imaginary unit. In the frequency domain $G(\mathbf{x},\mathbf{x}_s,\omega)$ satisfies $LG=\delta(\mathbf{x}-\mathbf{x}_s)$, with the differential operator $L=\rho\nabla\cdot(\rho^{-1}\nabla)+\omega^2/c^2$.



(a)

(b)

Figure 2.1 (a) Snapshot at $t = 0$ of the time-derivative of the time-reversed modeled direct-wave injection. (b) Snapshot at $t = 0$ of the time-derivative of the time-reversed retrieved homogeneous Green's function injection obtained from our iterative Marchenko algorithm. The red asterisk denotes the focal point $\mathbf{x}_s$ and the blue line represents the transducer locations.

We describe an iterative solution to focus a wavefield in the medium to a pre-defined location at $t = 0$ when injected into the medium. Our solution requires the direct-wave information modeled in the homogeneous medium (when ρ and c are constant) for a source at the focusing point $\mathbf{x}_s$. This is the known Green's function $G_0(\mathbf{x}, \mathbf{x}_s, t)$ in a homogeneous medium. Sending this direct-wave back into the inhomogeneous medium from a circular receiver array with the radius R in a time-reversed order creates a focus at the focal point at $t = 0$; however, in addition to the focal spot, other waves are present around the focusing point, and Figure 2.1(a) shows the snapshot at $t = 0$ of the time-reversed direct-wave injection into the heterogeneous medium shown in Figure 2.2 (about which the details will be provided in the following section). This shows that emitting the time-reversed direct-wave into an inhomogeneous medium does not restrict the focused field to the focusing point, and our goal is to remove the waves at other locations than the focusing point in Figure 2.1(a). Figure 2.1(b) shows the snapshot at $t = 0$ of the wavefield injection obtained by the iterative algorithm we propose. As shown in Figure 2.1(b), our algorithm creates a wavefield that focuses to the pre-defined focal point, which acts as a virtual source, and suppresses other waves at $t = 0$. We obtain our focusing wavefield by only using the direct-wave information modeled in the homogeneous medium (when ρ and c are constant) and the recorded scattering response. Unlike the conventional Marchenko methods, our method does not require the decomposition of the Marchenko equation to achieve focusing. In the following section, we discuss the iterative Marchenko equation we propose for wavefield focusing and show how to obtain better focuses in the medium than one can achieve with the direct waves only.

2.2.1 Iterative Scheme and The Marchenko Equation

We define the ingoing wavefield, $U^{in}(\hat{\mathbf{n}}', t)$, and outgoing wavefield, $U^{out}(\hat{\mathbf{n}}, t)$, where $\hat{\mathbf{n}}'$ and $\hat{\mathbf{n}}$ denote the locations on the circle with radius R; they are related via the scattering response $A(\hat{\mathbf{n}}, \hat{\mathbf{n}}', t)$ of the inhomogeneous medium. Following Rose (2001), Rose (2002a), Broggini & Snieder (2012), Wapenaar *et al.* (2013), and Wapenaar *et al.* (2014), we design a wavefield that becomes a delta function at the focus location with an iterative scheme that relates the ingoing wave U_k^{in} to the outgoing wave U_k^{out} at iteration k as

$$U_k^{out}(\hat{\mathbf{n}}, t) = \oint \int A(\hat{\mathbf{n}}, \hat{\mathbf{n}}', t-\tau) U_k^{in}(\hat{\mathbf{n}}', \tau) d\tau dn' \; . \tag{2.2}$$

The ingoing and outgoing waves and the scattering operator are defined on a circle with radius R, but for brevity, we omit the parameter dependence on R in equation (2.2).

The iterative scheme starts with injecting a delta function into the medium and the ingoing wavefield for the first iteration gives

$$U_0^{in}(\hat{\mathbf{n}}', \tau) = \delta(\tau + t_d(\hat{\mathbf{n}}')) \; , \tag{2.3}$$

where $t_d(\hat{\mathbf{n}}')$ is the arrival time of the direct waves that propagates from the focusing point to the point $\hat{\mathbf{n}}'$ on the circle.

Following Broggini & Snieder (2012), the purpose of the iterative scheme is to reconstruct a wavefield that after interacting with the heterogeneities in the medium collapses onto a delta function at the focusing point at $t = 0$. We create a symmetric field in time for $-t_d(\hat{\mathbf{n}}') < t < t_d(\hat{\mathbf{n}}')$. We later show that the symmetry in time leads to focusing. To achieve the symmetry for the iterative scheme, we define the ingoing wavefield as

$$U_k^{in}(\hat{\mathbf{n}}', \tau) = U_0^{in}(\hat{\mathbf{n}}', \tau) - \Theta(\hat{\mathbf{n}}', \tau) U_{k-1}^{out}(\hat{\mathbf{n}}, -\tau) \; , \tag{2.4}$$

where $\Theta(\hat{\mathbf{n}}', \tau)$ is a window function and defined as $\Theta(\hat{\mathbf{n}}', \tau) = 1$ when $-t_d(\hat{\mathbf{n}}') < \tau < t_d(\hat{\mathbf{n}}')$, and otherwise $\Theta(\hat{\mathbf{n}}', \tau) = 0$.

When the iterative scheme converges (hence when $U_k^{out} = U_{k-1}^{out}$), the iteration number can be dropped. Inserting equation (2.4) into equation (2.2) then gives

$$\begin{aligned} U^{out}(\hat{\mathbf{n}}, t) = & \oint \int A(\hat{\mathbf{n}}, \hat{\mathbf{n}}', t-\tau) U_0^{in}(\hat{\mathbf{n}}', \tau) d\tau dn' \\ & - \oint \int_{-t_d^\epsilon}^{t_d^\epsilon} A(\hat{\mathbf{n}}, \hat{\mathbf{n}}', t-\tau) U^{out}(\hat{\mathbf{n}}', -\tau) \; d\tau dn' \; , \end{aligned} \tag{2.5}$$

with $t_d^\epsilon = t_d - \epsilon$ where we introduce ϵ as a small positive constant to exclude the direct-wave at t_d. If we define $K = -U^{out}$, and substitute this into equation (2.5) using equation (2.3), we obtain

$$
\begin{aligned}
&K(\hat{\mathbf{n}}, t) + \oint A(\hat{\mathbf{n}}, \hat{\mathbf{n}}', t + t_d(\hat{\mathbf{n}}'))dn' \\
&+ \oint \int_{-t_d^\epsilon}^{t_d^\epsilon} A(\hat{\mathbf{n}}, \hat{\mathbf{n}}', t - \tau) K(\hat{\mathbf{n}}', -\tau)\ d\tau dn' = 0\ . \qquad (2.6)
\end{aligned}
$$

Burridge (1980) shows that the 1-D Marchenko equation, Gel'fand-Levitan equation, and the Gopinath-Sondhi equations of inverse scattering can be written in symbolic notation as $K + R + \int_W RK = 0$ where $\int_W$ shows the time interval, R is the recorded data, and K is the function we solve for. Equation (2.6) has the same structure as the equations derived by Burridge (1980) and, therefore, gives a 2-D Marchenko equation without using up/down decomposition. Equation (2.6) also has a similar relation with the equations derived by Newton (1980b, 1981, 1982) using the scattering data in multi-dimensional media.

2.3 Numerical Example and Green's Function Retrieval

We illustrate our method with a 2-D numerical example. Figure 2.2 shows the source and receiver geometry of a 2-D acoustic medium. The red asterisk in Figure 2.2 denotes the virtual source location and the blue line represents a circle on which 400 equidistant sources and receivers are placed. The virtual source location, $\mathbf{x}_s = (x, z)$, is at $x = 4$ cm and $z = 0.8$ cm. The medium has a constant background velocity and density, $c_0 = 2$ km/s and $\rho_0 = 2$ g/cm^3, respectively. Figure 2.2 also shows four different elliptical-shaped scatterers located in the medium with densities $\rho_1 = 4.5$ g/cm^3, $\rho_2 = 5$ g/cm^3, $\rho_3 = 7.5$ g/cm^3, $\rho_4 = 6$ g/cm^3, respectively. We use finite-difference modeling with absorbing boundaries and the source wavelet is a Ricker wavelet (Ricker, 1953) with a central frequency of 2 MHz. A challenge of the used geometry is that the focusing point is located inside one of the scatterers, which has a reflection coefficient of about 40% at the boundaries. As a result, the source generates strong reverberations within the scatterer.

The ingoing wavefield in the finite-difference modeling can be implemented by either changing the finite-difference stencil at the circular array, or by using the equivalent sources f (in equation form) in the acoustic wave equation (2.1) to produce the desired ingoing wavefield. We use the equivalent sources in the acoustic wave equation (2.1) for the finite-difference implementation where the equivalent sources are given by the normal derivative of the ingoing wavefield (see Appendix A). To solve the Marchenko equation iteratively, we start with $U_0^{in}(\hat{\mathbf{n}}', t) = U_d(\hat{\mathbf{n}}', -t)$ where U_d is the time-reversed direct-wave in the homogeneous background medium. We send the ingoing wave U_0^{in} from the receiver array into the medium and use the outgoing wave recorded at the array in equation (2.4) to determine the ingoing wave for the next iteration. We use seven iterations to get close to convergence but more iterations might be needed for more complicated media where velocity and density are varying.

We next inject the wavefield obtained by the iterative solution on the boundary. Figure 2.3(a) shows the total wavefield, $U_{total}(\hat{\mathbf{n}}', t) = U^{in}(\hat{\mathbf{n}}', t) + U^{out}(\hat{\mathbf{n}}', t)$, recorded on the boundary for the 7th iteration, which consists of the superposition of the ingoing and outgoing wavefield. The wavefield in Figure 2.3(a) is symmetric in time for $-t_d(\hat{\mathbf{n}}') < t < t_d(\hat{\mathbf{n}}')$ (approximately between -5 μs and 5 μs). If we take the difference between the total wavefield in Figure 2.3(a) and its time-reversed version, i.e., $U_{total}(\hat{\mathbf{n}}', t) - U_{total}(\hat{\mathbf{n}}', -t)$, all events in the interval $-t_d(\hat{\mathbf{n}}') < t < t_d(\hat{\mathbf{n}}')$ vanish as shown in Figure 2.3(b). A small amount of energy remains in Figure 2.3(b) for $-t_d(\hat{\mathbf{n}}') < t < t_d(\hat{\mathbf{n}}')$, this is due to numerical inaccuracies in our solution of the Marchenko equation. Since $U_{total}(\hat{\mathbf{n}}', t) - U_{total}(\hat{\mathbf{n}}', -t)$ is anti-symmetric in time, it vanishes for $t = 0$, also after injecting it into the medium. Hence we diagnose the focusing by showing the time derivative $\frac{\partial}{\partial t}(U_{total}(\hat{\mathbf{n}}', t) - U_{total}(\hat{\mathbf{n}}', -t))$, injected into the medium.

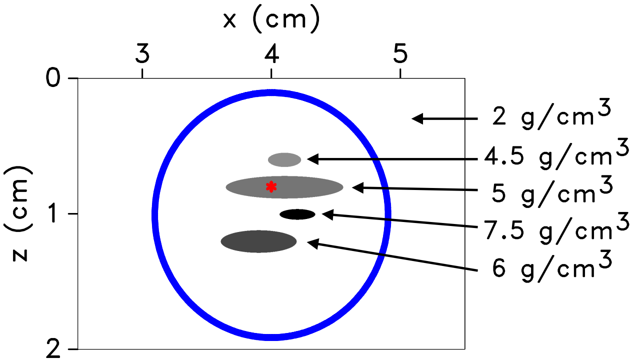


Figure 2.2 Geometry of the 2-D model. Sources and receivers (400 each) are located on the blue circle, and the red asterisk shows the virtual source location $\mathbf{x}_s$. The elliptical scatterers have contrasting densities which are given on the right hand side.

Figure 2.3(b) shows that for positive times, the wavefield $U_{total}(\hat{\mathbf{n}}', t) - U_{total}(\hat{\mathbf{n}}', -t)$ vanishes at the receivers for $t < t_d(\hat{\mathbf{n}}')$. If we consider this wavefield at $t = 0$, the direct waves radiated at $t = 0$ from $\mathbf{x}_s$ arrive at a receiver location $R(\hat{\mathbf{n}}')$ at $t_d(\hat{\mathbf{n}}')$. Suppose that waves would radiate at $t = 0$ from a point $\mathbf{x} \neq \mathbf{x}_s$. For some receivers, those waves would arrive at a time $t < t_d(\hat{\mathbf{n}}')$; however, as shown in Figure 2.3(b), no waves arrive at time $t < t_d(\hat{\mathbf{n}}')$. This means that waves do not radiate from any point $\mathbf{x} \neq \mathbf{x}_s$ at $t = 0$. Therefore, the time-derivative of the wavefield $U_{total}(\hat{\mathbf{n}}', t) - U_{total}(\hat{\mathbf{n}}', -t)$, injected into the medium, is only non-zero at $t = 0$ at the point $\mathbf{x}_s$, and the wavefield focuses at $t = 0$ at the virtual source location (see also, Appendix B).

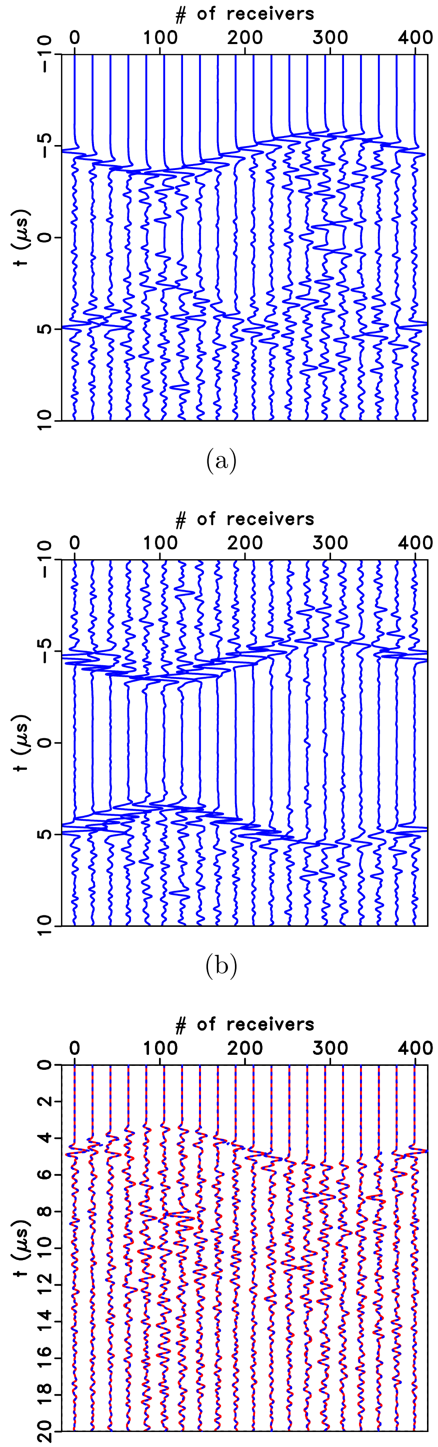
We let $p(\mathbf{x}, t)$ denote the total wavefield in the interior that is associated with the wavefield $U_{total}(\hat{\mathbf{n}}', t)$ on the boundary, and $p(\mathbf{x}, -t)$ denote the time-reversed version of this wavefield. The homogeneous Green's function ($G_h(\mathbf{x}, \mathbf{x}_s, t) = G(\mathbf{x}, \mathbf{x}_s, t) - G(\mathbf{x}, \mathbf{x}_s, -t)$) (Oristaglio, 1989), for the virtual source location $\mathbf{x}_s$ and the receiver location $\mathbf{x}$ is, up to a multiplicative constant, obtained from (see also Appendix B)

$$G_h(\mathbf{x}, \mathbf{x}_s, t) = p(\mathbf{x}, t) - p(\mathbf{x}, -t) \ . \tag{2.7}$$

If we want to focus a wavefield at the virtual source location where there is no actual source located, we must have a non-zero incident wavefield. The causal and acausal Green's functions satisfy the inhomogeneous acoustic wave equation, but the homogeneous Green's function G_h satisfies the homogeneous wave equation (Oristaglio, 1989). Equation (2.7), therefore, retrieves the Green's function for $t > 0$ for the virtual source location $\mathbf{x}_s$. Unlike other (interferometric) Green's function retrieval methods (Campillo & Paul, 2003; Duroux *et al.*, 2010; Roux *et al.*, 2004; Sabra *et al.*, 2005; Schuster, 2009; Snieder & Larose, 2013; Wapenaar *et al.*, 2005; Weaver & Lobkis, 2001), no physical receiver is required at the position of the virtual source; and unlike other Marchenko methods (Wapenaar *et al.*, 2013, 2014), we do not rely on an up/down decomposition of the wavefield. When one applies the Marchenko algorithm to two points in the interior, one obtains the Green's function for these two points recorded on the boundary. Using interferometric techniques, these Green's functions can be used to reconstruct the Green's function for waves propagating between two points in the interior (Brackenhoff *et al.*, 2019; Singh & Snieder, 2017).

Figure 2.3(c) shows the Green's function obtained from equation (2.7) with $\mathbf{x}$ taken at the boundary (blue lines), superimposed on the directly-modeled Green's function (red lines). For clarity, the traces have been multiplied by $\exp(2t)$ to emphasize the scattered waves. The latest arrival time for the single-scattered waves for our geometry is about 18 μs. All waves arriving after 18 μs therefore are multiply-scattered waves. For earlier times, the Green's function consists of a combination of single-scattered waves and multiply-scattered waves. As a result of our iterative solution, we retrieve the direct-wave and the scattered waves.

Figure 2.4 shows normalized vertical cross-sections of the wavefield at $t = 0$ taken from Figure 2.1(a) and Figure 2.1(b) for $x = 4$ cm. The red trace denotes the cross-section of Figure 2.1(a) and the blue trace denotes the cross-section of Figure 2.1(b). The snapshots (see Figure 2.1) and the cross-sections (see Figure 2.4) show that the reconstructed Green's function creates a focus only around the focusing point and cancels other arrivals around the focusing point to a large extent, whereas the results one can achieve with using only direct waves contain other arrivals that distort the focusing.



(a)

(b)

(c)

Figure 2.3 (a) $U_{total}(\hat{\mathbf{n}}', t)$ for the 7th iteration. (b) $U_{total}(\hat{\mathbf{n}}', t) - U_{total}(\hat{\mathbf{n}}', -t)$. (c) Comparison of the calculated (red line) and the retrieved (blue line) Green's functions. The traces have been multiplied by $\exp(2t)$ to emphasize the scattered waves.

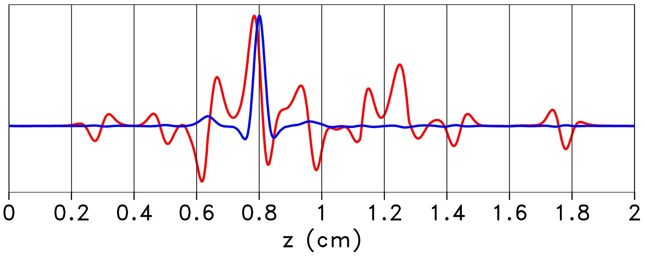


Figure 2.4 A normalized vertical cross-section of the snapshot at $x = 4$ cm in Figure 2.1(a) (red trace), and a normalized vertical cross-section of the snapshot at $x = 4$ cm in Figure 2.1(b) (blue trace). The blue trace is more concentrated at the focusing point at $z = 0.8$ cm than the red trace.

2.4 Conclusions

We derive the 2-D Marchenko equation for wavefield focusing and Green's function retrieval for an arbitrary point in an unknown highly scattering inhomogeneous medium with a closed receiver array. We successfully retrieve the Green's function for a pre-defined location and the comparison to the directly-modeled Green's function is found to be excellent (see Figure 2.3(c)). The cross-sections in Figure 2.4 show that we can create better focusing in the medium than one can achieve with the direct waves only. Our retrieved Green's function contains both the single- and multiply-scattered waves of the heterogeneous medium model. Because we use a constant background velocity model, our method requires the direct-wave information modeled only in the homogeneous medium (when ρ is constant), and the recorded scattering response $A(\hat{\mathbf{n}}, \hat{\mathbf{n}}', t)$ to solve the Marchenko equation iteratively like other multi-dimensional Marchenko methods proposed earlier (Wapenaar *et al.*, 2013, 2014); however, it does not require wavefield decomposition. We show that after the convergence, we retrieve the Green's function for any desired location in the medium without relying on prior information about the scatterers in the medium and wavefield decomposition to solve the Marchenko equation. The Marchenko equation we propose forms the basis for imaging the interior of a medium inside a closed array without up/down decomposition and makes the Marchenko methods more appropriate for imaging steeply dipping structures.

2.5 Acknowledgments

We thank the associate editor Greg McDaniel and two reviewers Matteo Ravasi and Leon Diekmann for their constructive reviews. This work is supported by the Consortium Project on Seismic Inverse Methods for Complex Structures at the Colorado School of Mines. The numerical examples in this paper were generated using the Madagascar software package (http://www.ahay.org). The research of K. Wapenaar has received funding from the European Research Council (grant no. 742703).

2.6 Appendix A: Source Selection In The Numerical Simulations

The main text of this paper shows that we use the ingoing wave at the boundary as sources. However, to incorporate the ingoing wave in the finite difference simulation, we use equivalent sources f for the source term in the acoustic wave equation (equation (2.1)) that produces the desired ingoing wave. In the numerical simulations, we radiate waves from the boundary into the medium from sources $f(\mathbf{x}, \omega)$, and the wavefield in the interior in the frequency domain is given by

$$p(\mathbf{x}, \omega) = \oint_{\mathbb{S}} G(R\hat{\mathbf{n}}, \mathbf{x}, \omega) f(\mathbf{x}, \omega) d\mathbb{S} \ , \tag{2.8}$$

where $p(\mathbf{x}, \omega)$ denotes the wavefield in the interior, G denotes the Green's function, R denotes the radius of the boundary, $\hat{\mathbf{n}}$ denotes the location on the circle $\mathbb{S}$, $R\hat{\mathbf{n}}$ denotes the location on the circle (see the main text), and ω denotes the angular frequency. To determine the source f that corresponds to an ingoing wave p^{in}, we use the representation theorem (Snieder & van Wijk, 2015)

$$p(\mathbf{x}, \omega) = \oint_{\mathbb{S}} \frac{1}{\rho} \Big\{ p(R\hat{\mathbf{n}}, \omega) \frac{\partial G(R\hat{\mathbf{n}}, \mathbf{x}, \omega)}{\partial n} - G(R\hat{\mathbf{n}}, \mathbf{x}, \omega) \frac{\partial p(R\hat{\mathbf{n}}, \omega)}{\partial n} \Big\} d\mathbb{S} \ , \tag{2.9}$$

where $p(R\hat{\mathbf{n}}, \omega)$ denotes the wavefield at the boundary. We decompose the wavefield at the boundary into incoming and outgoing waves as $p = p^{in} + p^{out}$, and in the far field, the

Green's function G is outgoing, hence

$$\frac{\partial G(R\hat{\mathbf{n}},\mathbf{x},\omega)}{\partial n} = ikG(R\hat{\mathbf{n}},\mathbf{x},\omega), \quad \text{and} \quad \frac{\partial p(R\hat{\mathbf{n}},\omega)}{\partial n} = -ikp^{in}(\hat{\mathbf{n}},\omega) + ikp^{out}(\hat{\mathbf{n}},\omega)\ . \tag{2.10}$$

By using the expressions in (2.10), we represent equation (2.9) as

$$p(\mathbf{x},\omega) = \frac{2}{c} i\omega \oint_{\mathbb{S}} \frac{1}{\rho} \Big\{ G(R\hat{\mathbf{n}},\mathbf{x},\omega) p^{in}(\hat{\mathbf{n}},\omega) \Big\} d\mathbb{S}\ , \tag{2.11}$$

where we used $k = \omega/c$. If we substitute equation (2.11) into equation (2.8), we obtain in the time domain

$$f(\mathbf{x},t) = -\frac{2}{\rho c} \frac{\partial p^{in}(\hat{\mathbf{n}},t)}{\partial t}\ , \tag{2.12}$$

where, with the used Fourier convention, $-i\omega$ corresponds in the time domain to differentiation with respect to time.

Figure 2.5(a) shows the wavefield recorded at the boundary of the iteratively solved Marchenko equation for the 7th iteration using the time-derivative in the source function. This figure shows that the time-derivative in the source selection creates artifacts which mainly reside in the time window $-t_d(\hat{\mathbf{n}}) < t < t_d(\hat{\mathbf{n}})$ (approximately between -5 μs and 5 μs). The artifacts are caused by waves that, in the iterative process, propagate along the receiver array. Figure 2.5(c) - Figure 2.5(f) show the wavefield in the interior of the iteratively solved Marchenko equation for the 7th iteration using the time-derivative in the source selection. It is also shown in Figure 2.5(c) - Figure 2.5(f) that this choice for f leads to numerical instabilities along the receiver array (the blue line in Figure 2.5).

To avoid this instability, we relate f to the normal derivative of the wavefield instead of the time-derivative. With the far-field approximation used in the expressions in (2.10), Green's theorem in equation (2.9) can be written as

$$p(\mathbf{x},\omega) = -\frac{2}{\rho} \oint_{\mathbb{S}} \Big\{ G(R\hat{\mathbf{n}},\mathbf{x},\omega) \frac{\partial p^{in}(\hat{\mathbf{n}},\omega)}{\partial n} \Big\} d\mathbb{S}\ , \tag{2.13}$$

which leads to the source function in the time domain

$$f(\mathbf{x},t) = -\frac{2}{\rho}\frac{\partial p^{in}(\hat{\mathbf{n}},t)}{\partial n} \ . \tag{2.14}$$

Figure 2.5(b) shows the wavefield recorded at the boundary of the iteratively solved Marchenko equation for the 7th iteration using the normal derivative in the source function. This figure shows that the normal derivative removes the artifacts which are mainly present in the time window $-t_d(\hat{\mathbf{n}}) < t < t_d(\hat{\mathbf{n}})$ (approximately between -5 μs and 5 μs) in Figure 2.5(a). Figure 2.5(g) - Figure 2.5(j) show the wavefield in the interior of the iteratively solved Marchenko equation for the 7th iteration using the normal derivative in the source selection. The artifacts which propagate along the receiver array in Figure 2.5(c) - Figure 2.5(f) are eliminated by using the normal derivative. Because the normal derivative removes the waves which propagate along the receiver array, Figure 2.5(g) - Figure 2.5(j) show artifact-free snapshots along the receiver array (the blue line in Figure 2.5).

Note that the normal derivative of the pressure field is equal to $i\omega\rho v_n$, with v_n the component of the particle velocity normal to the surface, hence the source function is proportional to the normal component of the acceleration.

2.7 Appendix B: The Homogeneous Green's Function Reconstruction

We provide a reasoning for equation (2.7) in the main text that is based on kinematic arguments, symmetry, and causality. In the following we assume that:

1. The wavefield satisfies the wave equation for the real system.

2. The reconstructed wavefield is source free.

3. The system is lossless and the wavefield is invariant for time-reversal.

4. The wavefield is anti-symmetric in time.

5. For every point $R\hat{\mathbf{n}}$ on the circle the wavefield vanishes for $-|R\hat{\mathbf{n}} - \mathbf{x}_f|/c < t < |R\hat{\mathbf{n}} - \mathbf{x}_f|/c$.

In general, the wavefield from sources $f(\mathbf{x}_s, \tau)$ can be written as

$$\tilde{u}(\mathbf{x}, t) = \iint G(\mathbf{x}, \mathbf{x}_s, t - \tau) f(\mathbf{x}_s, \tau) d\tau dV_s \ , \tag{2.15}$$

where $\tilde{u}(\mathbf{x}, t)$ denotes the wavefield in the interior, and $G(\mathbf{x}, \mathbf{x}_s, t)$ denotes the Green's function. Because the Green's function satisfies the wave equation for the real system, property 1 is satisfied for the wavefield in expression (2.15). This expression specifies the wavefield in terms of the sources $f(\mathbf{x}_s, \tau)$. It is, however, known that the inverse source problem is ill-posed (Bleistein & Cohen, 1977; Devaney & Sherman, 1982), hence for a given value of the wavefield at the circle $\mathbf{x} = R\hat{\mathbf{n}}$, the source function $f(\mathbf{x}_s, \tau)$ is not unique. We show, however, that properties 2-5 constrain this function up to a multiplicative constant.

Because of the property 4, the wavefield is anti-symmetric in time and using the property 3, solutions are invariant for time-reversal, we can create an anti-symmetric wavefield by taking the difference of $\tilde{u}(\mathbf{x}, t)$ and its time-reversed version $\tilde{u}(\mathbf{x}, -t)$ as

$$u(\mathbf{x}, t) = \tilde{u}(\mathbf{x}, t) - \tilde{u}(\mathbf{x}, -t) = \iint \left(G(\mathbf{x}, \mathbf{x}_s, t - \tau) f(\mathbf{x}_s, \tau) - G(\mathbf{x}, \mathbf{x}_s, -t - \tau) f(\mathbf{x}_s, \tau) \right) d\tau dV_s \ . \tag{2.16}$$

By construction, this solution satisfies property 4.

The Green's function satisfies

$$LG(\mathbf{x}, \mathbf{x}_s, t) = \delta(\mathbf{x} - \mathbf{x}_s)\delta(t) \ , \tag{2.17}$$

where L denotes the differential operator for the acoustic wave equation. Applying the operator L to the property (2.16) and carrying out the integrations over τ and $\mathbf{x}_s$ gives

$$Lu(\mathbf{x}, t) = f(\mathbf{x}, t) - f(\mathbf{x}, -t) \ . \tag{2.18}$$

The right hand side of this expression gives the sources of the wavefield $u(\mathbf{x}, t)$. According to property 2, the wavefield is source-free, therefore, the right hand side of expression

(2.18) vanishes, so that

$$f(\mathbf{x},t) = f(\mathbf{x},-t) \ , \tag{2.19}$$

showing that $f(\mathbf{x},t)$ is symmetric in time.

By using expression (2.19), we can replace the last term $f(\mathbf{x}_s,\tau)$ in expression (2.16) by $f(\mathbf{x}_s,-\tau)$. Replacing next the integration variable τ by $-\tau$ in the second term reduces expression (2.16) to

$$u(\mathbf{x},t) = \iint \left(G(\mathbf{x},\mathbf{x}_s,t-\tau) - G(\mathbf{x},\mathbf{x}_s,\tau-t)\right) f(\mathbf{x}_s,\tau) d\tau dV_s \ . \tag{2.20}$$

This expression can be written as

$$u(\mathbf{x},t) = \iint G_h(\mathbf{x},\mathbf{x}_s,t-\tau) f(\mathbf{x}_s,\tau) d\tau dV_s \ , \tag{2.21}$$

where $G_h(\mathbf{x},\mathbf{x}_s,t) = G(\mathbf{x},\mathbf{x}_s,t) - G(\mathbf{x},\mathbf{x}_s,-t)$ is the homogeneous Green's function defined in equation (2.7). The homogeneous Green's function is source-free (Oristaglio, 1989), so it is natural that the homogeneous Green's function arises from the requirement that the wavefield is source-free.

We next apply the property 5 to further constrain $f(\mathbf{x}_s,\tau)$. For positive times, the homogeneous Green's function $G_h(\mathbf{x},\mathbf{x}_s,t-\tau)$ has a first arriving wave at location $\mathbf{x}$ that is excited at $\mathbf{x}_s$ and time τ at time

$$t = \tau + |\mathbf{x} - \mathbf{x}_s|/c \ . \tag{2.22}$$

According to expression (2.19), the function $f(\mathbf{x}_s,\tau)$ is symmetric in time. This means that for every time source at $\mathbf{x}_s$ at time τ, there is an equal contribution from a source at $\mathbf{x}_s$ at time $-\tau$, which generates a first arriving wave at

$$t = -\tau + |\mathbf{x} - \mathbf{x}_s|/c \ . \tag{2.23}$$

The time τ can be either positive or negative. Hence the first arriving wave arrives at $t = min(\tau,-\tau) + |\mathbf{x} - \mathbf{x}_s|/c$. Since $min(\tau,-\tau) = -|\tau|$, the first arrival excited by a source at $\mathbf{x}_s$ at time $\pm\tau$ arrives at the point $\mathbf{x}$ at time

$$t = -|\tau| + |\mathbf{x} - \mathbf{x}_s|/c \; . \tag{2.24}$$

For a point $R\hat{\mathbf{n}}$ on the circle, the first arriving wave excited at a point $\mathbf{x}_s$ at time $\pm\tau$ arrives at

$$t = -|\tau| + |R\hat{\mathbf{n}} - \mathbf{x}_s|/c \; . \tag{2.25}$$

Property 5 states that for positive time t and a point $R\hat{\mathbf{n}}$ on the circle, the wavefield vanishes when

$$t < |R\hat{\mathbf{n}} - \mathbf{x}_f|/c \; , \tag{2.26}$$

where $\mathbf{x}_f$ is the focusing point. Since the wavefield vanishes for these times, the first arriving waves that are excited at $\mathbf{x}_s$ must have arrival times greater or equal to $|R\hat{\mathbf{n}} - \mathbf{x}_f|/c$. Using expression (2.25), this implies that

$$-|\tau| + |R\hat{\mathbf{n}} - \mathbf{x}_s|/c \geq |R\hat{\mathbf{n}} - \mathbf{x}_f|/c \; , \tag{2.27}$$

We can also write this inequality as

$$c|\tau| \leq |R\hat{\mathbf{n}} - \mathbf{x}_s| - |R\hat{\mathbf{n}} - \mathbf{x}_f| \; , \tag{2.28}$$

Note that the focusing point $\mathbf{x}_f$ is specified, while the point $\mathbf{x}_s$ can be anywhere within the circle of radius R.

Let us consider a point $\mathbf{x}_s \neq \mathbf{x}_f$, as shown in Figure 2.6. For the point P in that figure $|R\hat{\mathbf{n}} - \mathbf{x}_s| < |R\hat{\mathbf{n}} - \mathbf{x}_f|$, the inequality (2.28) reduces to $c|\tau| \leq |R\hat{\mathbf{n}} - \mathbf{x}_s| - |R\hat{\mathbf{n}} - \mathbf{x}_f| < 0$, or $c|\tau| < 0$. This inequality cannot be satisfied for any value of τ, which means that a point $\mathbf{x}_s \neq \mathbf{x}_f$ cannot be a source of the wavefield.

Consider next the case $\mathbf{x}_s = \mathbf{x}_f$. In that case the inequality (2.28) reduces to $c|\tau| \leq 0$. This inequality can only be satisfied for $\tau = 0$. Together with the fact that $f(\mathbf{x}_s, \tau)$ is only nonzero for $\mathbf{x}_s = \mathbf{x}_f$, this implies that the source-time function is local in space and time

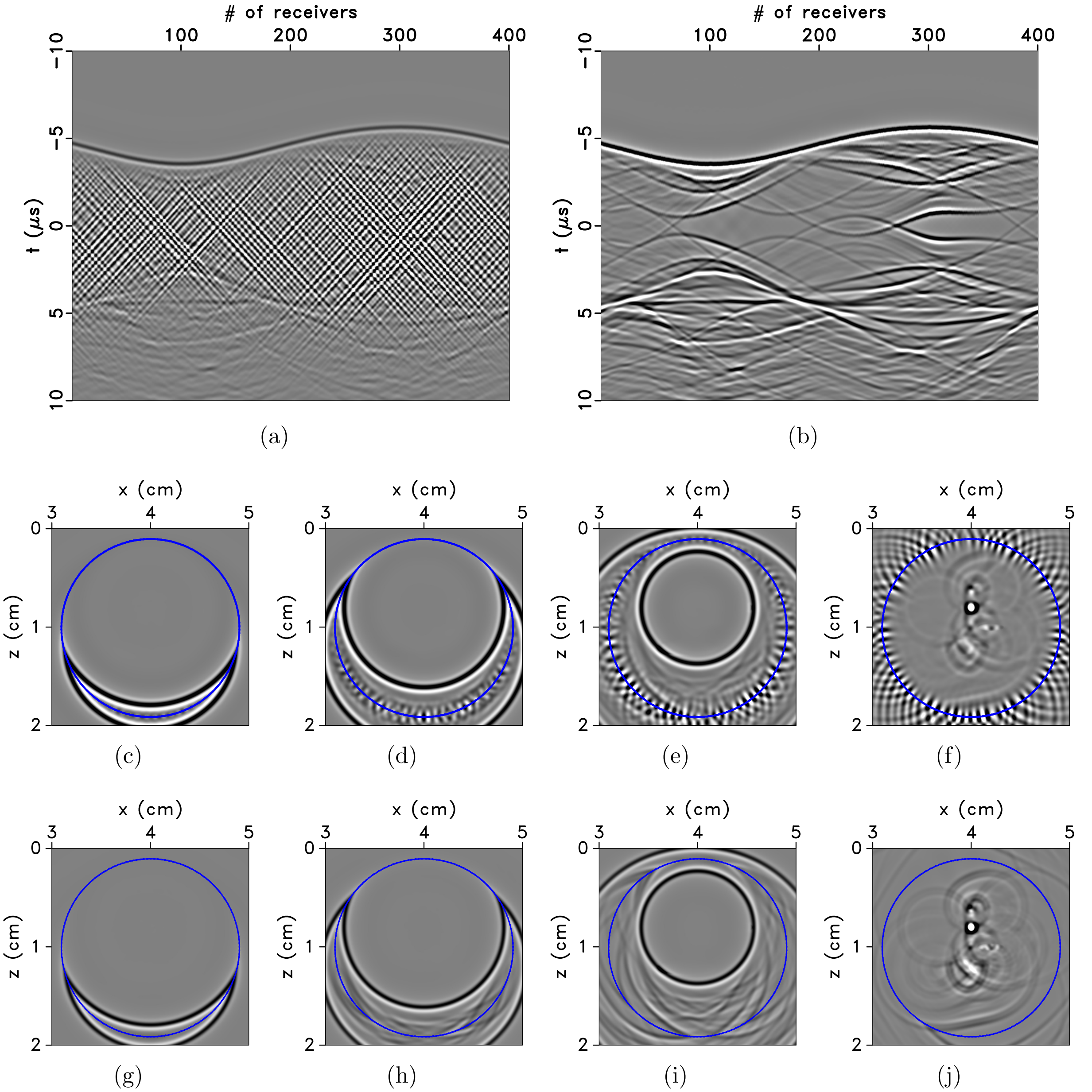
$$f(\mathbf{x}_s, \tau) = S\delta(\mathbf{x}_s - \mathbf{x}_f)\delta(\tau) \; , \tag{2.29}$$

where S is a multiplicative constant. Note that any even time derivative of $\delta(t)$ would also give the required localization in time. Since in any physical experiment and computer simulation the wavefield is convolved with a wavelet, we ignore this subtlety.

Inserting this source function into expression (2.21), and carrying out the integration over $\mathbf{x}_s$ and τ gives

$$u(\mathbf{x}, t) = S G_h(\mathbf{x}, \mathbf{x}_f, t) \; . \tag{2.30}$$

This implies that, up to a multiplicative constant S, the wavefield is given by the homogeneous Green's function with a source at the focusing point.



(a) (b)

(c) (d) (e) (f)

(g) (h) (i) (j)

Figure 2.5 Boundary wavefield for the 7th iteration using (a) time-derivative in the source selection and (b) normal derivative in the source selection. Note the cleaned artifacts for the time interval $-t_d(\hat{\mathbf{n}}) < t < t_d(\hat{\mathbf{n}})$ (approximately between -5 μs and 5 μs) in (b). Also, snapshots of the wavefield in the interior obtained by the iterative Marchenko algorithm for the 7th iteration with the time-derivative in the source selection at (c) t = -5.1 μs, (d) t = -4.2 μs, (e) t = -3 μs, and (f) t = 0 μs; with the normal derivative in the source selection at (g) t = -5.1 μs, (h) t = -4.2 μs, (i) t = -3 μs, and (j) t = 0 μs. The blue line shows the receiver locations. Note that the snapshots have different scaling than those shown in the main text.

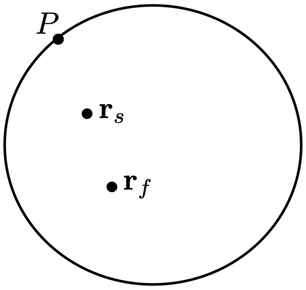


Figure 2.6 Definition of the point P given locations $\mathbf{x}_s$ and $\mathbf{x}_f$.

CHAPTER 3
THE ROLE OF THE BACKGROUND VELOCITY MODEL FOR THE MARCHENKO FOCUSING OF REFLECTED AND REFRACTED WAVES

A paper submitted to *Geophysical Journal International*

Mert Sinan Recep Kiraz [1,2,3], Roel Snieder[3] & Kees Wapenaar[4]

Marchenko algorithms retrieve the wavefields excited by virtual sources in the subsurface, these are the Green's functions consisting of the primary and multiple reflected waves. The requirements for these algorithms are the same as for conventional imaging algorithms; they need an estimate of the velocity model and the recorded reflected waves. We investigate the dependence of the retrieved Green's functions using the Marchenko equation on the background velocity model and address the question: "How well do we need to know the velocity model for accurate Marchenko focusing?". We present three different background velocity models and compare the Green's functions retrieved using these models. We show that these retrieved Green's functions using the Marchenko equation give correlation coefficients with the exact Green's function larger than 90% on average except near the edges of the receiver aperture. We also examine the presence of refracted waves in the retrieved Green's function. We show with a numerical example that the Marchenko focusing algorithm produces refracted waves only if the initial velocity model used for the iterative scheme is sufficiently detailed to model the refracted waves.

3.1 Introduction

The inverse scattering community utilized the Marchenko equation to relate scattered data to the scattering potential to determine the medium properties (Agranovich &

[1] Primary researcher and author.

[2] Author for correspondence. Direct correspondence to mertkiraz@gmail.com.

[3] Center for Wave Phenomena, Colorado School of Mines, 1500 Illinois St., Golden, CO 80401, USA.

[4] Department of Geoscience and Engineering, Delft University of Technology, P.O. Box 5048, Delft, GA 2600, The Netherlands.

Marchenko, 1963; Burridge, 1980; Chadan & Sabatier, 1989; Colton & Kress, 1998; Gel'fand & Levitan, 1955; Gladwell, 1993; Marchenko, 1955; Newton, 1980a). The connection between wavefield focusing and the Marchenko equation was made by Rose (2001, 2002b) so that the wavefield focusing at a location in an unknown medium can be achieved once the Marchenko equation is solved. Broggini & Snieder (2012) connect the Marchenko equation and seismic interferometry (Curtis *et al.*, 2006; Derode *et al.*, 2003; Snieder & Larose, 2013; Wapenaar *et al.*, 2005; Weaver & Lobkis, 2001) and show that one can retrieve the Green's function between any point in the subsurface and points on the acquisition surface without a physical receiver at the virtual source location and with one-sided illumination. Wapenaar *et al.* (2013) discuss the three-dimensional Green's function retrieval, and present an example of the two-dimensional Green's function retrieval.

A thorough description of the Marchenko redatuming and imaging method and its numerical implementation is given by Wapenaar *et al.* (2014), van der Neut *et al.* (2015), Thorbecke *et al.* (2017), and Lomas & Curtis (2019). Marchenko methods have been widely used for various applications such as internal multiple elimination (Meles *et al.*, 2015, 2016; Thorbecke *et al.*, 2021), elastic wave applications (da Costa Filho *et al.*, 2014, 2015; Wapenaar, 2014), subsurface imaging and for comparisons with the conventional imaging results (e.g., reverse time migration) (Behura *et al.*, 2014; Jia *et al.*, 2018; Ravasi *et al.*, 2016; Wapenaar *et al.*, 2014). Additionally, various field data set applications of the Marchenko method have been performed such as imaging of a North Sea field data set (Ravasi *et al.*, 2016), target-oriented subsalt imaging of the Gulf of Mexico data set (Jia *et al.*, 2018), time-lapse monitoring of the Frio carbon sequestration data set (Kiraz & Nowack, 2018), multiple suppression on an Arabian Gulf field data set (Staring *et al.*, 2021), and an offshore Brazil data set for imaging a reservoir under basalt (Jia *et al.*, 2021). With growing interest in machine learning applications in seismology, a convolutional neural network-based one-dimensional wavefield focusing is also proposed by

Kiraz & Snieder (2022).

Recent studies have aimed to address the limitations of the up/down separation of the Marchenko equation. Using the data collected on a closed received array, Kiraz *et al.* (2020, 2021) retrieve the full-field (e.g., without component separation) Green's function, and show that the full-field Marchenko focusing provides better focusing results in the subsurface than achievable using only the direct waves. Diekmann & Vasconcelos (2021) and Wapenaar *et al.* (2021) show alternative methods where one can retrieve the Green's function using single-sided acquisition data without up/down decomposition.

In this paper, we use the one-sided Marchenko focusing to retrieve the Green's function at an arbitrary depth location using different subsurface models with variable velocity and variable density profiles. In Section 3.2, we describe the Marchenko focusing algorithm and show that it requires only two inputs; surface-recorded data and the initial estimate of the velocity model, which are the same inputs as for conventional imaging algorithms. In Section 3.3, we provide a visual tour to describe the iterative Marchenko focusing algorithm. In Section 3.4, we investigate the background velocity model dependence of the Marchenko method and show the accuracy of the retrieved Green's function by presenting Correlation Coefficients (CCs) between the retrieved and numerically modeled Green's functions. Lastly, in Section 3.4, we use the Marmousi model to investigate the presence of the refracted waves in the Marchenko focusing, and show that the presence of the refracted waves depends only on the initial estimate of the velocity model.

3.2 Methodology

We use the Marchenko algorithm proposed by Wapenaar *et al.* (2013) which builds on the earlier work of Rose (2001), Rose (2002b), and Broggini & Snieder (2012). We denote spatial coordinates as $\mathbf{x} = (x, z)$, and the receiver coordinates as $\mathbf{x}_R = (x_R, z_R)$. The receivers are located at the transparent acquisition surface at $z = 0$, and the multiples caused by the free surface (e.g., air-water interface) are excluded.

We relate the ingoing wave, U_k^{in}, to the outgoing wave, U_k^{out}, at iteration k as

$$U_k^{out}(\mathbf{x}_R, t) = \int R(\mathbf{x}_R, \mathbf{x}, t) * U_k^{in}(\mathbf{x}, t) dx \ , \tag{3.1}$$

where R corresponds to the reflection response of the medium and the asterisk denotes temporal convolution. The iterative scheme starts with modeling the direct wave, and we denote the arrival time of the direct wave from the virtual source location, $\mathbf{x}_s$, for which we aim to retrieve the Green's function, to the receivers at the surface as t_d. Following Broggini & Snieder (2012), we design an iterative scheme so that at $t = 0$, the wavefield becomes a delta function at the pre-defined focal (or virtual source) location. We start the iterative algorithm by defining the ingoing wavefield at $z = 0$ for iteration k as

$$U_k^{in}(\mathbf{x}, t) = U_0^{in}(\mathbf{x}, t) - \Theta(\mathbf{x}, t) U_{k-1}^{out}(\mathbf{x}, -t) \ , \tag{3.2}$$

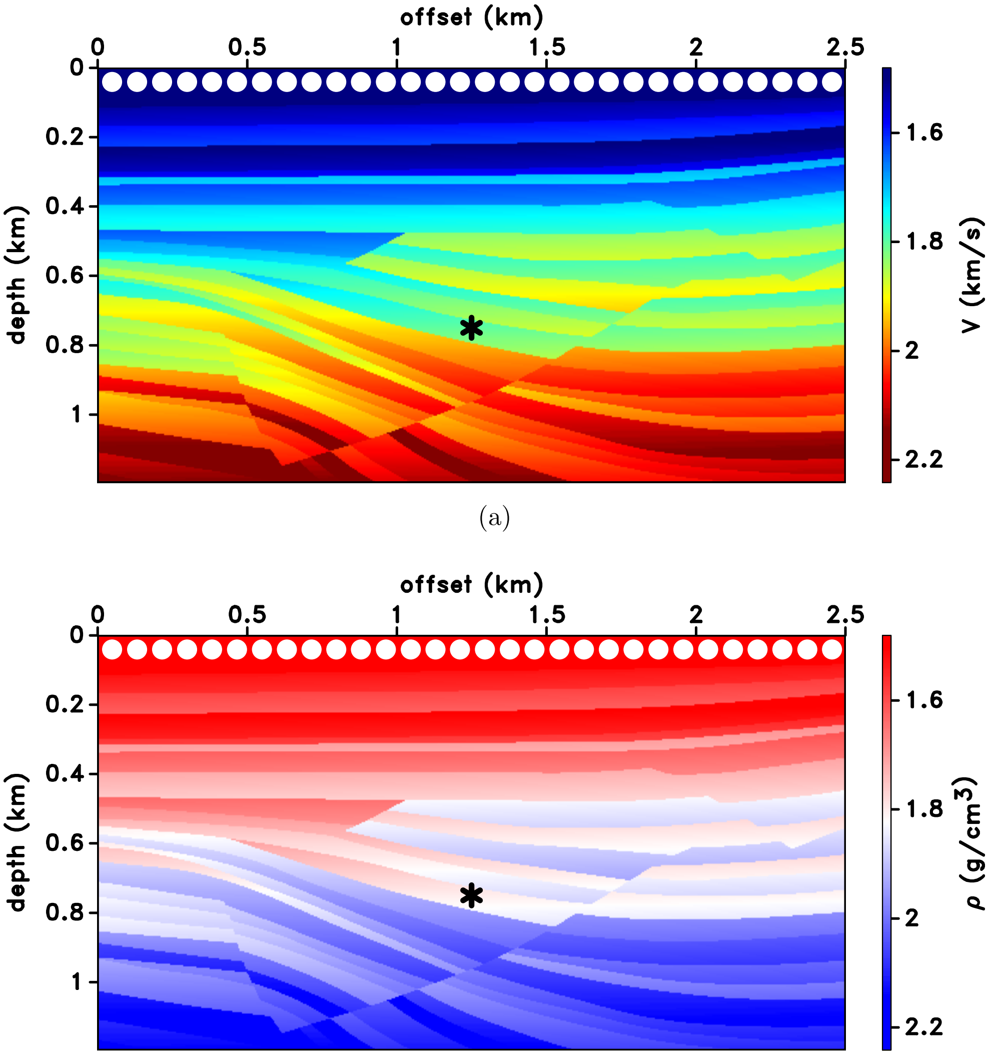
where $\Theta(\mathbf{x}, t)$ defines a window function where $\Theta(\mathbf{x}, t) = 1$ when $-t_d^{\epsilon}(\mathbf{x}) < t < t_d^{\epsilon}(\mathbf{x})$ and otherwise $\Theta(\mathbf{x}, t) = 0$ with $t_d^{\epsilon} = t_d - \epsilon$ where we introduce ϵ as a small positive constant to exclude the direct wave at t_d. After the convergence is achieved (i.e., $U_k^{out} = U_{k-1}^{out}$) we can drop the iteration number, and insert equation (3.2) into equation (3.1), and obtain

$$U^{out}(\mathbf{x}_R, t) = \int R(\mathbf{x}_R, \mathbf{x}, t) * U_0^{in}(\mathbf{x}, t) \ dx - \int R(\mathbf{x}_R, \mathbf{x}, t) * \Theta(\mathbf{x}, t) U^{out}(\mathbf{x}, -t) \ dx \ . \tag{3.3}$$

Once the convergence is achieved, we denote the recorded data at the receivers as $U_{total}(\mathbf{x}, t) = U^{in}(\mathbf{x}, t) + U^{out}(\mathbf{x}, t)$ which consists of the superposition of the ingoing and outgoing wavefield. We then define $p(\mathbf{x}, t)$ as the total wavefield that is associated with $U_{total}(\mathbf{x}, t)$. We obtain the homogeneous Green's function ($G_h(\mathbf{x}, \mathbf{x}_s, t) = G(\mathbf{x}, \mathbf{x}_s, t) - G_(\mathbf{x}, \mathbf{x}_s, -t)$) (Oristaglio, 1989), for the virtual source location $\mathbf{x}_s$ and the receiver location $\mathbf{x}$ as

$$G_h(\mathbf{x}, \mathbf{x}_s, t) = p(\mathbf{x}, t) - p(\mathbf{x}, -t) \ . \tag{3.4}$$

Equation (3.4) satisfies the homogeneous wave equation, and, hence, retrieves the Green's function for times $t > 0$ for the virtual source location $\mathbf{x}_s$ (Kiraz *et al.*, 2021; Oristaglio, 1989; Wapenaar *et al.*, 2013). The iterative scheme we use follows the algorithm presented by Wapenaar *et al.* (2013), and equations (3.1), (3.2), and (3.4) are given in Wapenaar *et al.* (2013) in equations (13), (12), and (14), respectively.



(a)

(b)

Figure 3.1 (a) Velocity model and (b) density model of the synthetic example which are extracted from the Sigsbee model. The black asterisk shows the virtual source location and the white dots at the top indicate every 30th source/receiver location.

3.3 Visualizing The Iterative Scheme

In this section, we present a numerical example to aid the visual understanding of the iterative scheme described in Section 3.2. Figure 3.1 shows the subsurface model and source and receiver geometry of our first numerical experiment. Figure 3.1(a) and Figure 3.1(b) show the variable velocity and density models used for the numerical example, respectively, which are extracted from the Sigsbee model (Stoughton *et al.*, 2001) with 2.5 km horizontal and 1.2 km vertical extent. The virtual source location is at $x_s = 1.25$ km and $z_s = 0.75$ km in depth which is shown with the black asterisk in Figure 3.1. The white dots located at the surface of the models in Figure 3.1 represent every 30th receiver location. We use point impulsive sources and record pressure at the receivers, and exclude the presence of the free surface. During the iterative Marchenko scheme, we use the normal derivative of the pressure field to send the wavefield into the medium from the receiver array (Kiraz *et al.*, 2021).

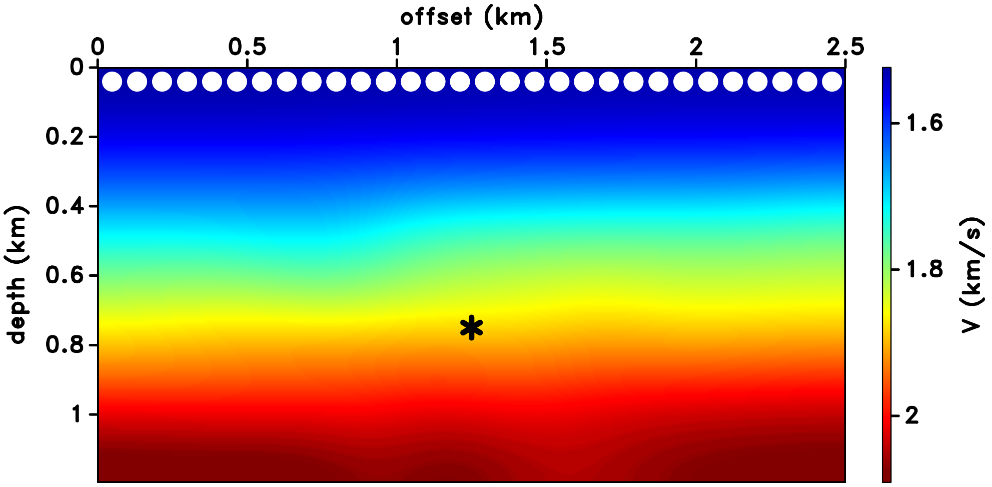
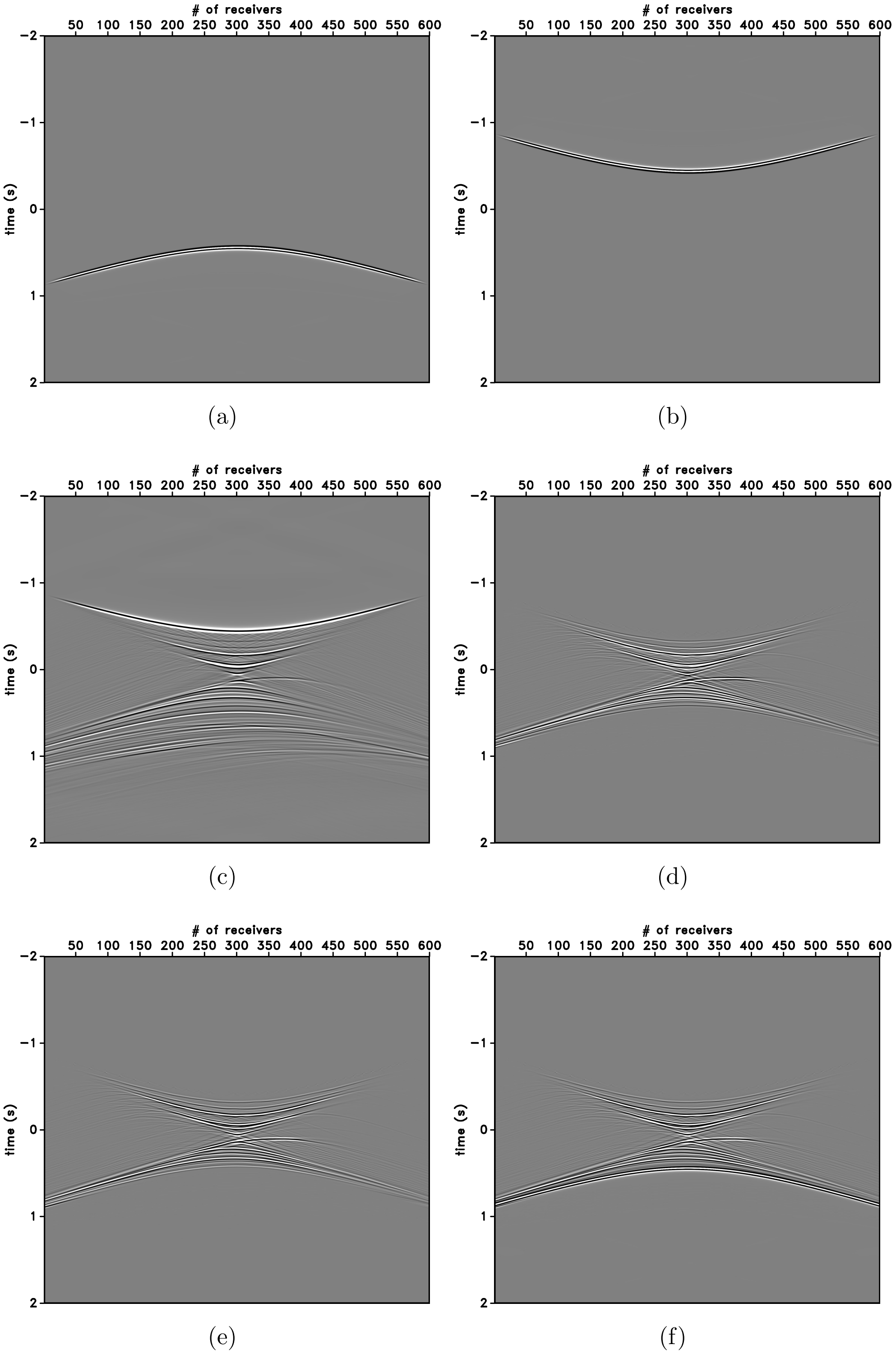


Figure 3.2 Smoothed version of the velocity model used for the iterative algorithm. The black asterisk shows the virtual source location and the white dots at the top indicate every 30th source/receiver location.



(a)

(b)

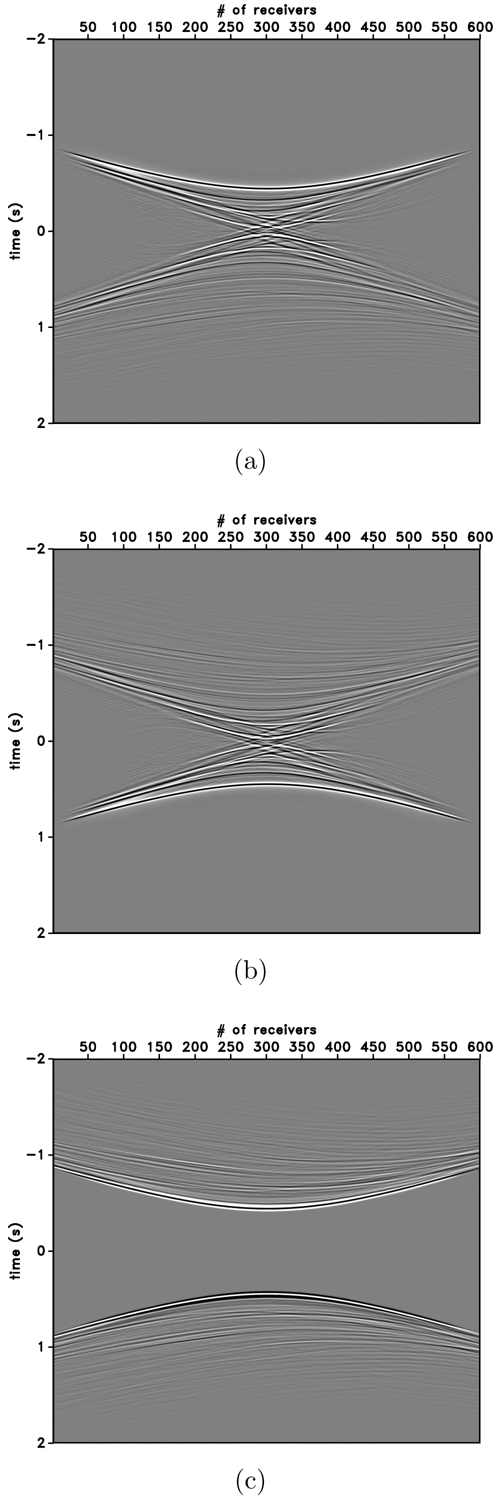
(c)

(d)

(e)

(f)

Figure 3.3 (a) Modeled direct wave. (b) Time-reversed direct wave. (c) Recorded wavefield, $U_{total}(\mathbf{x}, t)$, for the first iteration after sending in the time-reversed direct wave. (d) Wavefield after windowing applied. (e) Wavefield after multiplying with (-1). (f) Wavefield after adding the modeled direct wave. This is also the input for the second iteration.



(a)

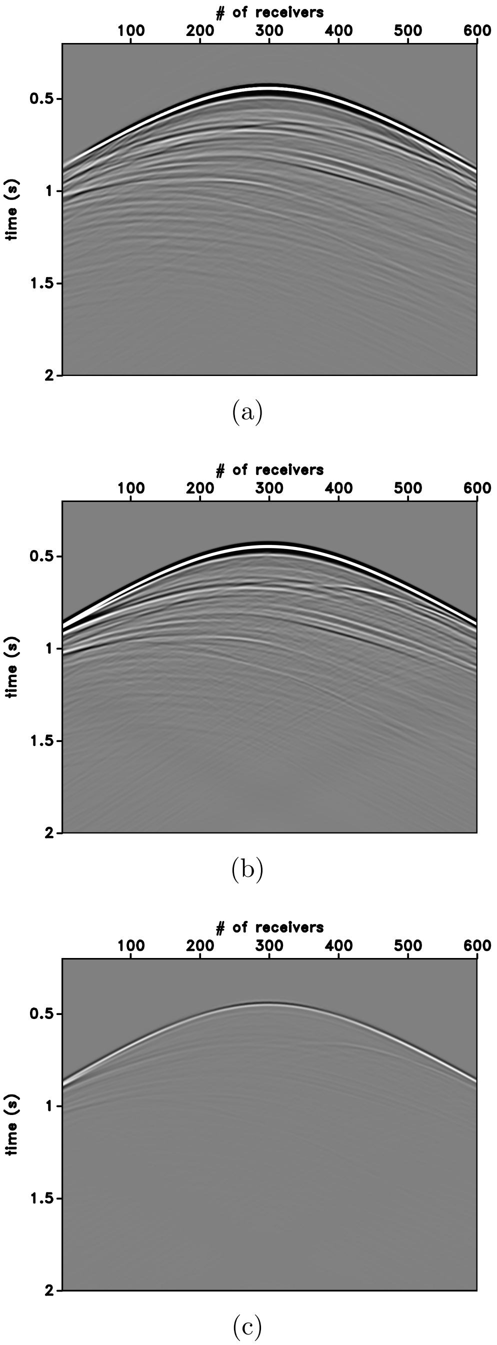
(b)

(c)

Figure 3.4 (a) $U_{total}(\mathbf{x}, t)$ for the fourth iteration. (b) $U_{total}(\mathbf{x}, -t)$ for the fourth iteration. (c) $G_h(\mathbf{x}, \mathbf{x}_s, t) = p(\mathbf{x}, t) - p(\mathbf{x}, -t)$ for the fourth iteration.

Figure 3.2 shows the smoothed version of the velocity model which is obtained by smoothing the slowness. This model is so strongly smoothed that it presents mostly one-dimensional information about the velocity. As we follow the iterative scheme described in Section 3.2, the first step is to model the direct wave using the smooth background velocity model (shown in Figure 3.2). Figure 3.3(a) shows the modeled direct wave. After modeling the direct wave, the next step is to time-reverse the direct wave which is shown in Figure 3.3(b). We then inject the time-reversed direct wave on the boundary (one can also convolve the time-reversed direct wave with the reflection response). Figure 3.3(c) shows the total wavefield, $U_{total}(\mathbf{x}, t)$. This is the recorded data at the receiver array after sending in the time-reversed direct wave (Figure 3.3(b)) from the receiver array located at the surface of the medium. We use Figure 3.3(a) and Figure 3.3(b) to define the window function (Θ in equation (3.2)). After using this window function, Figure 3.3(d) shows the muted data. Following the iterative algorithm, we next negate the muted data and the resulting wavefield is shown in Figure 3.3(e). As the last step, we add the direct wave (Figure 3.3(a)) to this wavefield (U_0^{in} in equation (3.2)) and Figure 3.3(f) shows the combined wavefield (U_k^{in} in equation (3.2)). The time-reverse version of the wavefield shown in Figure 3.3(f) is, therefore, the input for the second iteration and is ready to be sent back into the medium using the receiver array at the surface.

Figure 3.4(a) shows $U_{total}(\mathbf{x}, t)$ for the fourth iteration of the iterative scheme. Note that Figure 3.4(a) is nearly symmetric in time for the times $-t_d < t < t_d$, defined using the arrival time of the direct wave (approximately between -1s and 1s). Figure 3.4(b) shows $U_{total}(\mathbf{x}, -t)$ for the fourth iteration which is the time-reversed recorded data at the receivers after the fourth iteration. Figure 3.4(c) shows the homogeneous Green's function, $G_h(\mathbf{x}, \mathbf{x}_s, t)$ (see equation (3.4)), for the fourth iteration after muting the waves between the direct arrivals.



(a)

(b)

(c)

Figure 3.5 (a) Retrieved Green's function using the Marchenko focusing (times when $t > 0$ of G_h in Figure 3.4(c)). (b) Numerically modeled Green's function. (c) Difference between the numerically modeled Green's function in (a) and the retrieved Green's function in (b).

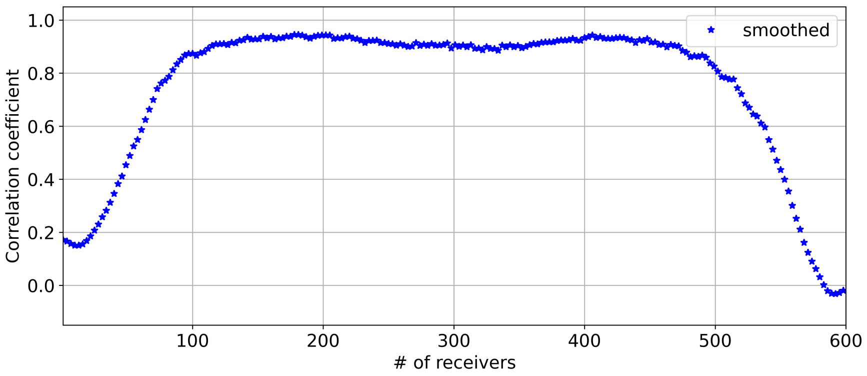


Figure 3.6 Trace-by-trace calculated correlation coefficient between the retrieved Green's function (Figure 3.5(a)) and the numerically modeled Green's function (Figure 3.5(b)).

Figure 3.5(a) shows $G_h(\mathbf{x}, \mathbf{x}_s, t)$ for the fourth iteration for positive times only which is the retrieved Green's function and Figure 3.5(b) shows the numerically modeled Green's function for the virtual source location. The difference between the numerically modeled and the retrieved Green's functions is shown in Figure 3.5(c). Figure 3.5 shows that the numerically modeled Green's function closely matches the retrieved Green's function for $t \geq t_d$. However, we also see in Figure 3.5(c) that there are overall mismatches in amplitudes and right and left edges of the wavefield that are due to the limited aperture used during the injection of the wavefield back into the medium from the receiver array.

We measure the accuracy of the Green's function retrieval using the Marchenko equation results by calculating trace-by-trace CCs between the retrieved Green's function and the numerically modeled Green's function. The CCs between the retrieved Green's function (Figure 3.5(a)) using the smoothed version of the velocity model and the numerically modeled Green's function (Figure 3.5(b)) are shown in Figure 3.6. The average CC in Figure 3.6 is 0.76. The low CCs around the right and left edges of the CC plot in Figure 3.6 are due to the limited aperture used during the injection of the wavefield. For the receivers where the limited aperture effects are not evident (receivers from 100 to 500), the average CC is 0.91. This shows a high accuracy Green's function retrieval by the Marchenko focusing algorithm using the smoothed version of the velocity model

(Figure 3.2).

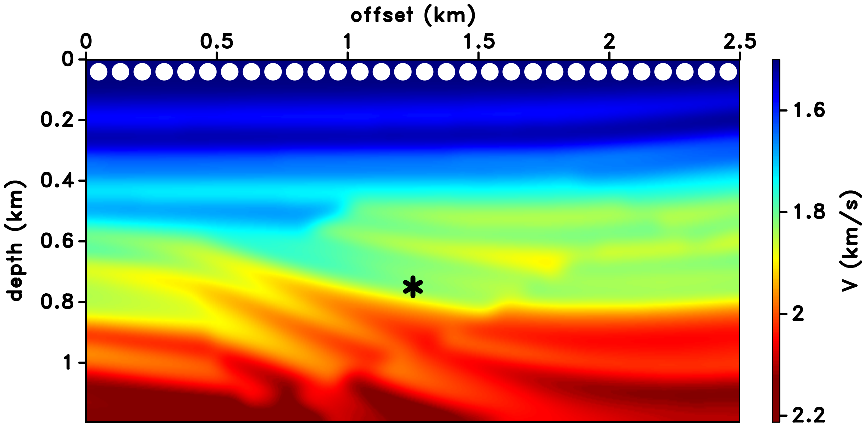
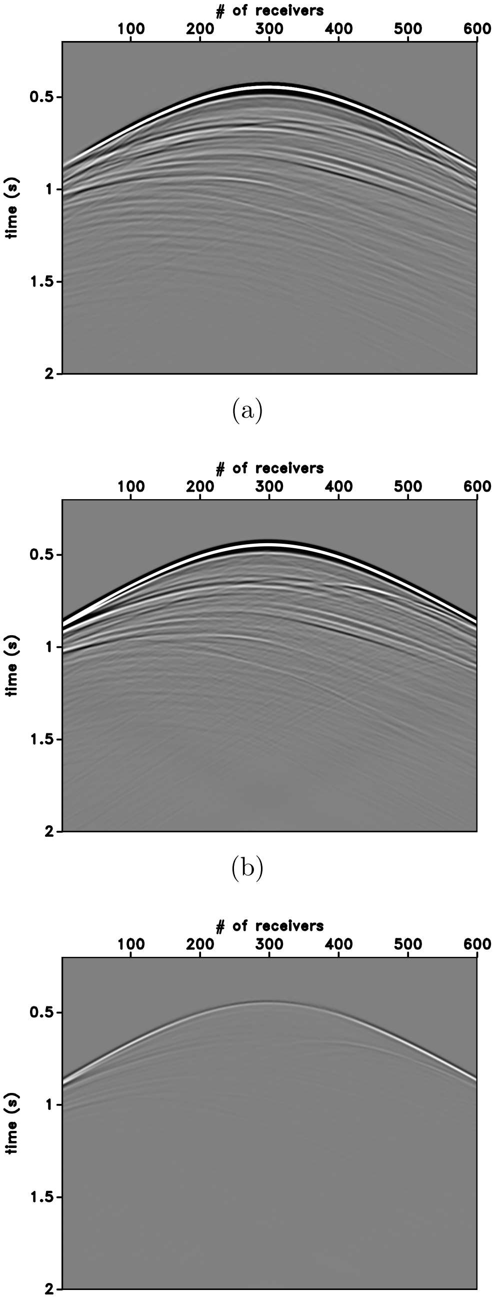


Figure 3.7 Less smoothed version of the velocity model used for the iterative algorithm. The black asterisk shows the virtual source location and the white dots at the top indicate every 30th source/receiver location.

3.4 Importance Of The Initial Background Velocity Model

In this section, we investigate the effect of the smoothness of the background velocity model on the retrieved Green's function. The second numerical example consists of the same velocity and density models as with the first example (see Figure 3.1(a) and Figure 3.1(b)); however, this time we use a less smoothed version of the background velocity model than the one presented in Figure 3.2 to retrieve the Green's function using the Marchenko focusing. Figure 3.7 shows a less smoothed velocity model which has more detailed information about the subsurface structures than the one shown in Figure 3.2. As with the iterative algorithm, we use the less smoothed velocity model to produce the direct wave and initiate the iterative scheme. By following the steps presented in Figure 3.3, we retrieve the Green's function. Figure 3.8(a) shows G_h for the fourth iteration for positive times only (being the retrieved Green's function) and Figure 3.8(b) shows the numerically modeled Green's function for the virtual source location. The difference between the numerically modeled Green's function (Figure 3.8(b)) and the retrieved Green's function Figure 3.8(a) is shown in Figure 3.8(c). Similar to Figure 3.5(c), Figure 3.8(c) shows that the retrieved and modeled Green's functions have mismatches in overall amplitude, and the right and left wavefield edges.



(a)

(b)

(c)

Figure 3.8 (a) Retrieved Green's function using the Marchenko focusing using the less smooth velocity model. (b) Numerically modeled Green's function (which is the same wavefield as Figure 3.5(b)). (c) Difference between the numerically modeled Green's function in (a) and the retrieved Green's function in (b).

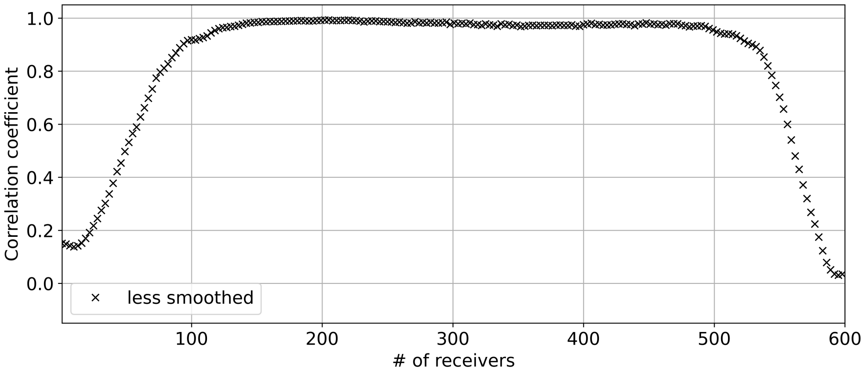


Figure 3.9 Trace-by-trace calculated correlation coefficient between the retrieved Green's function (Figure 3.8(a)) and the numerically modeled Green's function (Figure 3.5(b) and Figure 3.8(b)).

To quantify the quality of the retrieved Green's function using the less smoothed velocity model (Figure 3.7), we calculate the CCs between the retrieved Green's function (Figure 3.8(a)) and the numerically modeled Green's function (Figure 3.8(b)), which are shown in Figure 3.9. The average CC in Figure 3.9 is 0.83, and the average CC for receivers from 100 to 500 is 0.98. Therefore, by using a less smoothed version of the velocity model (Figure 3.7) for the iterative scheme, we retrieve a more accurate Green's function by the Marchenko focusing algorithm than the one presented in Figure 3.6.

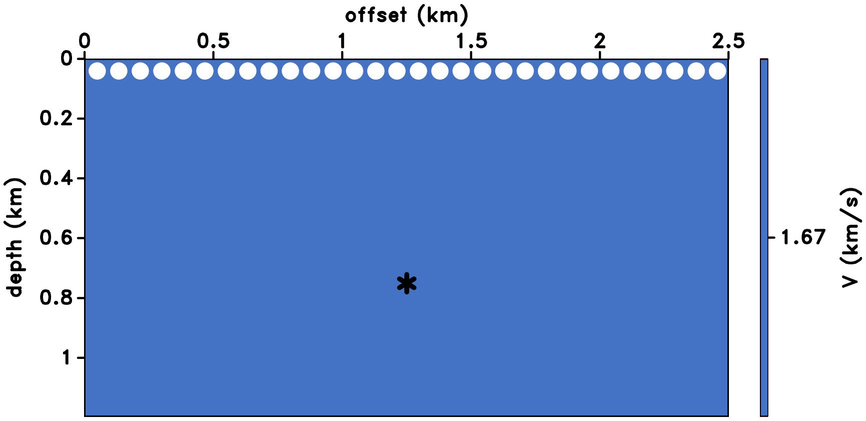
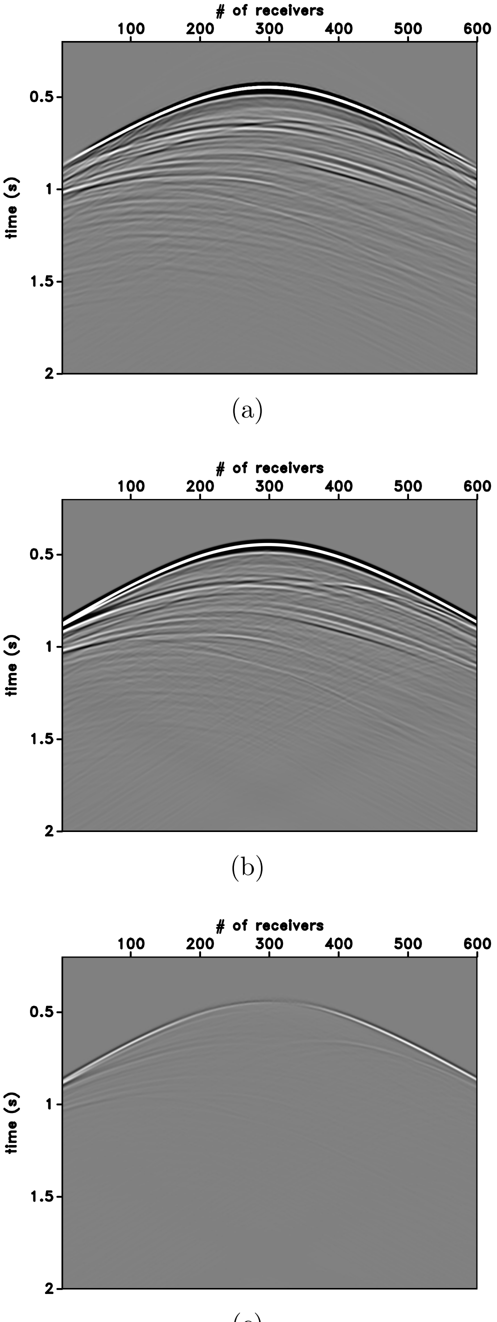


Figure 3.10 Constant velocity model used for the iterative algorithm. The black asterisk shows the virtual source location and the white dots at the top indicate every 30th source/receiver location.



(a)

(b)

(c)

Figure 3.11 (a) Retrieved Green's function using the Marchenko focusing using the constant velocity model. (b) Numerically modeled Green's function (which is the same wavefield as Figure 3.5(b) and Figure 3.8(b)). (c) Difference between the numerically modeled Green's function in (a) and the retrieved Green's function in (b).

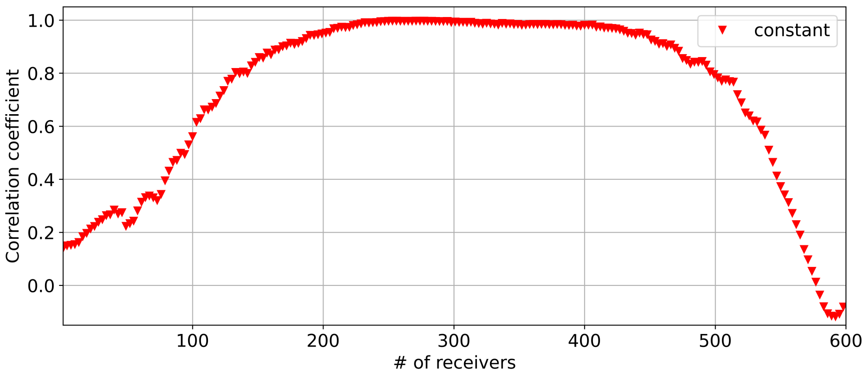


Figure 3.12 Trace-by-trace calculated correlation coefficient between the retrieved Green's function (Figure 3.11(a)) and the numerically modeled Green's function (Figure 3.5(b), Figure 3.8(b), and Figure 3.11(b)).

As the last step of the velocity model sensitivity analysis, we use a constant velocity model. As opposed to the first two velocity models used (see Figure 3.2 and Figure 3.7), the constant velocity model does not include any geological or geophysical information, including information about the possible dipping layers and the velocity variations. The constant value of the velocity is calculated using the average slowness between the surface and the depth of the focal point and is shown in Figure 3.10, which is used to model the direct wave for the iterative algorithm. After following the iterative scheme, Figure 3.11(a) shows G_h for the fourth iteration for positive times only (the retrieved Green's function), and Figure 3.11(b) shows the numerically modeled Green's function for the virtual source location. Figure 3.11(c) shows the difference between the numerically modeled (Figure 3.11(b)) and the retrieved (Figure 3.11(a)) Green's functions. The difference in Figure 3.11(c) using the constant velocity model (Figure 3.10) is similar to the ones presented in Figure 3.5(c) and Figure 3.8(c). There are also mismatches in overall amplitudes, and the right and left wavefield edges.

Similar to the previous examples, we show the accuracy of the retrieved Green's function using the constant velocity model (Figure 3.10) by calculating the CCs between the retrieved (Figure 3.11(a)) and the numerically modeled (Figure 3.11(b)) Green's functions, which are shown in Figure 3.12. The average CC in Figure 3.9 is 0.73; however, the average CC for receivers from 100 to 500 is 0.93. The CC for receivers from 100 to 500 in Figure 3.12 is higher than the one presented in Figure 3.6 for receivers from 100 to 500; however, the CC for receivers from 0 to 600 in Figure 3.12 is lower than the CC presented in Figure 3.6. Using a constant velocity model for the iterative scheme, we retrieve just as accurate Green's function as with using the smoothed velocity model for the Marchenko focusing algorithm for receivers close to the virtual source location, however, as the offset (or the horizontal extent of the model) increases, the accuracy of the retrieved Green's function decreases for the constant velocity model.

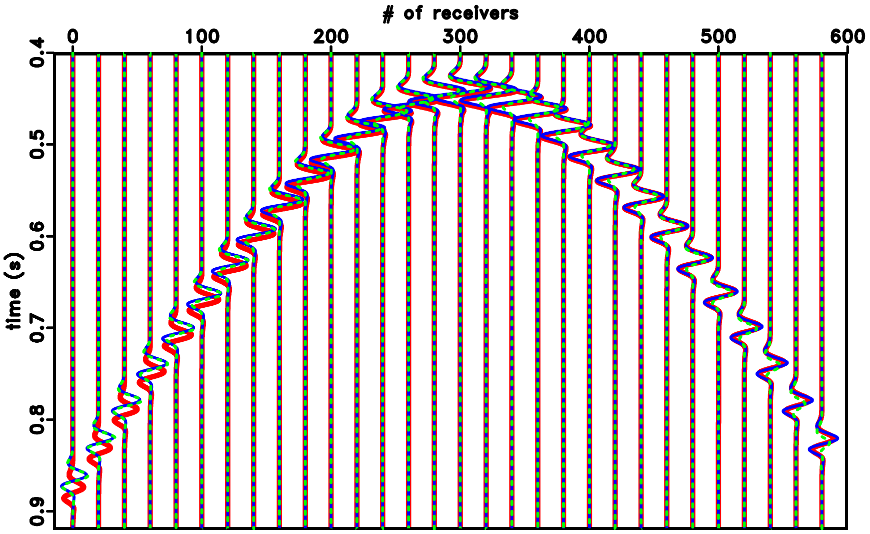


Figure 3.13 Comparison of the modeled direct waves using the velocity models shown in Figure 3.7 (thin blue lines) and Figure 3.10 (thick red lines), overlain with the direct wave modeled using the true velocity model in Figure 3.1(a) (dashed green lines). The traces have been multiplied by $exp(2t)$.

As described and shown in Sections 2 and 3, we model the direct wave using the background velocity model and start the iterative scheme. To evaluate the differences only in the modeled direct waves using different velocity models (Figure 3.7 and Figure 3.10), we show in Figure 3.13 the comparison of the modeled direct waves using the velocity models shown in Figure 3.7 (thin blue lines) and Figure 3.10 (thick red lines), overlain with the direct wave modeled using the true velocity model in Figure 3.1(a) (dashed green lines). For the receivers between 100 and 500 in Figure 3.13, the modeled direct waves are almost identical for the less smoothed velocity model (thin blue lines) and constant velocity model (thick red line) with the true velocity model (dashed green line). This high similarity in the modeled direct waves also produces high-accuracy CCs (Figure 3.9 and Figure 3.12).

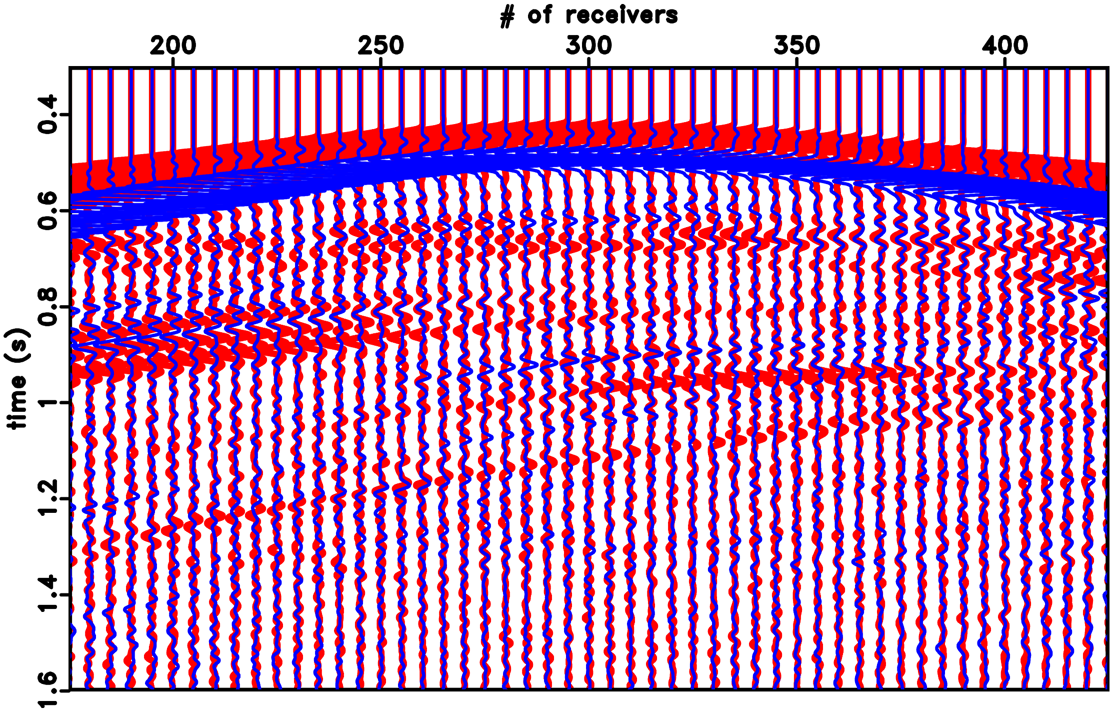


Figure 3.14 The retrieved Green's function using the constant background velocity model with 10% error (blue lines) and the modeled Green's function (red lines). The traces have been multiplied by $exp(2t)$ to emphasize the match for the later times.

We further test the sensitivity of the Marchenko method to the velocity model by using a 10% slower velocity model than the one shown in Figure 3.10 and assume the constant velocity as 1.5 km/s. We show the retrieved Green's function using the constant background velocity model with 10% error (thin blue lines) and the modeled Green's function (thick red lines) superimposed in Figure 3.14 after multiplying traces by $exp(2t)$. The mismatch in time, amplitudes, and phase in Figure 3.14 indicate that the constant value of the velocity should be calculated using the average slowness between the surface and the depth of the focal point, and erroneous constant velocity models will not retrieve the accurate Green's function.

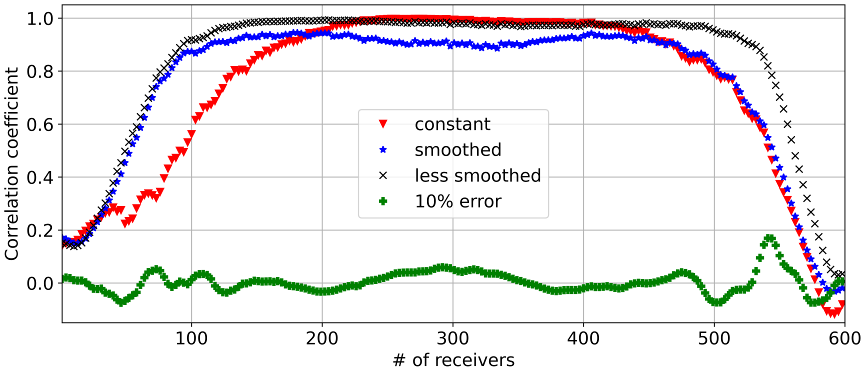
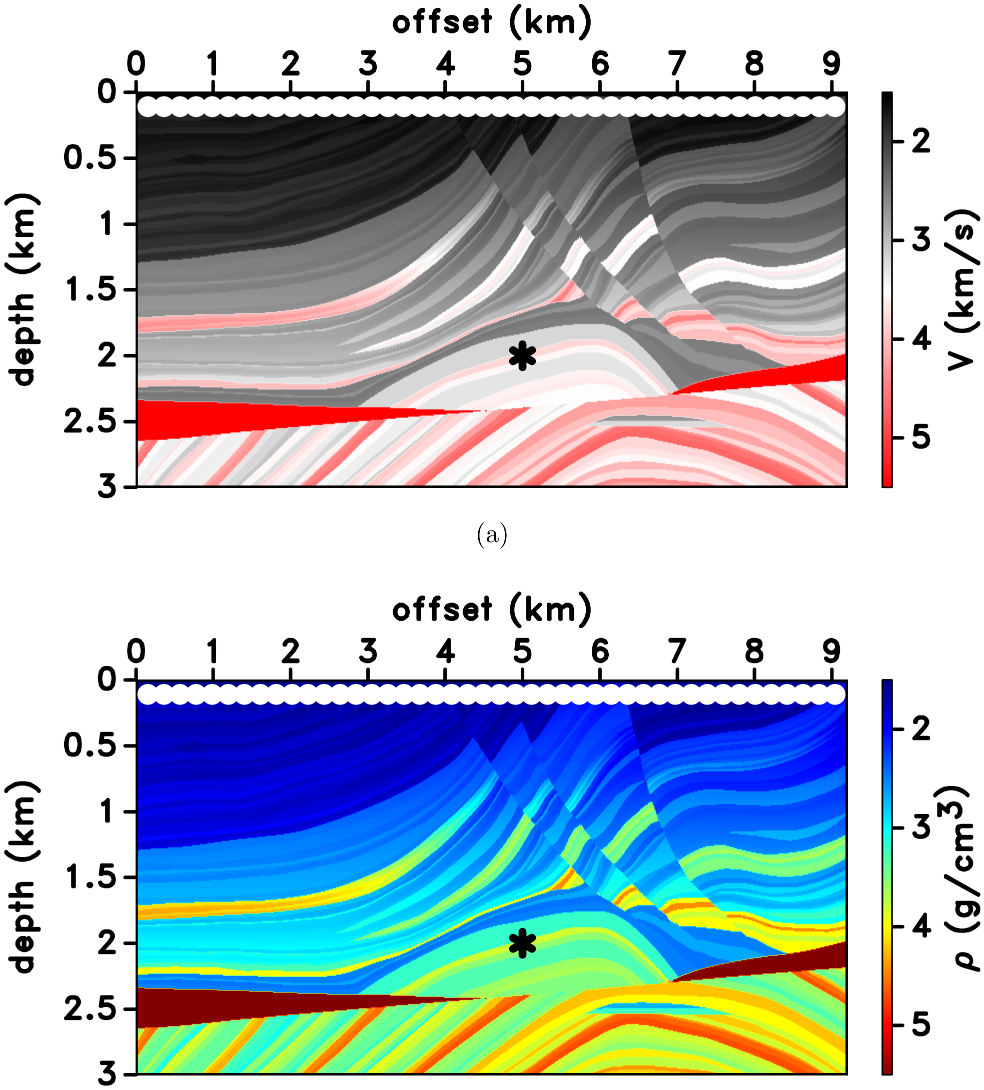


Figure 3.15 Trace-by-trace calculated between the numerically modeled Green's function (Figure 3.5(b), Figure 3.8(b), and Figure 3.11(b)) and the retrieved Green's functions using the velocity models from Figure 3.2, Figure 3.7, Figure 3.10, and the velocity model in Figure 3.10 with 10% error.

Lastly, we present the CCs in Figure 3.15 between the numerically modeled Green's function and the retrieved Green's functions using the velocity models from Figure 3.2, Figure 3.7, Figure 3.10, and the velocity model in Figure 3.10 with 10% slower velocity model using the blue star markers, the grey cross markers, the red triangle markers, and the green plus markers, respectively. We see in Figure 3.15 that the similarity in the modeled direct wave for the Marchenko focusing creates a high accuracy in the retrieved Green's functions. The star, the cross, and the triangle markers show CCs around 0.9; however, the plus marker shows a CC around 0. Therefore, we conclude that the Green's function retrieval using the Marchenko equation successfully retrieves the Green's functions as long as the correct average slowness between the surface and the depth of the focal point is known.

3.5 Refracted Waves

In this section, we investigate the refracted wave presence in the retrieved Green's function using the Marchenko focusing. To model refracted waves in the Green's functions, we use the Marmousi velocity model (Versteeg, 1994) for the numerical experiments in this section. Figure 3.16(a) shows the Marmousi velocity model and Figure 3.16(b) shows the density model of our experiment. The white dots in Figure 3.16 represent every 30th receiver location at the surface and the black asterisk denotes the virtual source location for which the Green's function will be retrieved.



(a)

(b)

Figure 3.16 (a) Marmousi velocity model and (b) density model of the synthetic example. The black asterisk shows the virtual source location and the white dots at the top indicate every 30th source/receiver location.

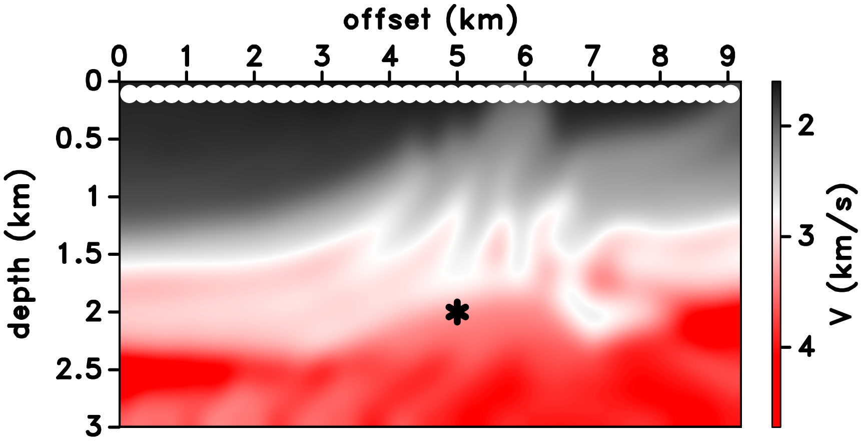
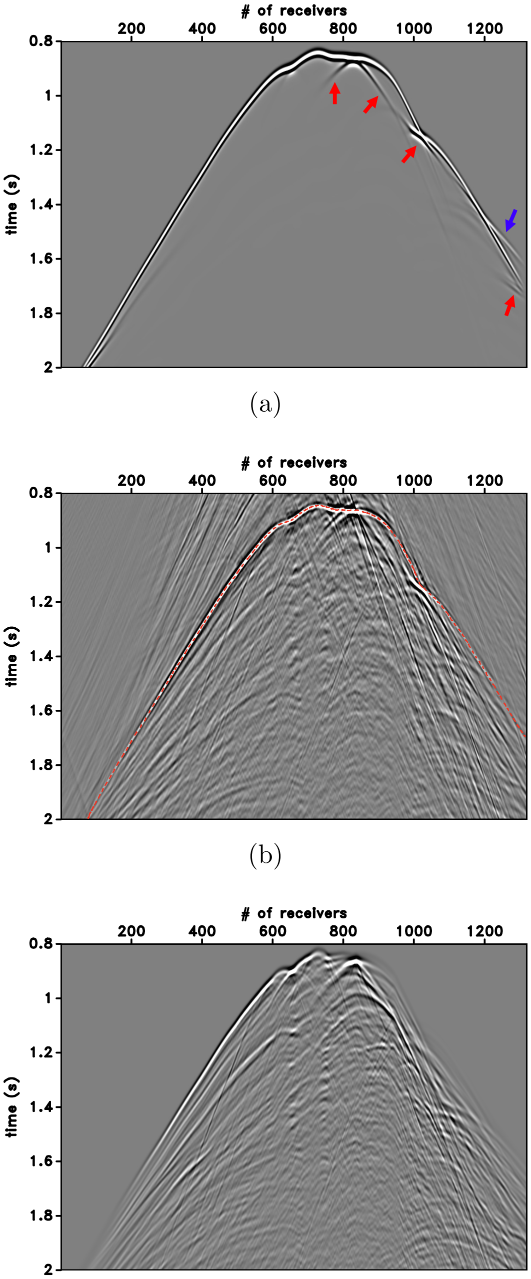


Figure 3.17 Smoothed version of the Marmousi velocity model used for the iterative algorithm. The black asterisk shows the virtual source location and the white dots at the top indicate every 30th source/receiver location.

In order to start the iterative algorithm, we model the direct wave using the smoothed background velocity model shown in Figure 3.17, and the modeled direct wave is shown in Figure 3.18(a). The red arrows in Figure 3.18(a) point to some of the triplicated arrivals and the blue arrow points out the refracted wave. After following the iterative scheme, we show the retrieved Green's function after four iterations in Figure 3.18(b), and the numerically modeled Green's function for the virtual source location in Figure 3.18(c). The red dashed curve in Figure 3.18(b) indicates the arrival of the direct wave (including some triplicated waves), and the waves before the red dashed curve in Figure 3.18(b) can be removed by applying a muting function. The main difference in Figure 3.18 between the retrieved and the modeled Green's functions (Figure 3.18(b) and Figure 3.18(c), respectively) occurs between the receivers 800 and 1320 around the arrival time of the direct wave (around the red dashed curve in Figure 3.18(b)). The numerically modeled Green's function (Figure 3.18(c)) contains refracted and horizontally propagating events recorded between the receivers 800 and 1320 before the arrival time of the direct wave (also indicated with the red dashed curve in Figure 3.18(b)); however, the refracted and horizontally propagating events are not present before the arrival time of the direct wave in the retrieved Green's function (Figure 3.18(b)).



(a)

(b)

(c)

Figure 3.18 (a) Modeled direct wave using the smoothed version of the Marmousi velocity model given in Figure 3.17. The red arrows show some of the triplicated arrivals and the blue arrow shows the refracted wave. (b) Retrieved Green's function using the Marchenko focusing using the smoothed version of the Marmousi velocity model. The red dashed curve shows the arrival of the direct wave (including some triplicated waves), approximately. (c) Numerically modeled Green's function.

To further investigate the presence of the refracted wave in the Marchenko focusing, we use a less smoothed version of the Marmousi velocity model than the one shown in Figure 3.17 which presents more detailed subsurface information. Figure 3.19 shows the less smoothed Marmousi model and the white dots show every 30th receiver location and the black asterisk denotes the virtual source location. The modeled direct wave using the less smoothed Marmousi model is shown in Figure 3.20(a), and the red arrows indicate some of the triplicated arrivals, and the blue arrows indicate the refracted wave. Figure 3.20(b) shows the retrieved Green's function by using the direct wave in Figure 3.20(a), and Figure 3.20(c) shows the numerically modeled Green's function. This time, between the receivers 800 and 1320 and around the direct arrival times, the retrieved Green's function and the numerically modeled Green's function match very well. The refracted wave information in the modeled Green's function is also present in the retrieved Green's function. The less smoothed version of the background velocity model used for the iterative algorithm enables the refracted waves to appear in the retrieved Green's function.

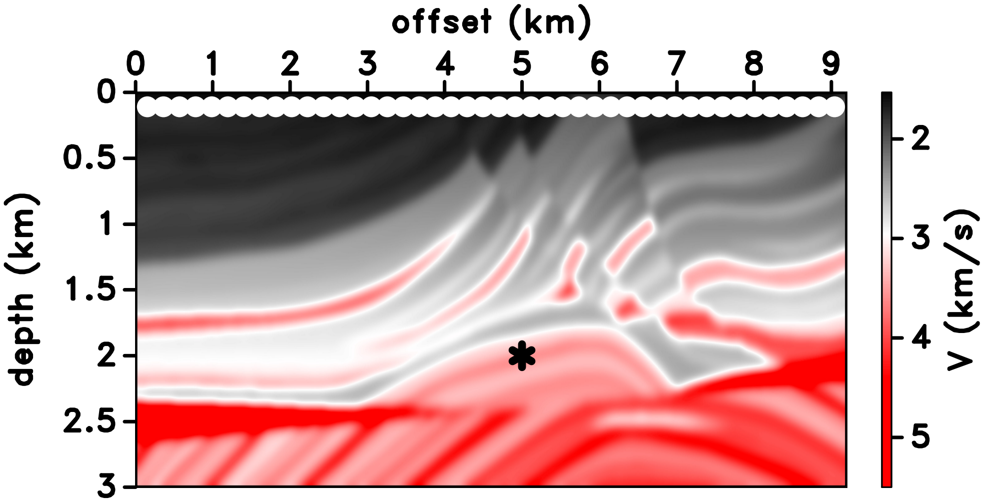
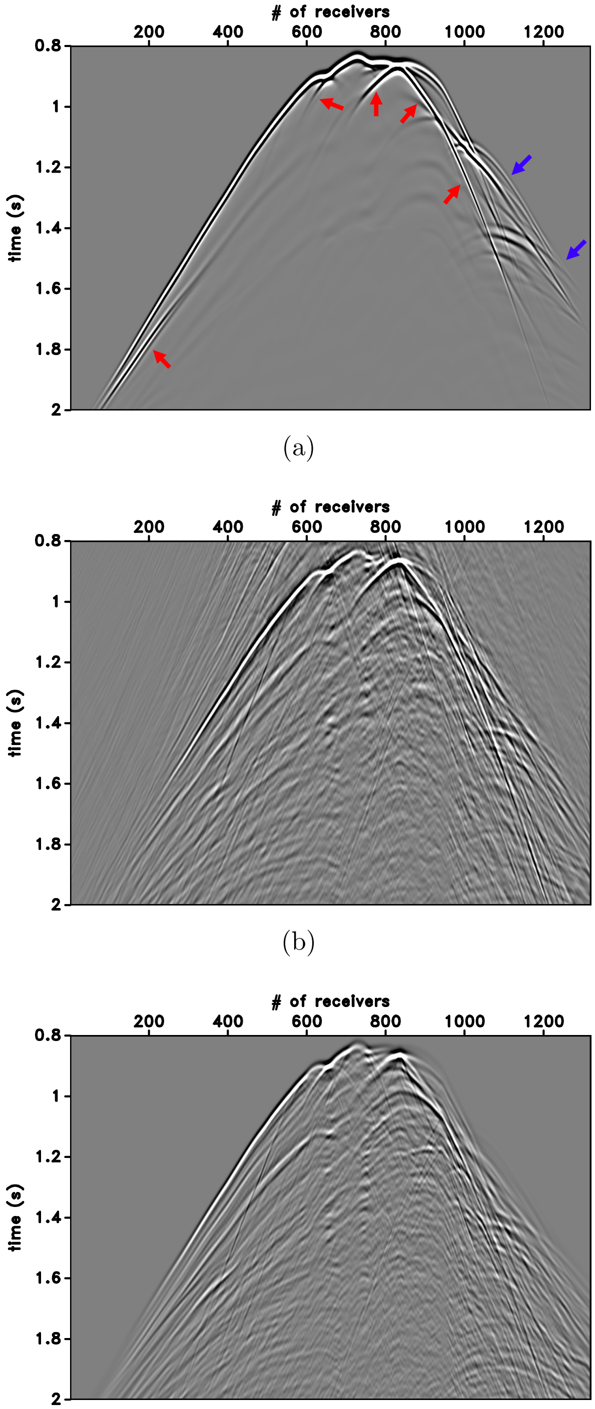


Figure 3.19 Less smoothed version of the Marmousi velocity model used for the iterative algorithm. The black asterisk shows the virtual source location and the white dots at the top indicate every 30th source/receiver location.



(a)

(b)

(c)

Figure 3.20 (a) Modeled direct wave using the less smoothed version of the Marmousi velocity model given in Figure 3.19. The red arrows show some of the triplicated arrivals and the blue arrows show the refracted wave. (b) Retrieved Green's function using the Marchenko focusing using the less smoothed version of the Marmousi velocity model. (c) Numerically modeled Green's function (which is the same wavefield as Figure 3.18(c)).

If we compare the modeled direct waves in Figure 3.18(a) and Figure 3.20(a), the modeled direct wave in Figure 3.18(a) does not include most of the refracted waves (events shown with blue arrows in Figure 3.20(a)). However, the modeled direct wave in Figure 3.20(a) includes the refracted waves shown with the blue arrows. Figure 3.18(b) and Figure 3.20(b) show that the refracted waves modeled using the smooth velocity model are mapped directly in the retrieved Green's functions. In other words, if the refracted waves are modeled using the background velocity model, those events are also present in the retrieved Green's function. But, if the refracted waves are not present in the modeled direct wave, they are not present in the retrieved Green's function. We conclude that the presence of refracted waves only depends on the background velocity model in the Marchenko focusing and it is not a result of the iterations of the Marchenko focusing algorithm.

3.6 Conclusions

We present the Green's function retrieval using Marchenko focusing and investigate the background velocity model dependence of the Marchenko focusing. We compare the retrieved Green's functions for three different background velocity models used for modeling the direct wave. We show that the Marchenko focusing algorithm can retrieve the Green's function with high accuracy. We also investigate the presence of the refracted waves in the retrieved Green's function using the Marmousi velocity model. We show that the refracted waves are incorporated in the retrieved Green's function by the background velocity model used to model the direct wave for the iterative algorithm, and the Marchenko algorithm does not produce the refracted waves.

3.7 Acknowledgments

This work is supported by the Consortium Project on Seismic Inverse Methods for Complex Structures at the Colorado School of Mines. The numerical examples in this paper were generated using the Madagascar software package (Fomel *et al.*, 2013). The research of K. Wapenaar has received funding from the European Research Council (grant

no. 742703).

CHAPTER 4

REMOVING FREE-SURFACE EFFECTS FROM SEISMIC DATA USING CONVOLUTIONAL NEURAL NETWORKS – PART 1: THEORY AND SENSITIVITY ANALYSIS

A paper submitted to *Geophysical Prospecting*

Mert Sinan Recep Kiraz [1,2,3], Roel Snieder[3] & Jon Sheiman

Multiple events reflect more than once in the subsurface, and they are considered as noise since they shadow the useful information about the subsurface. In marine seismic data, the strongest multiples are those which reflect at least once from the free surface (or the air-water interface). The elimination of this type of multiple has been extensively studied and numerous approaches have been used to suppress free-surface multiples. However, free-surface multiple elimination is an expensive seismic data processing step that often results in removing primary events from the seismic data along with the free-surface multiples when primary and multiple events overlap. We present an algorithm to attenuate free-surface multiples using Convolutional Neural Networks (CNNs) and show that data from the CNN-based free-surface multiple elimination results give a correlation coefficient of 0.97 on average with the numerically modeled data without free-surface multiples. We train a network using subsets of the Marmousi and Pluto velocity models, and make predictions using subsets of the Sigsbee velocity model. We demonstrate the robustness of CNNs for free-surface multiple elimination using three numerical examples and show that CNNs are able to attenuate surface-related multiples even in the presence of overlapping primary and multiple events without removing the primary reflections. The network

[1] Primary researcher and author.

[2] Author for correspondence. Direct correspondence to mertkiraz@gmail.com.

[3] Center for Wave Phenomena, Colorado School of Mines, 1500 Illinois St., Golden, CO 80401, USA.

processes a single trace at a time, and therefore is not sensitive to missing traces or to sparse data acquisition, as is the case for ocean-bottom nodes.

4.1 Introduction

Although the presence of multiples in seismic data once was a doubtful question and was not considered as a serious limitation (Dix, 1948; Ellsworth, 1948), there is no doubt now that the removal of multiples from seismic data is one of the most challenging problems in seismic data processing. In general, there are three main categories on which the multiple elimination algorithms rely: (1) Periodicity - so that the multiples can be attenuated if they have a periodic behavior. To address this type of periodic multiple, one can use deconvolution, τ-p deconvolution, deconvolution after stack (Backus, 1959; Diebold & Stoffa, 1981; Yilmaz, 2001). (2) Moveout - so that the multiples can be attenuated if they have a moveout that is sufficiently different than that of the primaries. To address this type of multiple, one can use stacking, f-k filtering (Mayne, 1962; Yilmaz, 2001). (3) Data-driven techniques - so that the multiples can be attenuated based on the information recorded in seismic data. Although there are several ways to address this type of multiple (e.g., Inverse Scattering Multiple Attenuation (ISMA), Interbed Multiple Prediction (IMP)) (Araújo *et al.*, 1994; Weglein *et al.*, 1997), they are out of the scope of this paper, and in this category, we only focus on Surface-Related Multiple Elimination (SRME) method (Verschuur *et al.*, 1992).

Multiple reflections are one of the most challenging problems in recorded marine seismic data since they contaminate the primary reflections. The multiples directly related to the free surface (or the sea surface) have at least one downward reflection at the free surface and are the most significant multiples to address. SRME (Verschuur *et al.*, 1992) is a data-driven technique, and it predicts multiples by exploiting the physical relationship between multiple reflections and their sub-events. The arrival time of a particular surface multiple can be predicted by convolving the traces that contain the multiples' sub-events. SRME has been widely used to predict and iteratively subtract the multiples from the seismic data, and it is one of the most effective tools in seismic processing. However, it is computationally demanding and it does not always guarantee perfect solutions. For instance, during the subtraction step, one can remove the primary reflections from the seismic data in addition to the unwanted multiple reflections because of the overlapping primary and multiple events. Additionally, a profound limitation of SRME is that it requires a dense source and receiver distribution which makes it computationally demanding and which makes missing data gaps a complication that must be dealt with. Although two-dimensional (2-D) versions of SRME can be computationally efficient, they fall short in the presence of complex subsurface requiring three-dimensional (3-D) spatial distribution of source and receiver distribution (Dragoset *et al.*, 2010) (e.g., when multiples with strong cross-line ray path components present in the data). This condition becomes even more difficult to fulfill in 3-D surveys.

There have been several attempts to address free-surface multiple elimination. Riley & Claerbout (1976) present free-surface multiple elimination theory applied to one-dimensional (1-D) data, and they also present an algorithm for surface-generated diffracted multiple elimination. Verschuur *et al.* (1992) show that free-surface multiples can be eliminated by predicting the multiples and subtracting them from the seismic data. Jakubowicz (1998) shows that interbed multiples can be removed using a technique that is an extension of SRME. There have been field data applications of SRME (Dragoset & MacKay, 1993; Dragoset & Jeričević, 1998; Kelamis & Verschuur, 2000; Verschuur *et al.*, 1995).

SRME removes the events that reflect at least once from the free surface (or the air-water interface). To predict the surface-related multiples only, the measured field data are usually used and no information about the subsurface is needed. However, SRME makes great requirements about the data acquisition. For a successful implementation of SRME, some of the requirements are; a dense distribution of the source and receiver wavefields, known source wavelet, interpolation/extrapolation of the data, and available near-offset data. Conventionally, an SRME processing flow follows; (1) Data pre-processing, (2) Prediction of the surface-related multiples, (3) Matching filtering (or estimation of wavelet), (4) Adaptive subtraction.

We present and discuss a CNN-based approach to attenuate free-surface multiples. We train a network using 2-D subsets of Marmousi velocity model (Versteeg, 1994), and Pluto velocity model (Stoughton *et al.*, 2001), and make predictions using three subsets of Sigsbee velocity model (Paffenholz *et al.*, 2002) with salt bodies in the subsurface. As the training input data, we use 2-D data modeled *with* free-surface multiples using the finite-difference forward modeling, and also use 2-D data modeled *without* free-surface multiples using the finite-difference forward modeling as the training output data. Although we use 2-D subsets of the velocity models for the training, we use 1-D convolutional filters to find the optimal weights for the network that relate the input and output datasets. After training the network using the input and output data, we make predictions using three new datasets which were not involved in the training process. We show that CNNs are able to make predictions that give a Correlation Coefficient (CC) of 0.97 on average between the numerically modeled results and the CNN predictions, and when the free-surface multiples overlap with the primary events, the CNN-based solution is able to remove the unwanted free-surface multiple reflections effectively (e.g., CC of 0.95) from the seismic data while preserving the primary reflections.

4.2 CNN Architecture and Training

Machine Learning (ML) is a subset of artificial intelligence and it uses mathematical and statistical methods to let machines and computers improve specified tasks with experience. Deep Learning (DL) is a subset of ML and it is inspired by the structure of our brains and the interconnections between its neurons. DL uses multiple layers of computational units and every layer learns its own representation of the input data (Ekman, 2021).

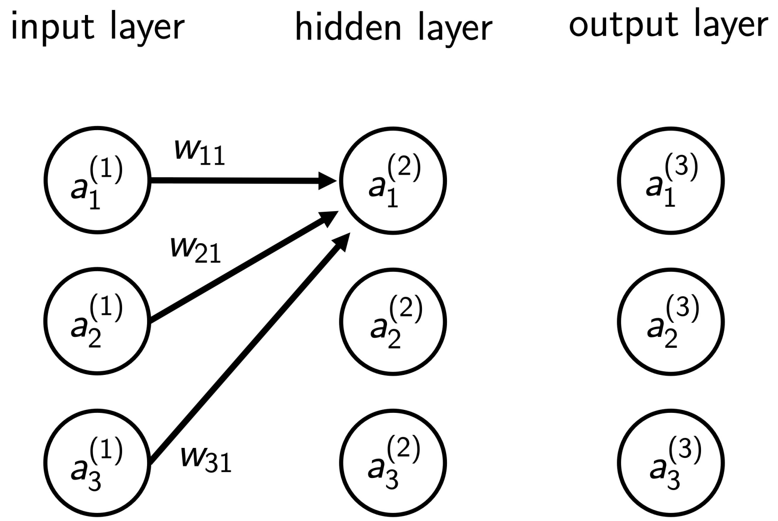


Figure 4.1 A sketch of a simple one-hidden-layer neural network with an input layer, hidden layer, and an output layer. For illustration purposes, the input layer, hidden layer, and the output layer have three neurons.

Figure 4.1 shows a simple one-hidden-layer (and in total three-layer) neural network where the input layer, hidden layer, and the output layer have three neurons to illustrate the propagation of a neuron in the input layer through the network. In Figure 4.1, the black arrows indicate that the three neurons in the input layer are connected to the first neuron in the hidden layer. As the number of hidden layers increases, the network extracts more complex information, and the networks with many hidden layers are called “deep neural networks”.

We obtain the output value of a neuron shown in Figure 4.1 by

$$a_j^k = \sigma\Big(\sum_{i=1}^{N}(a_i^{(k-1)}w_{ij}^k) + b_j^k\Big) \ , \tag{4.1}$$

where N denotes the number of neurons in the previous layer, a_j^k denotes the value a to be calculated of the jth neuron in the kth layer, $a_i^{(k-1)}$ denotes the value a of the ith neuron in the $(k-1)$th layer, w_{ij}^k denotes the weight w of the ith neuron which is connected to the jth neuron in the kth layer, b_j^k denotes the bias b of the jth neuron in the kth layer, and σ denotes the activation function. *Training* is a process in ML such that we adjust the weights, w, and biases, b, by an iterative data-fitting process. During the start of the training process, a set of initial adjustable random weights and biases are assigned. As the training progresses, the weights and biases are updated. The activation function introduces the non-linearity in machine learning algorithms. If there is no non-linearity between layers, using several linear transformations of stack of layers, we only get linear transformations and are not able to solve complex problems (Ekman, 2021; Géron, 2019).

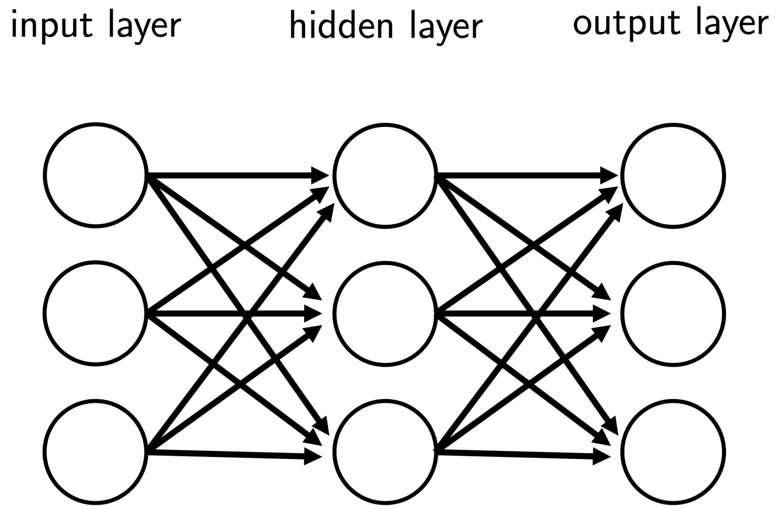


Figure 4.2 A sketch of a fully connected neural network with an input layer, hidden layer, and an output layer. The input layer, hidden layer and the output layer have three neurons.

When each neuron of a layer is connected to every neuron in the other layer, the network is called a fully connected network. Figure 4.2 shows an example of a fully connected network with an input layer, hidden layer, and an output layer where the input layer, hidden layer and the output layer have three neurons. Convolutional neural networks (CNNs) are important building blocks in DL methods inspired by the brain's visual cortex and they have been widely incorporated in image recognition studies since the 1980's (Géron, 2019). CNNs are different from other DL algorithms because the neurons in the CNN architecture are connected to the region of pixels/neurons (receptive fields) in the input image as opposed to every individual pixels/neurons (for more detailed information about CNNs, we refer readers to Appendix A).

During the training phase, once an output of a network (or the output of each neuron in the output layer) is obtained, we compare the output of the network to the desired output (or the ground truth) by defining an error function. One of the most commonly used error functions is the Mean-Squared Error (MSE) metric that measures the difference between the predicted and the ground truth results,

$$MSE = \frac{1}{m}\sum_{i=1}^{m}(\hat{y}_i - y_i)^2 \ , \tag{4.2}$$

where $\hat{y}_i$ is the output of the network, y_i is the ground truth data for a given input data during the training, and m is the number of training examples. To update the parameters, the MSE function is minimized commonly using the gradient descent method by calculating the gradient of the error function with respect to the weights, and training CNNs is commonly done by backpropagation where the choice loss function and gradient descent optimization algorithm play essential roles (Bishop, 2006; Ekman, 2021; Géron, 2019).

Different DL/CNN methods and their subsets have been used in exploration seismic studies to address different problems. Siahkoohi *et al.* (2018) utilize CNNs to remove the free-surface multiples and numerical dispersion; Das *et al.* (2019) use CNNs to obtain an elastic subsurface model using recorded normal-incidence seismic data; Wu *et al.* (2019) use CNNs for 3-D seismic fault segmentation; Almuteri & Sava (2023) use CNNs to address to ghost removal from seismic data; Kiraz & Snieder (2022) utilize CNNs for 1-D wavefield focusing where the solution of the Marchenko equation is not needed to retrieve the Green's function once the network is trained. Recently, CNNs have been used to tackle free-surface multiples in various ways (Ovcharenko *et al.*, 2021; Siahkoohi *et al.*, 2018, 2019; Tao *et al.*, 2021; Zhang *et al.*, 2021). In this paper, our network predicts a single trace at a time, and this makes our algorithm insensitive to the data gaps in the acquisition and does not require a meticulous data-filling pre-processing step.

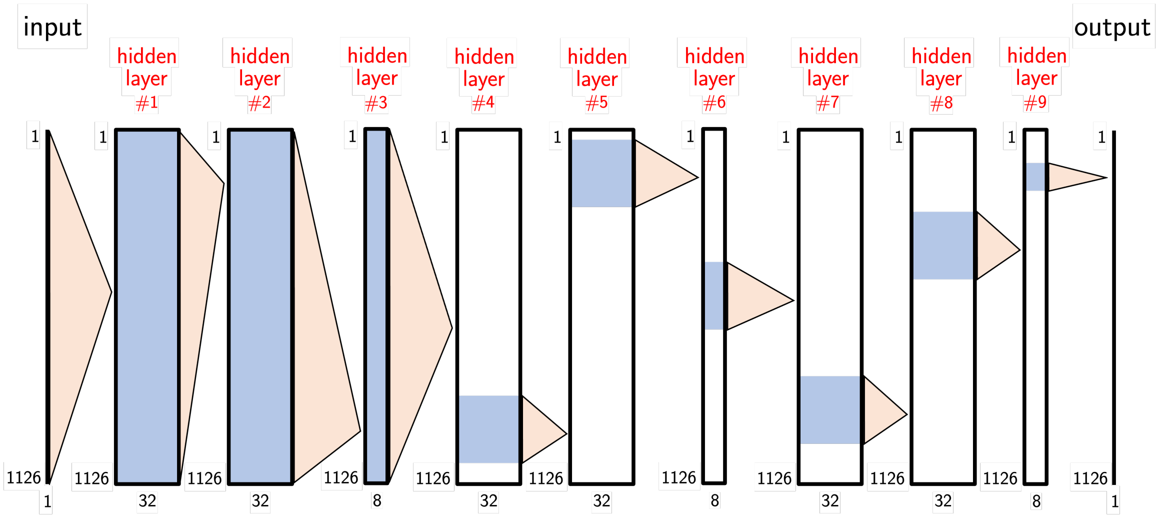


Figure 4.3 Cartoon of the 1-D network used for surface-related multiple elimination on 2-D data. The height of each trace corresponds to the sample size in our experiment which is 1126 samples. The numbers located below each trace represents the number of convolutional filters used (e.g., 1, 32, 32, 8, ...). Each blue area visually represents the kernel size used for each layer in the network (e.g., 1125, 101, 51).

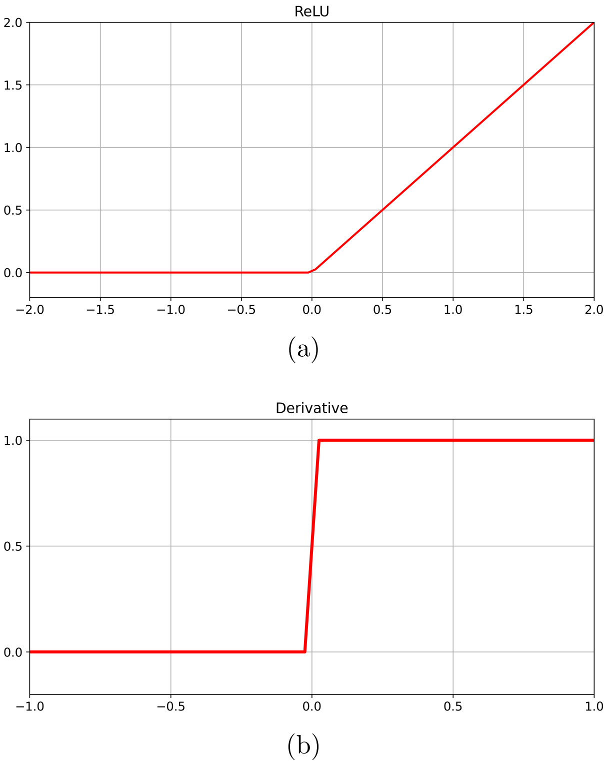


Figure 4.4 (a) ReLU activation function. (b) Derivative of the ReLU activation function.

4.2.1 Network Architecture

For the prediction of the free-surface multiple attenuated data, the CNN architecture we use consists of an input layer, nine hidden 1-D convolutional layers, and an output layer. The first and second, fourth and fifth, and seventh and eighth hidden layers have 32 filters; the third, sixth, and ninth hidden layers have eight filters (see Appendix A for more information about the action of the filters). The length of the 1-D convolutional window (kernel size) for the first three hidden layers is 1125 samples; for the following five hidden layers is 101 samples; and for the last hidden layer is 51 samples. Overall, the network consists of 1,773,697 total parameters all of which are trainable. Figure 4.3 shows a cartoon of the network architecture used for this study.

For the input layer and the hidden convolutional layers, we use the Rectified Linear Unit (ReLU) activation function (Ekman, 2021) which computes the function $f(x) = max(0, x)$. The ReLU activation function helps optimize the model easily using the gradient-based methods (Géron, 2019). While minimizing the error function, a backpropagation (or a backward pass) (Ekman, 2021) is used to update the weights of the network. The derivative of the activation function is calculated at this step, and the derivative of the ReLU activation function does not suffer from "vanishing gradients" phenomena (Ekman, 2021; Géron, 2019). Figure 4.4(a) shows the ReLU activation function and Figure 4.4(b) shows its derivative. Lastly, the output layer does not have an activation function.

For all the layers, we use a stochastic gradient descent optimization algorithm with momentum along with the mean-squared loss function as the objective function (Géron, 2019). To prevent our model from overfitting, and for an accurate evaluation, we split the input data into training dataset, testing dataset, and validation dataset using 60%, 20%, and 20% ratios, respectively. The testing data are used to monitor the network training, and adjust hyperparameters if needed. We tune the hyperparameters using this testing dataset and compare the output to the ground truth. One should be careful at this step because we want the network to learn the relationship between the input and output training data pairs, and if we tune the hyperparameters only according to the testing dataset, the network would memorize the relationship between the input and output data pairs instead of learning. This would also introduce the risk of overfitting, and the network would perform poorly when we use a different dataset which is not involved in the training process. After defining the network parameters (or hyperparameters), we use 100 epochs (iterations) to train the network. Figure 4.5 shows the learning curve of our network where the blue and red curves denote validation and training data, respectively.

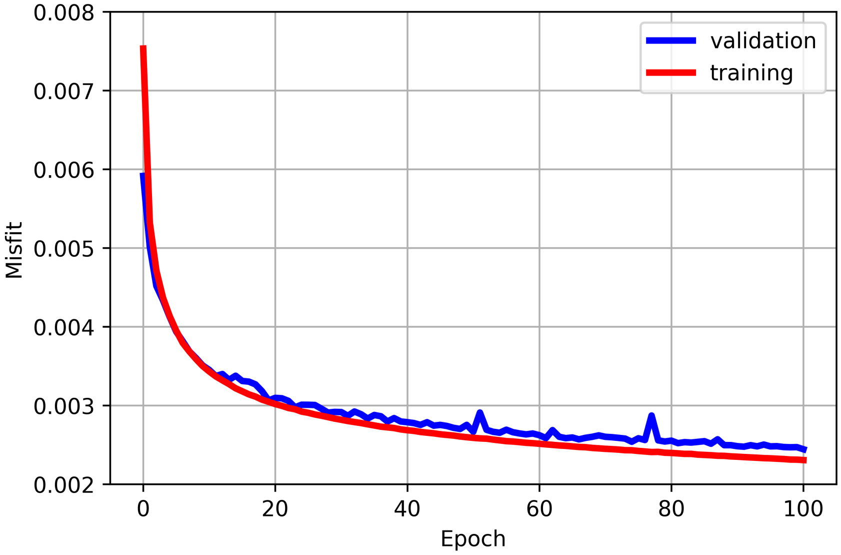
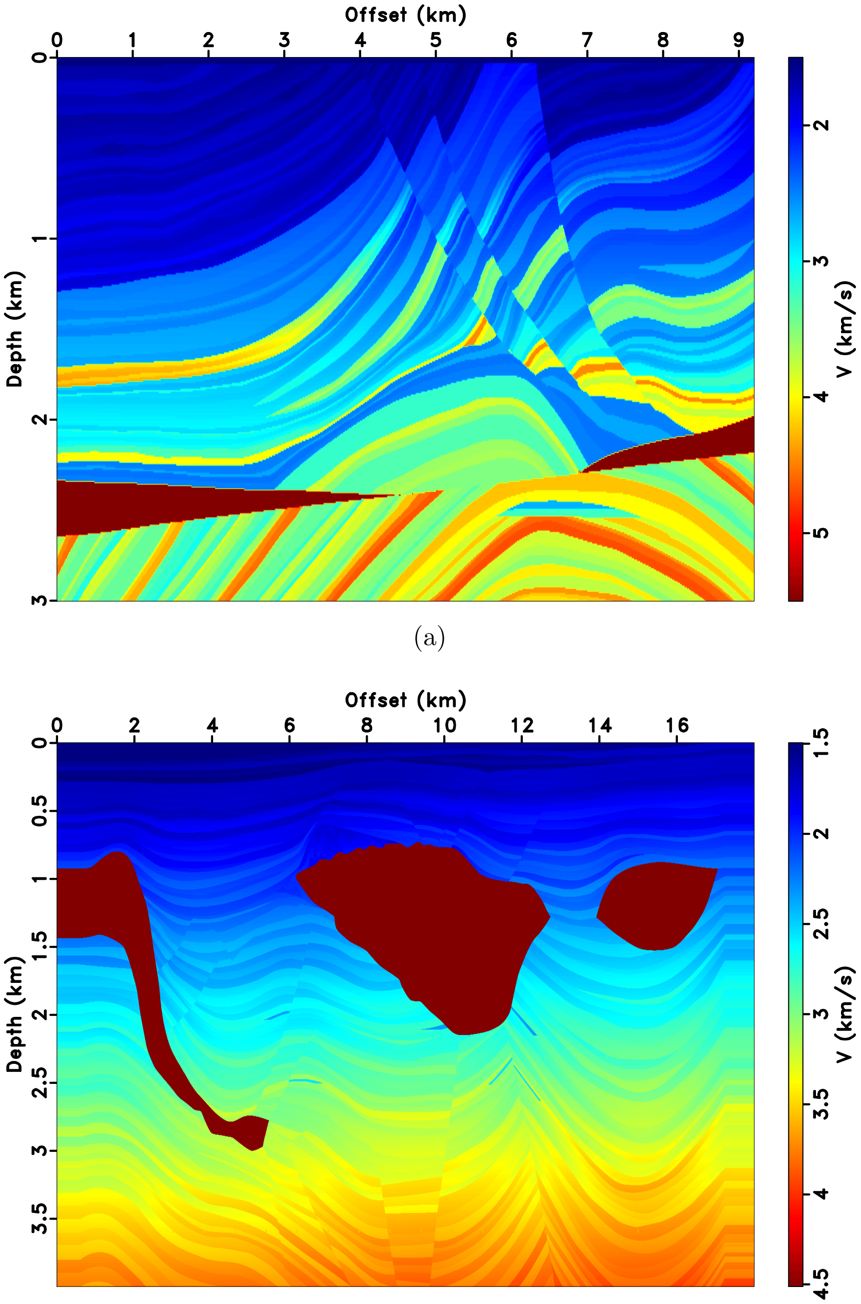


Figure 4.5 Learning curve for the convolutional neural network trained using the training and validation datasets.

The choice of hyperparameters in this paper is based on trial and error. After changing a set of hyperparameters, we evaluate the performance of the network by comparing the predictions to the expected results, by observing the training and validation curves in Figure 4.5) to make sure that we are not overfitting the data (one of the most common ways to check overfitting is to observe the training and validation curves during training (Ekman, 2021; Géron, 2019)).



(a)

(b)

Figure 4.6 (a) Marmousi velocity model. (b) Pluto velocity model.

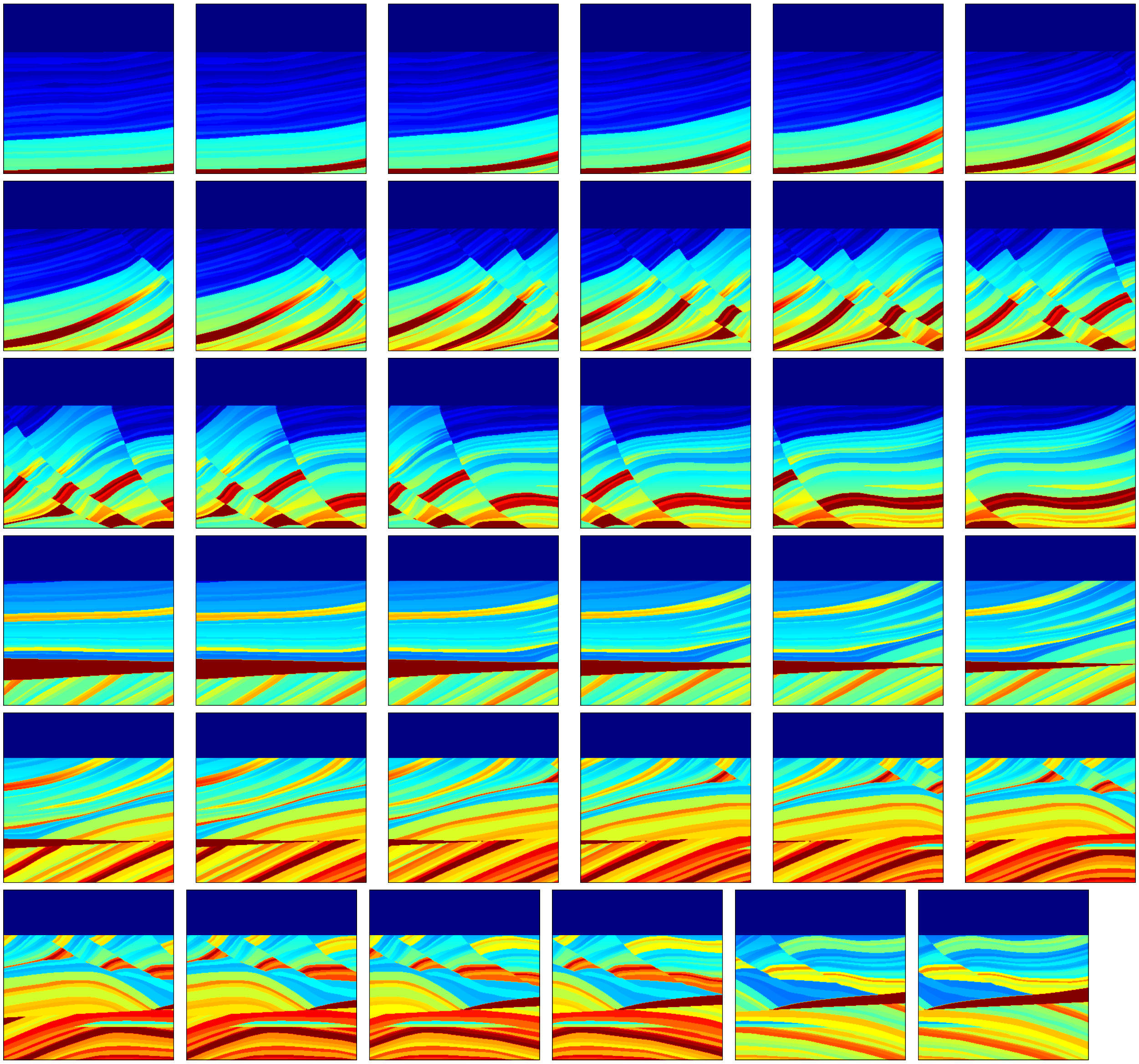


Figure 4.7 Subsets of Marmousi velocity model used for training.

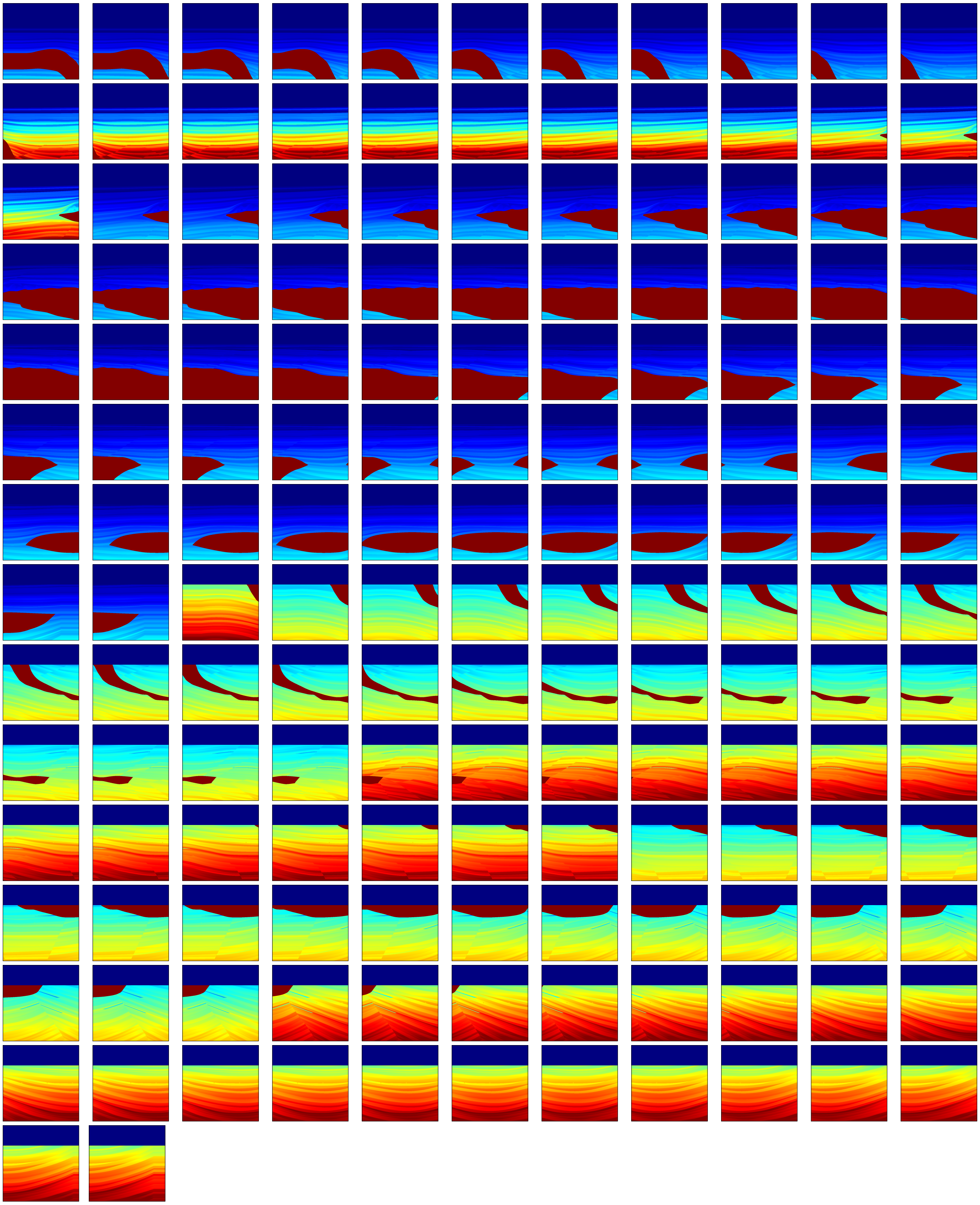
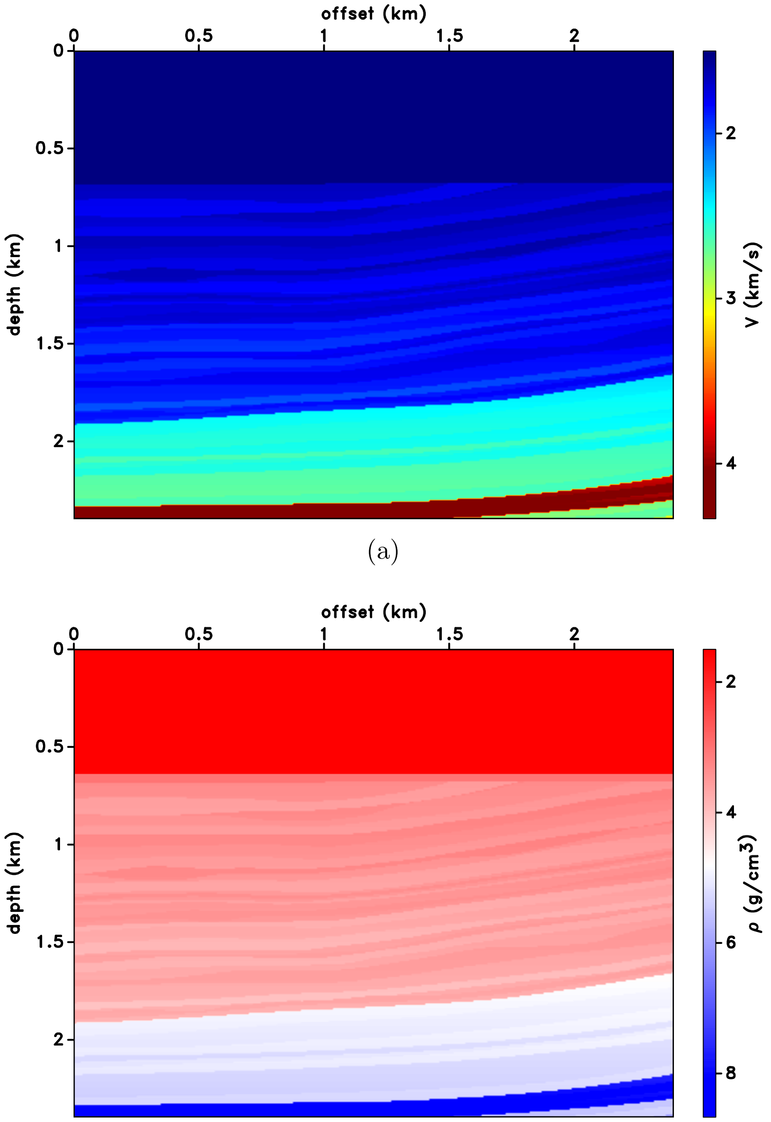


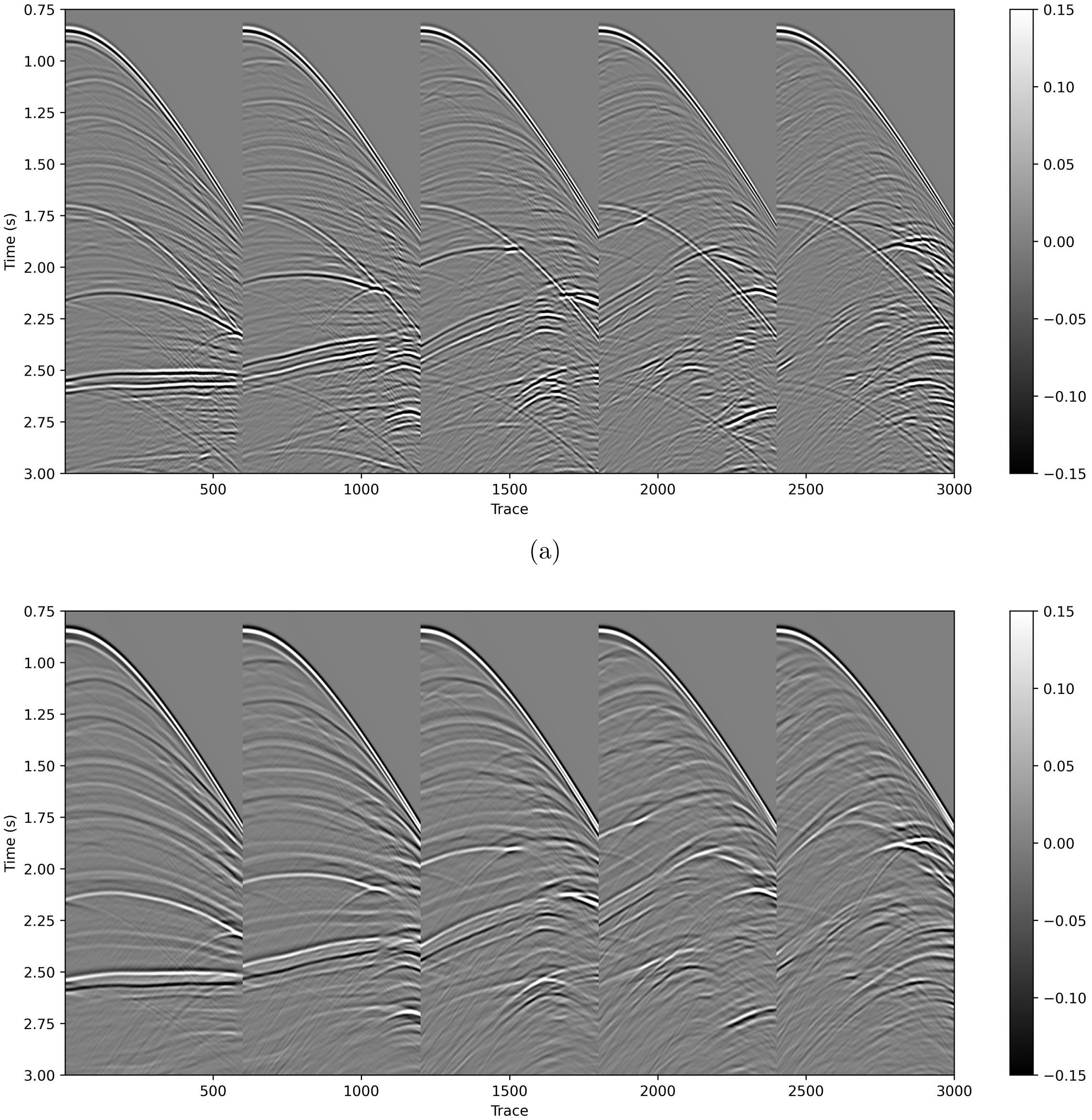
Figure 4.8 Subsets of Pluto velocity model used for training.



(a)

(b)

Figure 4.9 (a) One of the training velocity sub-models extracted from Marmousi velocity model. (b) One of the training density sub-models scaled from the velocity model in (a).



(a)

(b)

Figure 4.10 (a) 5 of 192 shot gathers of the input training data (*with* a free surface). (b) 5 of 192 shot gathers of the output training data (*without* a free surface).

4.3 Network Training

We create 192 input shot gathers where each gather consists of 600 traces (total of 115,200 input traces) using 2-D finite-difference forward modeling through the models with free-surface effects, and 192 output shot gathers where each gather consists of 600 traces (total of 115,200 output traces) using 2-D finite-difference forward modeling through the models without free-surface effects. Throughout the forward modeling for each model, 600 receivers are placed along 2.4 km horizontal extent with the receiver sampling of 4 m, and are located at 8 m depth. We use one source per model, and the coordinates of the source coordinates are (x_s, z_s) = (25,7) m.

For the training model selection, we use the Marmousi and Pluto velocity models. Figure 4.6(a) shows the Marmousi velocity model and Figure 4.6(b) shows the Pluto velocity model. We extract 36 sub-models from the Marmousi model and 156 sub-models from the Pluto model. We also add 640 m of water layer to the top of each model with 1500 m/s velocity. Each one of 192 models used for training has 2.4 km horizontal extent, and 2.4 km depth which includes the water column, 600 receivers, and a single shot location. Figure 4.7 shows 36 sub-models of the Marmousi velocity model used for the training, and Figure 4.8 shows 156 sub-models of the Pluto velocity model used for the training. For each velocity model, we use a density model that is a scaled version of the velocity model by a factor two (and the units are converted from km/s to g/cm^3), and has a water layer of 640 m with the velocity of 1500 m/s. Figure 4.9 shows 1 of 192 velocity and density models used during the data generation for training. Figure 4.9(a) shows a training velocity sub-model extracted from Marmousi velocity model. Figure 4.9(b) shows a training density sub-model scaled from the velocity model presented in Figure 4.9(a). Scaling the velocity model by two and adding the water layer into the density model creates a range of reflection coefficients that changes from approximately 0.33 to 0.89. The minimum reflection coefficient at the water bottom is set to mimic the seafloor conditions which is comparable to the seafloor reflection coefficient given by Kim *et al.* (2004). Figure 4.10 shows 5 of 192 shot gathers of the input and output data displayed next to each other used for training after removing the direct arrival. We remove the direct arrival from the recorded data during the training and prediction steps. Figure 4.10(a) shows 5 of 192 shot gathers of the data modeled with free-surface effects which are used as the training input, and Figure 4.10(b) shows 5 of 192 shot gathers of the data modeled without free-surface effects which are used as the training output. Figure 4.10(a) shows the first appearance of free-surface multiples at each shot gather starting around 1.7 s in the first trace of a shot gather, and ending around 2.3 s in time. Although not as strong, the shot gathers also have other types of multiples (e.g., top and/or bottom salt peg-legs, internal).

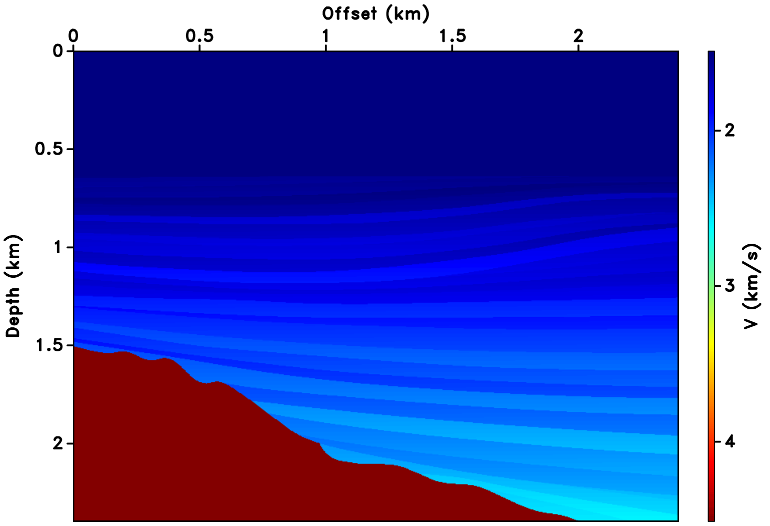
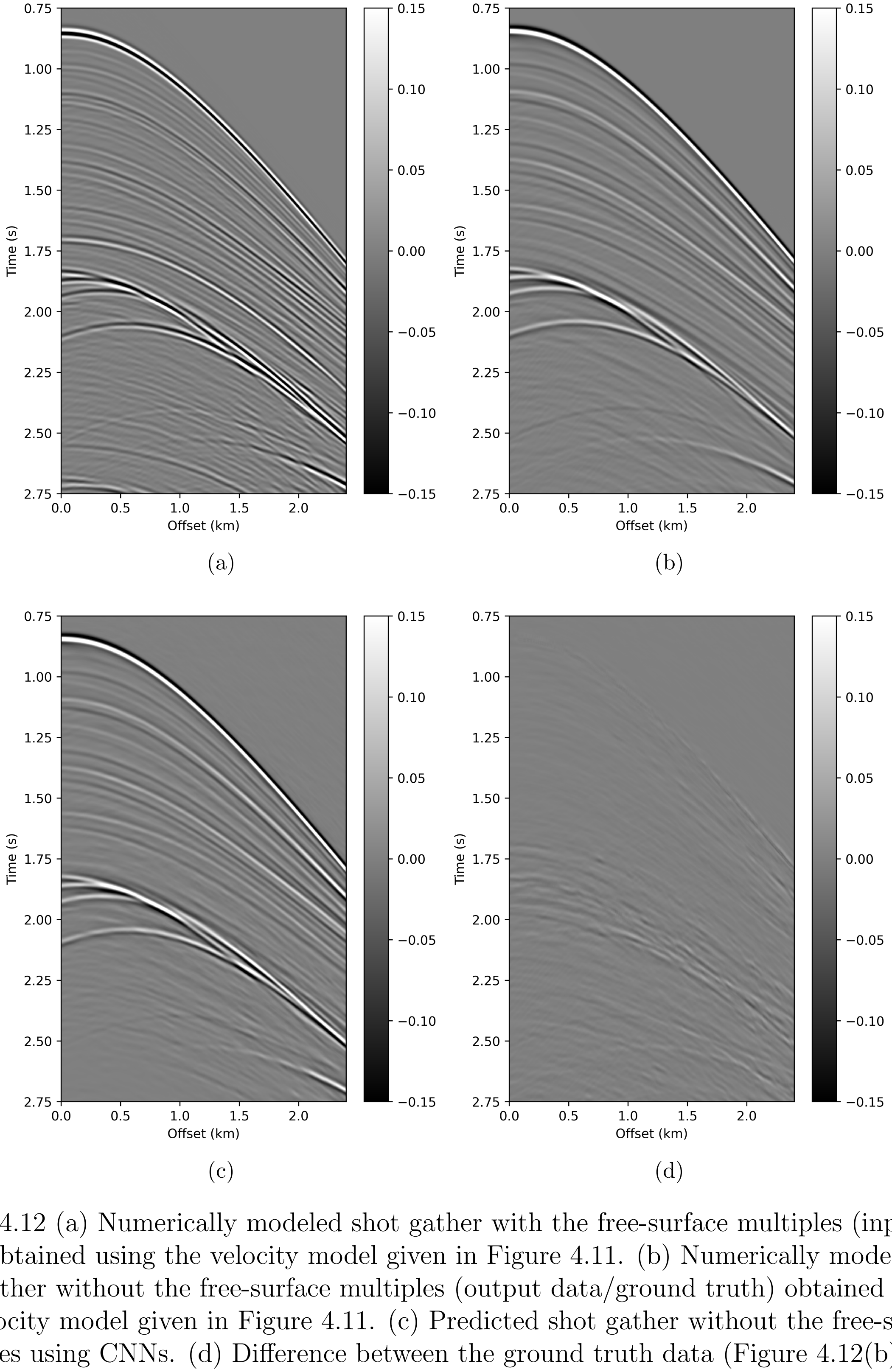


Figure 4.11 A sub-model of Sigsbee velocity model used for the prediction.

4.4 Numerical Examples and CNN Prediction

For the numerical simulation, we use three different sub-models of the Sigsbee velocity model which are not used in the training process and are unknown to the network to demonstrate our CNN-based solution for free-surface multiple elimination.

For the first example, Figure 4.11 shows a subset of the Sigsbee velocity model which consists of relatively flat subsurface structures along with a part of a salt body located at the deeper parts of the model. Similar to the training models, 600 receivers are placed along 2.4 km horizontal extent with the receiver sampling of 4 m at a depth of 8 m. The coordinates of the source coordinates are (x_s, z_s) = (25,7) m. The water column is 640 m deep with a velocity of 1500 m/s and a density of 1 gr/cm^3. The sea bottom for this example also has a reflection coefficient of approximately 0.33.



(a)

(b)

(c)

(d)

Figure 4.12 (a) Numerically modeled shot gather with the free-surface multiples (input data) obtained using the velocity model given in Figure 4.11. (b) Numerically modeled shot gather without the free-surface multiples (output data/ground truth) obtained using the velocity model given in Figure 4.11. (c) Predicted shot gather without the free-surface multiples using CNNs. (d) Difference between the ground truth data (Figure 4.12(b)) and the CNN prediction (Figure 4.12(c)).

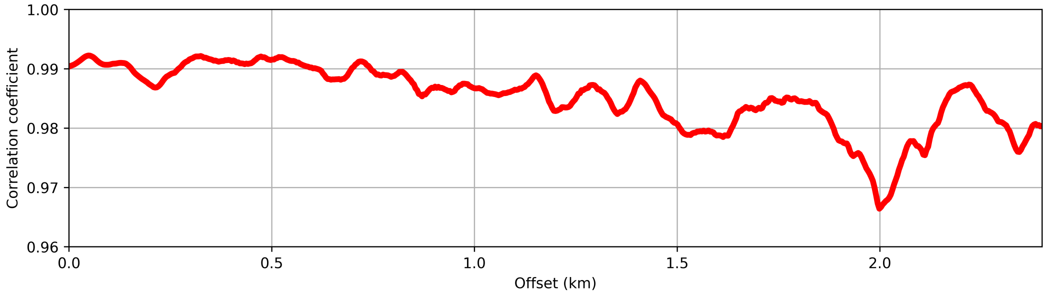
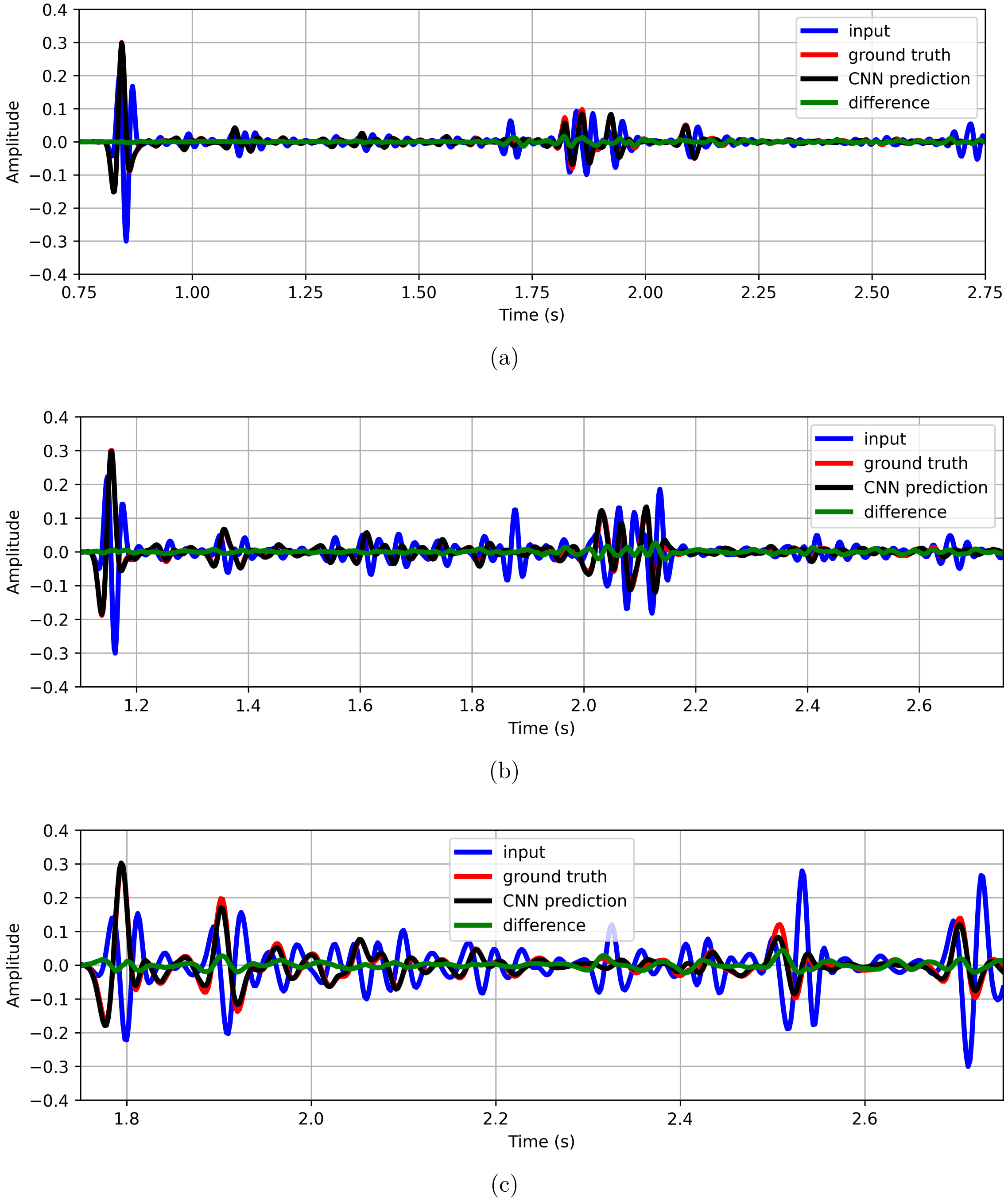


Figure 4.13 Trace-by-trace calculated correlation coefficient between the ground truth data (Figure 4.12(b)) and the CNN prediction (Figure 4.12(c)).

Figure 4.12(a) shows the data modeled with free-surface effects using the velocity model shown in Figure 4.11, and Figure 4.12(b) shows the waveforms modeled without free-surface effects using the velocity model shown in Figure 4.11. Figure 4.12(a) shows the input data, and starting around 1.7 s in time at 0 km in offset, and ending around 2.3 s in time at 2.4 km in offset, the first multiple caused by the free surface is located. Additionally, starting around 2.55 s in time at 0 km in offset, and ending around 2.75 s in time at 1.5 in km offset, another multiple caused by the free surface is present. Although not as strong as these two events, there are other multiples arriving after the first multiple which are caused by the free surface. Figure 4.12(b) shows the desired output, the ground truth. The free-surface multiples described in Figure 4.12(a) are not present in this figure because of the excluded free-surface effects. For a successful CNN network training, we expect the CNN predictions to be similar to the ground truth (Figure 4.12(b)). After training the network, Figure 4.12(c) shows our CNN-based free-surface multiple elimination results. Figure 4.12(d) shows the difference between the ground truth data (Figure 4.12(b)) and our CNN-based free-surface multiple elimination results (Figure 4.12(c)).

We measure the accuracy of the CNN-based free-surface multiple elimination results by calculating trace-by-trace CCs between the ground truth and the CNN prediction. Figure 4.13 shows the CCs between the ground truth (Figure 4.12(b)) and the CNN prediction (Figure 4.12(c)), and the average CC is 0.98.



(a)

(b)

(c)

Figure 4.14 Trace comparison of the input (solid blue line), ground truth (solid red line), CNN prediction (solid black line), and the difference between the ground truth and the CNN prediction (solid green line) data from the zero-, mid-, and far-offsets, respectively, extracted from the data shown in Figure 4.12. (a) Comparison of the zero-offset traces. (b) Comparison of the mid-offset (offset 1.2 km) traces. (c) Comparison of the far-offset (offset 2.3 km) traces.

Figure 4.14 shows several trace comparisons of the input data (solid blue line), output data (or ground truth) (solid red line), the CNN predictions (solid black line), and the difference between the ground truth and the CNN prediction (solid green line) shown in Figure 4.12 for the zero-, mid-, and far-offsets. Figure 4.14(a) shows the trace used as an input to the network which is the zero-offset trace extracted from Figure 4.12(a) (solid blue line), the ground truth that is the expected result from the CNN prediction which is the zero-offset trace extracted from Figure 4.12(b) (solid red line), the predicted zero-offset trace using our CNN-based free-surface multiple elimination operator extracted from Figure 4.12(c) (solid black line), and the difference zero-offset trace between the ground truth and our CNN-based free-surface multiple elimination operator extracted from Figure 4.12(d) (solid green line). In Figure 4.14(a), the two strongest multiples caused by the free surface in the input trace (solid blue line) at times around 1.7 s and 2.7 s have been successfully eliminated in the CNN prediction trace (solid black line) which matches the ground truth trace (solid red line). Figure 4.14(b) shows the mid-offset (offset 1.2 km) trace used as an input to the network from Figure 4.12(a) (solid blue line), the ground truth that is the expected result from the CNN prediction which is the mid-offset (offset 1.2 km) trace extracted from Figure 4.12(b) (solid red line), the predicted mid-offset (offset 1.2 km) trace using our CNN-based free-surface multiple elimination operator extracted from Figure 4.12(c) (solid black line), and the difference mid-offset trace between the ground truth and our CNN-based free-surface multiple elimination operator extracted from Figure 4.12(d) (solid green line). In Figure 4.14(b), the strongest multiple caused by the free surface in the input trace (solid blue line) at time around 1.85 s has been successfully eliminated in the CNN prediction trace (solid black line) which matches the ground truth trace (solid red line). Lastly, Figure 4.14(c) shows the far-offset (offset 2.3 km) trace used as an input to the network from Figure 4.12(a) (solid blue line), the ground truth that is the expected result from the CNN prediction which is the far-offset (offset 2.3 km) trace extracted from Figure 4.12(b) (solid red line), the predicted far-offset (offset 2.3 km) trace using our CNN-based free-surface multiple elimination operator extracted from Figure 4.12(c) (solid black line), and the difference far-offset trace between the ground truth and our CNN-based free-surface multiple elimination operator extracted from Figure 4.12(d) (solid green line). In Figure 4.14(c), the strongest multiple caused by the free surface in the input trace (solid blue line) at time around 2.35 s has been successfully eliminated in the CNN prediction trace (solid black line) which matches the ground truth trace (solid red line). The difference traces for the near-, mid-, and far-offset traces indicate a small amplitude mismatch between the ground truth and the CNN-based free-surface multiples. Figure 4.14 shows that the CNN-based free-surface multiple elimination is able to attenuate the free-surface multiples despite their different arrival times due to the offset change. The CNN-based free-surface multiple elimination is able to preserve the primary events after removing the unwanted multiples from the data. Additionally, the phase change in the input trace is successfully accounted for through the CNN prediction. The presence of the free surface changes the phase and the amplitude spectrum by the interference with the ghost wave. Thus, the algorithm suppresses multiples and does the deghosting simultaneously (Amundsen & Zhou, 2013; Riley & Claerbout, 1976).

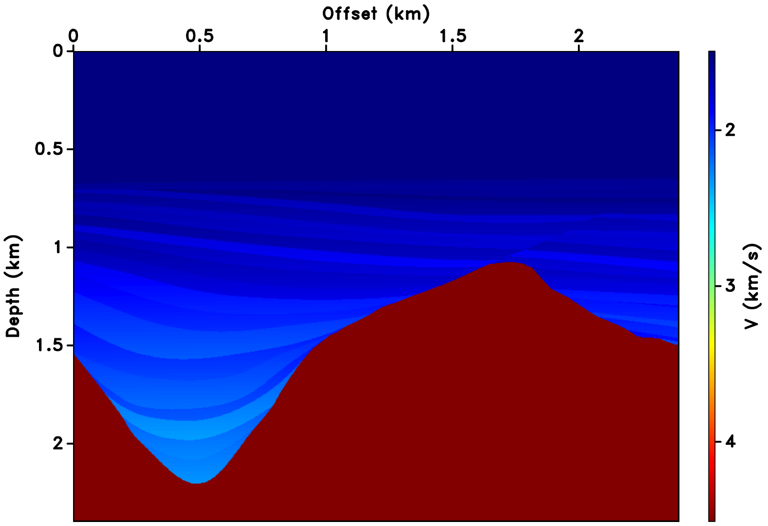
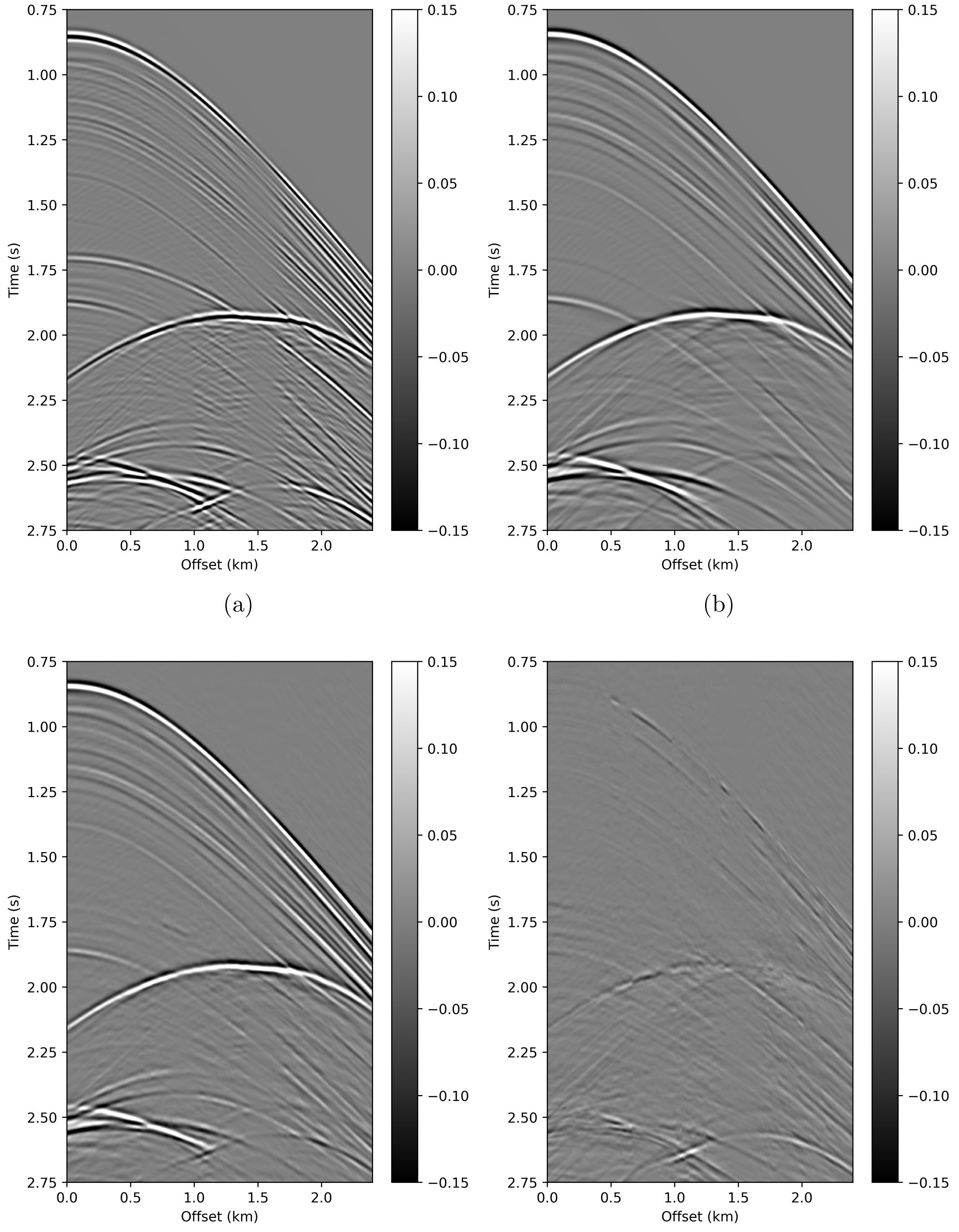


Figure 4.15 A sub-model of Sigsbee velocity model used for the prediction.

For the second example, Figure 4.15 shows another subset of the Sigsbee velocity model which consists of relatively flat subsurface structures with a fault near the far-offsets and a salt body located at the deeper parts of the model. Similar to the training models, 600 receivers are placed along 2.4 km horizontal extent with the receiver sampling of 4 m, and are located at 8 m depth. The coordinates of the source coordinates are $(x_s, z_s) = (25{,}7)$ m. The water column has 640 m depth with a velocity of 1500 m/s. The sea bottom for this example also has a reflection coefficient of approximately 0.33.

Figure 4.16(a) shows the data modeled with free-surface effects using the velocity model shown in Figure 4.15, and Figure 4.16(b) shows the data modeled without free-surface effects using the velocity model shown in Figure 4.15. Figure 4.16(a) shows the input data, and starting around 1.7 s in time at 0 km in offset, and ending around 2.3 s in time at 2.4 km in offset, the first multiple caused by the free surface appears. The challenge in this example is that the first event caused by the free surface overlaps with a primary event around 1.5 km in offset and 1.85 s in time in Figure 4.16(a). Conventional SRME techniques require an adaptive subtraction step in which the presence of overlapping primary and multiple events might result in removal or reduction of the primary energy. Although not as strong as the first event caused by the free surface, there are other multiples arriving after the first multiple which are caused by the free surface.
Figure 4.16(b) shows the desired output. The events described in Figure 4.16(a) caused by the free surface are not present in this figure because of the excluded free surface. For a successful CNN network training, we expect the CNN predictions to be similar to the ground truth (Figure 4.16(b)). After training the network, Figure 4.16(c) shows our CNN-based free-surface multiple elimination shot gather. Figure 4.16(d) shows the difference between the ground truth data (Figure 4.16(b)) and our CNN-based free-surface multiple elimination shot gather (Figure 4.16(c)).



(a) (b)

(c) (d)

Figure 4.16 (a) Numerically modeled shot gather with the free-surface multiples (input data) obtained using the velocity model given in Figure 4.15. (b) Numerically modeled shot gather without the free-surface multiples (output data/ground truth) obtained using the velocity model given in Figure 4.15. (c) Predicted shot gather without the free-surface multiples using CNNs. (d) Difference between the ground truth data (Figure 4.16(b)) and the CNN prediction (Figure 4.16(c))

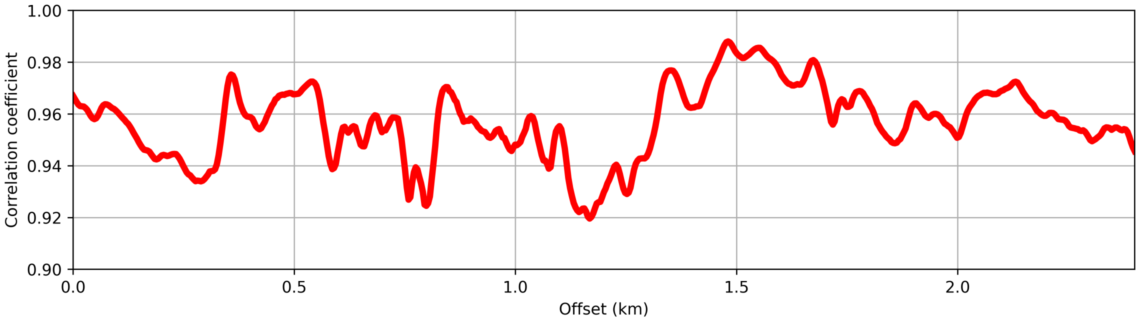
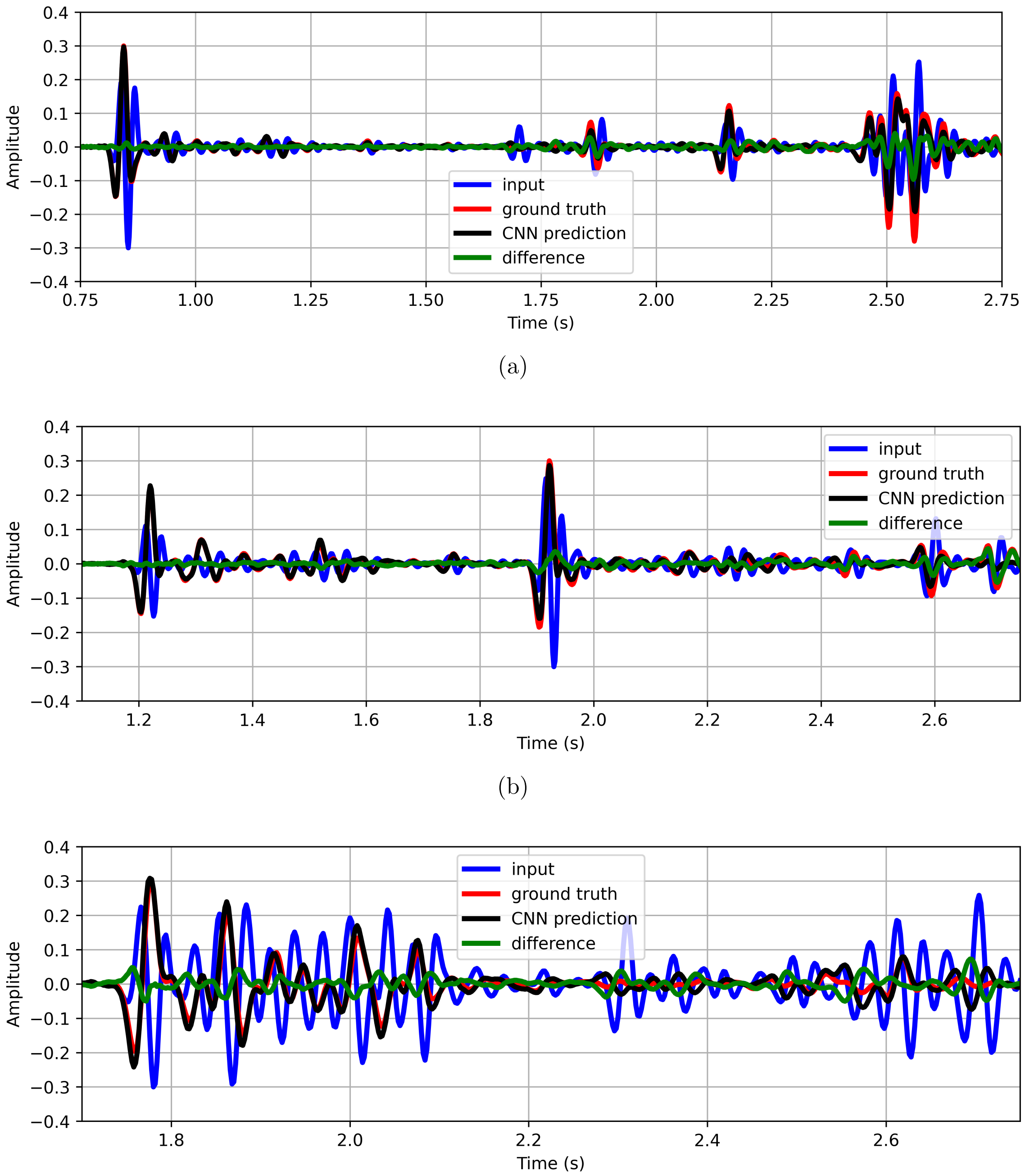


Figure 4.17 Trace-by-trace calculated correlation coefficient between the ground truth data (Figure 4.16(b)) and the CNN prediction (Figure 4.16(c)).

Figure 4.17 shows the CCs between the ground truth (Figure 4.16(b)) and the CNN prediction (Figure 4.16(c)), and the average CC is 0.95 which indicates a strong similarity between the ground truth data and the CNN prediction. Figure 4.18 shows several trace comparisons of the input data (solid blue line), output data (or ground truth) (solid red line), the CNN predictions (solid black line), and the difference between the ground truth and the CNN prediction (solid green line) shown in Figure 4.16 for the zero-, mid-, and far-offsets. Figure 4.18(a) shows the trace used as an input to the network which is the zero-offset trace extracted from Figure 4.16(a) (solid blue line), the ground truth that is the expected result from the CNN prediction which is the zero-offset trace extracted from Figure 4.16(b) (solid red line), the predicted zero-offset trace using our CNN-based free-surface multiple elimination operator extracted from Figure 4.16(c) (solid black line), and the difference zero-offset trace between the ground truth and our CNN-based free-surface multiple elimination operator extracted from Figure 4.16(d) (solid green line). In Figure 4.18(a), the strongest multiple caused by the free surface in the input trace (solid blue line) at time around 1.7 s has been successfully eliminated in the CNN prediction trace (solid black line) which matches the ground truth trace (solid red line). Figure 4.18(b) shows the mid-offset (offset 1.5 km) trace used as an input to the network from Figure 4.16(a) (solid blue line), the ground truth that is the expected result from the CNN prediction which is the mid-offset (offset 1.5 km) trace extracted from Figure 4.16(b) (solid red line), the predicted mid-offset (offset 1.5 km) trace using our CNN-based free-surface multiple elimination operator extracted from Figure 4.16(c) (solid black line), and the difference mid-offset trace between the ground truth and our CNN-based free-surface multiple elimination operator extracted from Figure 4.16(d) (solid green line). However, in Figure 4.18(b), the strongest multiple caused by the free surface in the input trace (solid blue line) at time around 1.9 s overlaps with a primary event. Although the free-surface reflection is embedded in the high-amplitude energy around 1.9 s in Figure 4.18(b), the CNN prediction is able to remove the free-surface multiple without removing the primary events. The CNN prediction trace (solid black line) closely matches the ground truth trace (solid red line) in Figure 4.18(b). Lastly, Figure 4.18(c) shows the far-offset (offset 2.3 km) trace used as an input to the network from Figure 4.16(a) (solid blue line), the ground truth that is the expected result from the CNN prediction which is the far-offset (offset 2.3 km) trace extracted from Figure 4.16(b) (solid red line), the predicted far-offset (offset 2.3 km) trace using our CNN-based free-surface multiple elimination operator extracted from Figure 4.16(c) (solid black line), and the difference far-offset trace between the ground truth and our CNN-based free-surface multiple elimination operator extracted from Figure 4.16(d) (solid green line). In Figure 4.18(c), the strongest multiple caused by the free surface in the input trace (solid blue line) at time around 2.35 s has been successfully eliminated in the CNN prediction trace (solid black line) which matches the ground truth trace (solid red line). The difference traces for the near-, mid-, and far-offset traces indicate a small amplitude mismatch between the ground truth and the CNN-based free-surface multiples. Figure 4.18 shows that the CNN-based free-surface multiple elimination is able to predict and remove the free-surface multiples and preserve the primary reflection even when the primary reflection and the free-surface multiple arrive simultaneously.



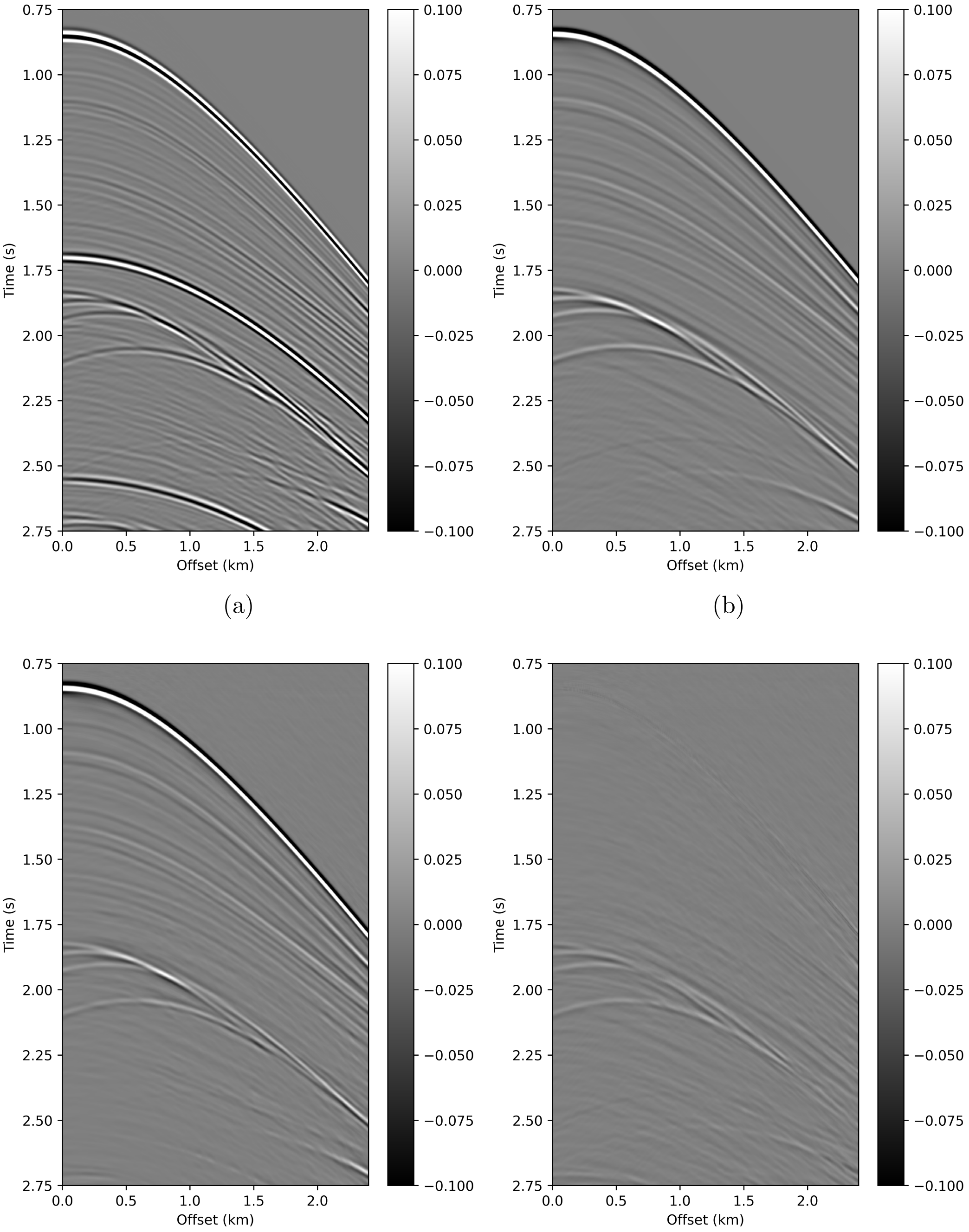
(a)

(b)

(c)

Figure 4.18 Trace comparison of the input (solid blue line), ground truth (solid red line), CNN prediction (solid black line), and the difference between the ground truth and the CNN prediction (solid green line) data from the zero-, mid-, and far-offsets, respectively, extracted from the data shown in Figure 4.16. (a) Comparison of the zero-offset traces. (b) Comparison of the mid-offset (offset 1.5 km) traces. (c) Comparison of the far-offset (offset 2.3 km) traces.

So far in this paper, we used different sub-models from Sigsbee velocity model for prediction with a reflection coefficient of around 0.33 at the sea-bottom. However, to test the accuracy of the CNN-based free-surface multiple elimination prediction, without changing the training data, we use a sub-model from Sigsbee velocity model for prediction with a reflection coefficient of approximately 0.5 at the sea-bottom (we create the reflection coefficient of approximately 0.5 at the sea-bottom by scaling the density model accordingly and adding the water layer). Figure 4.19(a) shows the data modeled with free-surface effects using the velocity model shown in Figure 4.11, but this time with a reflection coefficient of approximately 0.5 at the sea-bottom, and Figure 4.19(b) shows the waveforms modeled without free-surface effects using the velocity model shown in Figure 4.11 with a reflection coefficient of approximately 0.5 at the sea-bottom. Figure 4.19(a) shows the input data, and starting around 1.7 s in time at 0 km in offset, and ending around 2.3 s in time at 2.4 km in offset, the first multiple caused by the free surface (or the first-order free-surface multiple) is located. Additionally, starting around 2.55 s in time at 0 km in offset, and ending around 2.75 s in time at 1.5 in km offset, another multiple caused by the free surface is present. Note that in Figure 4.19(a) the multiples are of higher amplitude than in Figure 4.12(a) because of the higher reflection coefficient at the sea-bottom. Figure 4.19(b) shows the desired output. Figure 4.19(c) shows our CNN-based free-surface multiple elimination results. Figure 4.19(d) shows the difference between the ground truth data (Figure 4.19(b)) and our CNN-based free-surface multiple elimination results (Figure 4.19(c)). This example shows that our CNN-based method attenuates the free-surface multiples with different reflection coefficients at the sea-bottom.



(a)

(b)

(c)

(d)

Figure 4.19 (a) Numerically modeled shot gather with the free-surface multiples (input data) with a reflection coefficient of 0.5 at the sea-bottom obtained using the velocity model given in Figure 4.11. (b) Numerically modeled shot gather without the free-surface multiples (output data/ground truth) obtained using the velocity model given in Figure 4.11. (c) Predicted shot gather without the free-surface multiples using CNNs. (d) Difference between the ground truth data (Figure 4.19(b)) and the CNN prediction (Figure 4.19(c)).

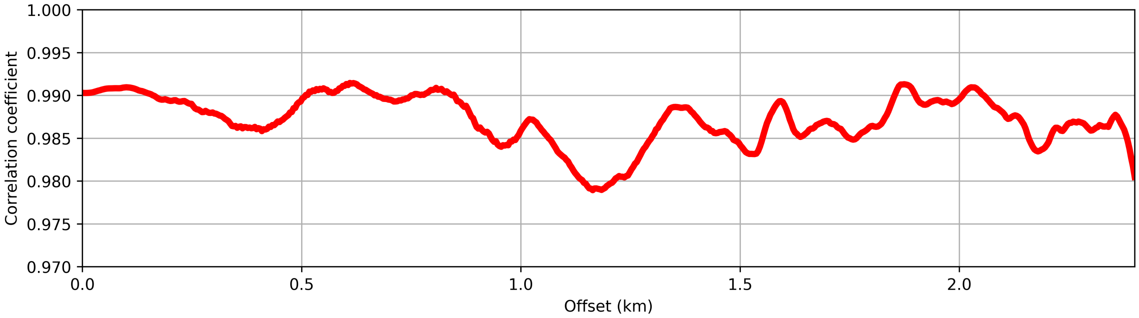
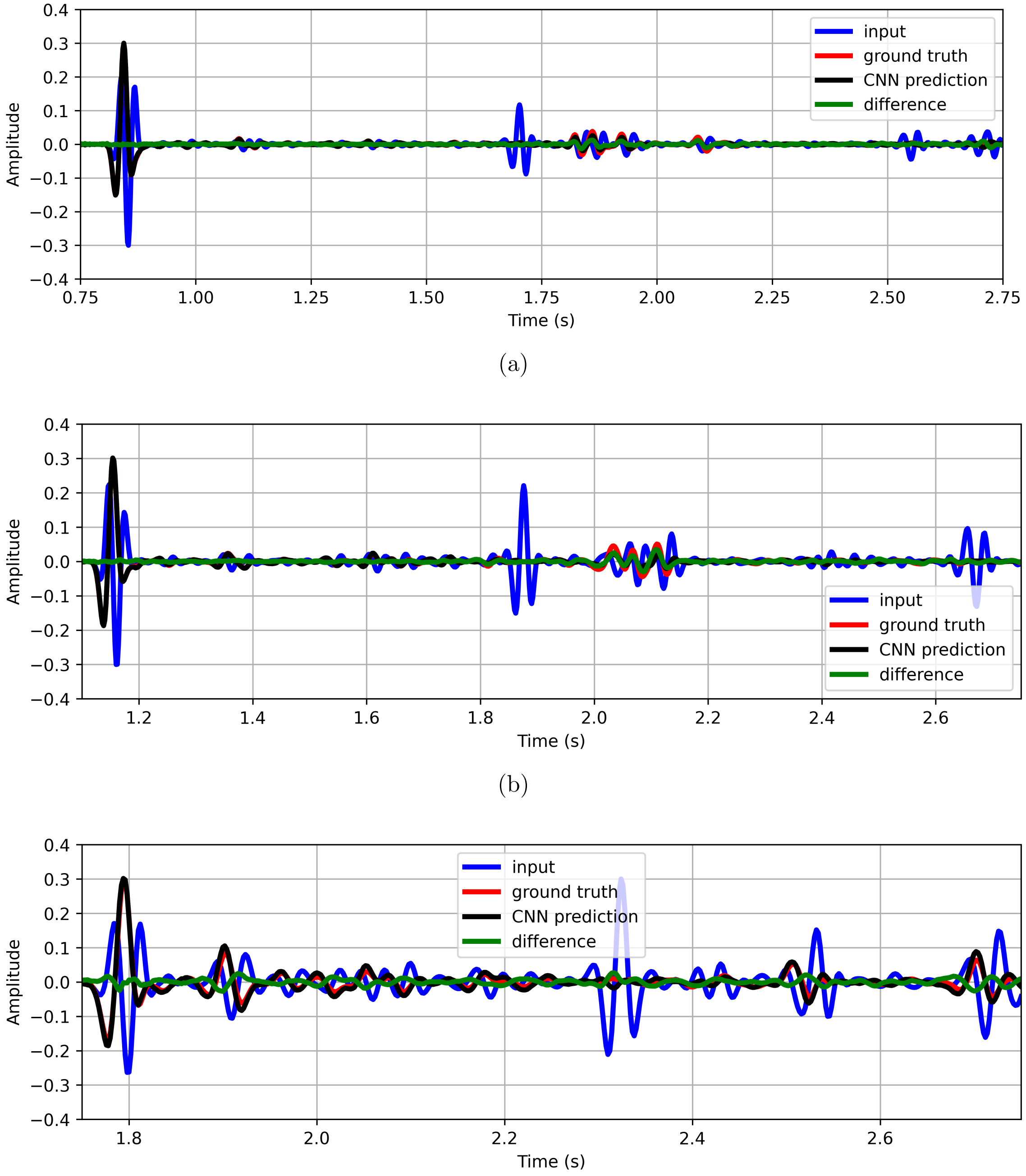


Figure 4.20 Trace-by-trace calculated correlation coefficient between the ground truth data (Figure 4.19(b)) and the CNN prediction (Figure 4.19(c)).

Figure 4.20 shows the CCs between the ground truth (Figure 4.19(b)) and the CNN prediction (Figure 4.19(c)), and the average CC is 0.98 which indicates a strong similarity between the ground truth data and the CNN prediction. Figure 4.21 shows the trace view comparison of the input, ground truth, and predicted data with a reflection coefficient of 0.5 at the sea bottom. Figure 4.21(a) shows a zero-offset trace comparison with the first-order free-surface multiple arriving around 1.7 s, and some other free-surface multiples arriving at 2.55 s and 2.7 s. Figure 4.21(b) shows a mid-offset (1.2 km) trace comparison with the first-order free-surface multiple arriving around 1.85 s, and some other free-surface multiples arriving at 2.4 s and 2.65 s. Figure 4.21(c) shows a far-offset (2.4 km) trace comparison with the first-order free-surface multiple arriving around 2.35 s. The difference traces for the near-, mid-, and far-offset traces indicate a small amplitude mismatch between the ground truth and the CNN-based free-surface multiples. Figure 4.21 shows that our CNN-based free-surface multiple elimination method successfully attenuates the surface-related multiples caused by a different reflection coefficient at the sea-bottom.



(a)

(b)

(c)

Figure 4.21 Trace comparison of the input (solid blue line), ground truth (solid red line), CNN prediction (solid black line), and the difference between the ground truth and the CNN prediction (solid green line) data from the zero-, mid-, and far-offsets, respectively, extracted from the data shown in Figure 4.19. (a) Comparison of the zero-offset traces. (b) Comparison of the mid-offset (offset 1.2 km) traces. (c) Comparison of the far-offset (offset 2.3 km) traces.

4.5 Discussion and Conclusions

The examples presented in this paper show that the CNN-based free-surface multiple elimination is able to remove the multiples caused by the free surface so that the CC between the numerically modeled data without the free-surface multiples and the CNN predictions is found to be around 0.97 on average. The change in the arrival time of the free-surface multiples with offset is correctly predicted by the CNN and the multiples are removed from the shot gathers without removing the primary events even when the primary and multiple reflections overlap. Although the data obtained by the CNN prediction mostly match the numerically modeled data, for late times of the shot gathers (i.e., for times around 2.75 s), there are mismatches between the CNN prediction and the ground truth. This is due to the length of the 1-D convolutional window (kernel size) running into the zero-padding towards the end of each trace to match the input and output data size in the network (see Figure 4.22). Kernel size (which changes between 51, 101, and 1125 samples in our case) determines the sliding window length to perform convolution along the sample, and to match the size of the input and output samples, we use zero-padding at the beginning and the end of each trace of a layer (see Figure 4.22). This causes both the early and late arrivals of the CNN predictions to have more mismatches than the rest of the data. Although the same happens for the early times of the shot gathers (i.e., for times around 0.75 s), there are no events recorded at such early times in the shot gathers. Although the CNN prediction and the desired output have a strong similarity (see Figure 4.13, Figure 4.17, and Figure 4.20), Figure 4.12(d), Figure 4.16(d), and Figure 4.19(d) show that there is an overall amplitude mismatch between the desired output and CNN prediction. The network is able to make predictions where the free-surface multiples are mostly eliminated, however for a better amplitude match between the desired output and CNN prediction, further hyperparameter tuning may be required (e.g., different combinations of the activation functions, number of hidden layers, number of filters, kernel sizes, strides, iterations (or epochs), choice of the loss function).

As we describe above, in the conventional SRME workflow, it is hard to accurately know if the predicted multiples have the precise amplitude information with the matching filtering, and/or if the adaptive subtraction step only removes the multiples without removing the primary events from the data. Additionally, the dense source/receiver distribution requirement makes SRME computationally expensive and hard to implement in case of sparse data acquisition (e.g., Ocean-Bottom Node (OBN) data acquisition). Since our CNN-based approach does not require dense source/receiver distribution, once the network is trained it can be applied to any type of acquisition where we have either 1 trace or 1,000,000 traces available. It also does not require an additional step to subtract the multiples from the data because the network predicts the data without free-surface multiples.

The CNN-based free-surface multiple elimination is able to remove the surface-related multiples caused by different sea-bottom reflection coefficients. To further test the accuracy of our CNN-based free-surface multiple elimination method, we can also test the conditions where we have variations in the source signature, source and receiver depth, water column depth, and application to the OBN and/or Carbon Capture, Utilization and Storage (CCUS) data. However, the variations of these parameters fall out of the scope of this paper and will be discussed in a future paper.

The use of 1-D convolutional filters makes our CNN-based free-surface multiple elimination computationally efficient (e.g., training takes 81 minutes and prediction of a trace takes 2.4 ms). Additionally, trace-by-trace elimination of free-surface multiples does not require a dense spatial distribution of sources and receivers. Our CNN-based method is suitable for irregular and sparse source/receiver acquisition geometry, and the accuracy of the prediction does not rely upon the availability of the near-offset and neighboring traces (which is one of the important requirements for the conventional SRME algorithms).

Our numerical examples show that free-surface multiples for subsurface models with variable velocity and density profiles with salt structures can be attenuated using CNNs once a network is trained. We show that when the primary and multiple reflections overlap, our CNN-based free-surface multiple elimination method successfully removes the multiples while preserving the primary reflections. Unlike the conventional SRME methods, our algorithm does not require modeling and subtraction of the multiples from the full dataset for free surface multiple elimination, and eliminates the risk of removing the primary energy from the recorded seismic data by providing a fast and effective alternative to the existing free surface multiple elimination methods. Our algorithm does not require dense spatial distribution of source and receivers, and it is also independent of the availability of the near-offset and neighboring traces.

4.6 Acknowledgments

We thank Khalid Almuteri from the Center for Wave Phenomena for his help with the CNN setup. This work is supported by the Consortium Project on Seismic Inverse Methods for Complex Structures at the Colorado School of Mines. The finite-difference forward modeling examples in this paper were generated using the Madagascar software package (Fomel *et al.*, 2013). Machine learning implementations were completed using the TensorFlow source library (https://www.tensorflow.org/).

4.7 Appendix A - Convolutional Neural Networks

Figure 4.22 shows a sketch of a two-layer 1-D convolutional neural network with a three-sample 1-D convolutional window (kernel). In Figure 4.22, there are 4 neurons, $a_1^{(1)}$, ..., $a_4^{(1)}$ in *layer* 1, and 4 neurons, $a_1^{(2)}$, ..., $a_4^{(2)}$ in *layer* 2 where the subscript denotes the neuron number, and the superscript denotes the layer number. Note that the term *neuron* is usually used in the context of a sample. For example, if we consider the neurons in *layer* 1 or *layer* 2 as a column vector, each neuron corresponds to an element in that column vector. There is also a three-sample temporal window in Figure 4.22 denoted with w_1, w_2, and w_3. The temporal window (or kernel) size is defined by the user in CNNs, and this kernel is initialized with random weights, which will later be updated during the training process. We slide this kernel from the beginning to the end of the sample in *layer* 1. As the kernel slides down, we calculate a centered dot product between *layer* 1 and assign the result of the dot product to the neurons in *layer* 2. We preserve the size of the neurons in each layer by zero padding at the beginning and the end of the neurons, and the zero padding is represented by the red neurons with zeros in Figure 4.22. Therefore, the result of the zero-padded centered convolution in *layer* 2 matches the shape of the input trace in *layer* 1. Note that if we do not use zero padding, the number of neurons in *layer* 2 does not match the number of neurons in *layer* 1.

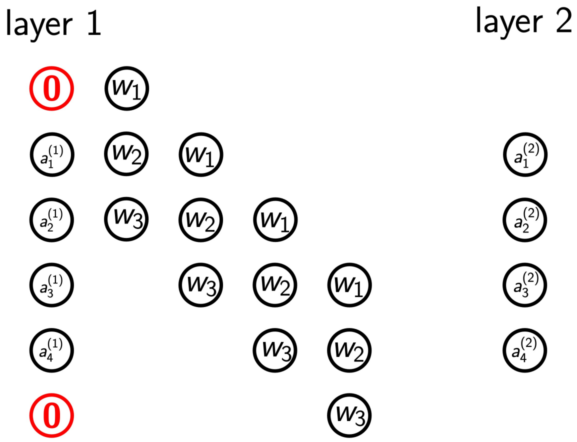


Figure 4.22 A sketch of a zero-padded 1-D convolutional neural network for a two-layer network where the number of output traces for a layer is three.

As the kernel slides down, we calculate a centered dot product between zero-padded *layer* 1 and the kernel, and assign the result of the dot product to the neurons in *layer* 2 as;

$$\begin{aligned} a_1^{(2)} &= \sigma\Big((a_1^{(1)}w_2) + (a_2^{(1)}w_3) + b_1\Big) \ , \\ a_2^{(2)} &= \sigma\Big((a_1^{(1)}w_1) + (a_2^{(1)}w_2) + (a_3^{(1)}w_3) + b_1\Big) \ , \\ a_3^{(2)} &= \sigma\Big((a_2^{(1)}w_1) + (a_3^{(1)}w_2) + (a_4^{(1)}w_3) + b_1\Big) \ , \\ a_4^{(2)} &= \sigma\Big((a_3^{(1)}w_1) + (a_4^{(1)}w_2) + b_1\Big) \ , \end{aligned} \tag{4.3}$$

where b_1 denotes the bias term and the subscript denotes the number of output traces for a layer, and σ denotes the activation function. The trainable parameters of the network shown in Figure 4.22 are w_1, w_2, w_3, and b_1. However, due to the zero padding around the beginning and the end of the sample, the prediction around these edges tends to be less accurate than the ones around the middle of the sample.

The number of output traces for a layer is another hyperparameter that needs to be determined. Figure 4.23 shows a cartoon of a 1-D convolutional neural network for a two-layer network where the number of output traces for a layer is three and kernel size is also three. Similar to the example with one filter (see Figure 4.22), as the three filters slide down, we calculate a centered dot product between *layer* 1 and each filter, and assign the result of the dot product to the neurons in *layer* 2. Figure 4.23(a) shows the first filter acting on *layer* 1, Figure 4.23(b) shows the second filter acting on *layer* 1, and Figure 4.23(c) shows the third filter acting on *layer* 1. Figure 4.23(d) shows the input layer, *layer* 1, and the outputs of three filters on *layer* 2. The trainable parameters of the network shown in Figure 4.23 are w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8, w_9, b_1, b_2, and b_3.

In summary, for each input trace to a layer and each output trace from a layer, an independent filter is applied, the number of such filters is the number of input traces times the number of output traces. For each output trace, these results are summed over the input traces, a bias term is added (one bias for each output trace), and then the activation function is applied as;

$$a_j^k = \sum_i^N \sigma(w_{ji} * a_i^{k-1} + b_j) \ , \tag{4.4}$$

where a_i^{k-1} is the ith output trace from layer $k-1$, a_j^k is the jth output trace from layer k, w_{ji} is the filter connecting input trace i to output trace j, and b_j is the bias for output trace j, $*$ denotes convolution in time, and N is the number of neurons in the previous layer.

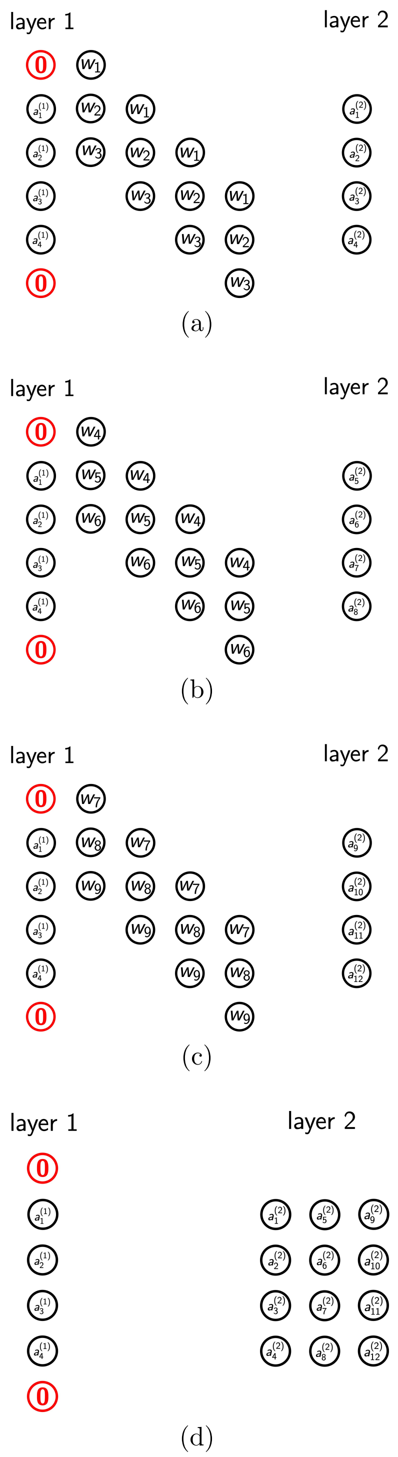


Figure 4.23 A sketch of a 1-D convolutional neural network for a two-layer network along with three three-sample temporal windows. (a) The first filter and its output on *layer 2*. (b) The second filter and its output on *layer 2*. (c) The third filter and its output on *layer 2*. (d) Output of three three-sample windows on *layer 2*.

Networks with multiple filters extract more information from the training input and output data pairs. However, increasing the number of filters also increases the computational demand of the network. Note that in Figure 4.22 and Figure 4.23 the kernels slides down one sample at a time and this parameter (which is one of the hyperparameters) is called *stride*. For a 1-D output in the next layer (*layer* 3), we define three different sets of weights which will be applied to the output on *layer* 2 shown in Figure 4.23(d), and the linear combination of these three different filtered outputs will form the output in *layer* 3 after adding the bias term and applying the activation function.

Although there is more hyperparameter selection and tuning involved during the training process, the illustration and description of each one of them are out of the scope of this paper. For more detailed information about DL methods and CNNs, we refer the reader to Ekman (2021), Géron (2019), Bishop (2006), and Higham & Higham (2019).

CHAPTER 5

REMOVING FREE-SURFACE EFFECTS FROM SEISMIC DATA USING CONVOLUTIONAL NEURAL NETWORKS – PART 2: AN APPLICATION TO THE MOBIL AVO VIKING GRABEN DATA SET

A paper submitted to *Geophysical Prospecting*

Mert Sinan Recep Kiraz [1,2,3], Roel Snieder[3] & Jon Sheiman

The presence of the air-water interface (or free surface) creates two major problems in marine seismic data; free-surface multiples and ghost reflections. The attenuation of free-surface multiples remains one of the most challenging noise attenuation problems in seismic data processing. Current solutions suffer from the removal of the primary event along with the multiple events especially when the primary and multiple events overlap (e.g., adaptive subtraction). The effective attenuation of ghost reflections (or *deghosting*) requires acquisition- and/or processing-related solutions which generally address the source-side and receiver-side ghosts separately. Additionally, an essential requirement for a successful implementation of free-surface multiple attenuation and dehosting is the requirement of dense seismic data acquisition parameters which is not realistic for two-dimensional (2-D) and/or three-dimensional (3-D) marine cases. We present a Convolutional Neural Network (CNN) approach for free-surface multiple attenuation and for seismic deghosting, simultaneously. Unlike the existing solutions, our approach operates on a single trace at a time, and neither relies on the dense acquisition parameters nor requires a subtraction process to eliminate free-surface multiples, and it removes both the source ghost and receiver ghost simultaneously. We illustrate the efficacy of the free-surface

[1] Primary researcher and author.

[2] Author for correspondence. Direct correspondence to mertkiraz@gmail.com.

[3] Center for Wave Phenomena, Colorado School of Mines, 1500 Illinois St., Golden, CO 80401, USA.

multiple attenuation and seismic deghosting technique using the Mobil AVO Viking Graben field data set. The application of our algorithm demonstrates that our CNN-based approach removes different orders of free-surface multiples (e.g., 1^{st} and 2^{nd} orders) and recovers the low-frequency content of the seismic data (which is essential for, for instance, full-waveform inversion applications and broadband processing) by successfully removing the ghost reflections while preserving and increasing the continuity of the primary reflections.

5.1 Introduction

The occurrence of multiples arises when waves interact with a strong impedance contrast. In marine seismic data, the free surface (or air-water interface) and water bottom are the two main impedance contrasts. Although the water bottom can be soft (e.g., weak impedance contrast) depending on the geology of the area, the free surface always is almost a perfect reflector (e.g., strong impedance contrast). This strong impedance contrast creates two main issues in the recorded seismic data; free-surface multiples and ghost reflections.

Free-surface multiples have at least one downward reflection at the free surface and are the most dominant set of multiples in marine seismic data. Free-surface multiples can have different numbers of orders which define the multiples' number of bounces from the free surface. With a higher number of orders, free-surface multiples become harder to attenuate later during the processing stage. Figure 5.1 shows an example of a 1st-order free-surface multiple (black line). The red star in Figure 5.1 denotes the source location at depth and the white triangle denotes the receiver location at depth. One of the widely used techniques to address free-surface multiples is called Surface-Related Multiple Elimination (SRME) (Verschuur *et al.*, 1992). This technique effectively predicts and iteratively subtracts the multiples (also known as adaptive subtraction) from the seismic data but does not always guarantee perfect solutions. During the subtraction step, the primary reflections might be removed from the seismic data along with the unwanted multiple reflections because of the overlapping primary and multiple events. Additionally, with a complex subsurface, SRME techniques fall short and require dense spatial distribution of source and receiver distribution (Dragoset *et al.*, 2010). This condition becomes even more difficult to fulfill in 3-D surveys.

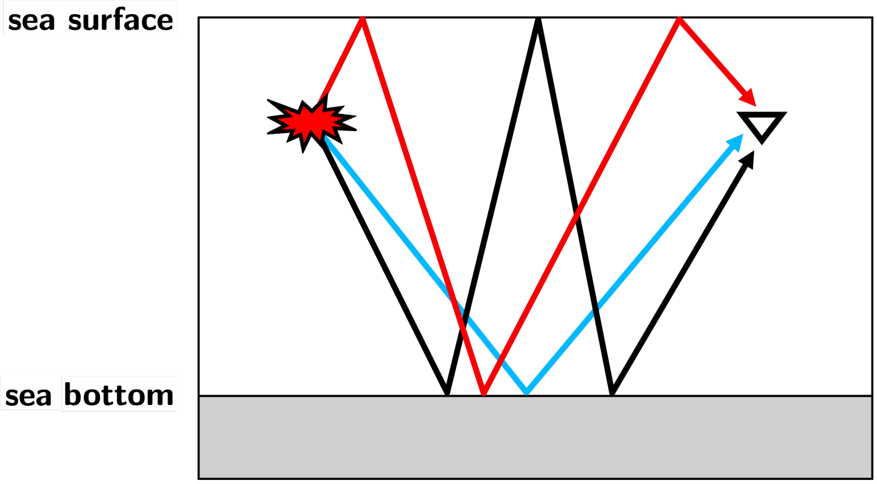


Figure 5.1 Schematic illustration of the ray paths of primary reflection (blue line), free-surface multiple (black line), and source- and receiver-side ghost reflections (red line). The white triangle denotes a receiver located at depth and the red shape denotes a source located at depth.

Ghost reflections create notches in the frequency content of the seismic data through destructive interference (Amundsen & Zhou, 2013). It is important to attenuate the effects of ghost reflections, especially for broadband data processing and full-waveform inversion applications (Kragh *et al.*, 2010; Virieux & Operto, 2009). The ghost reflections are recorded in the seismic data as the delayed version of the primary seismic signal with an opposite polarity (Dondurur, 2018). There are three types of ghost reflections that are present in the seismic data; source ghost, receiver ghost, and source and receiver ghosts. Figure 5.1 shows an example of the source and receiver ghost reflection (red line). Removing the ghost reflections from seismic data (or *deghosting*) is mainly performed in two different categories; acquisition-based solutions and processing-based solutions. Acquisition-based solutions include, but not limited to, slanted streamer acquisition

(Soubaras & Whiting, 2011), over/under streamer acquisition (Moldoveanu *et al.*, 2007), and dual-sensor streamer acquisition (Carlson *et al.*, 2007). Processing-based solutions include, but not limited to, deriving a low-frequency deghosting filter (Amundsen & Zhou, 2013), and joint deconvolution (Soubaras, 2010). Recently, Almuteri & Sava (2023) and Vrolijk & Blacquière (2021) have also utilized machine learning applications for removing ghost reflections from seismic data.

Seismic migration techniques assume that the input data only contain primary reflections and an example of a primary reflection is shown in Figure 5.1 (blue line). Therefore, it is imperative for seismic data to be free from the events caused by the presence of free surface (red and black lines in Figure 5.1). We propose a CNN-based solution that is designed to remove free-surface effects, namely, attenuating the free-surface multiples and seismic deghosting, simultaneously. We only use synthetic data set for training and make predictions using the field data set. We first introduce the CNN architecture we use in Section 5.2, then we illustrate the effectiveness of our CNN-based solution with a 2-D Mobil AVO Viking Graben field data set acquired in the North Viking Graben (Keys & Foster, 1998) in Section 5.3. The application of our CNN algorithm to field data demonstrates that we can attenuate free-surface multiples and remove ghost reflections simultaneously from the seismic data using the CNN. The CNN-based solution we propose requires neither dense source/receiver distribution nor is affected by the missing traces in the recorded data. We also show that during the simultaneous multiple attenuation and seismic deghosting using CNNs, primary reflections have been preserved and their continuity has been increased.

5.2 CNN Architecture and Training

Recently, different machine learning (ML) and/or convolutional neural network (CNN) methods have been proposed to attenuate free-surface multiples or remove ghost reflections from seismic data (Almuteri & Sava, 2023; Hu *et al.*, 2019; Ovcharenko *et al.*, 2021; Siahkoohi *et al.*, 2018, 2019; Tao *et al.*, 2021; Vrolijk & Blacquière, 2021; Zhang *et al.*, 2021). In this paper, we propose a CNN that predicts a single trace at a time, making our algorithm insensitive to the data gaps in the acquisition and does not require a meticulous data-filling pre-processing step, unlike the current techniques.

CNN is a subset of ML and is based on the mathematical and statistical methods that let machines and computers improve specified tasks with experience (Bishop, 2006; Ekman, 2021; Géron, 2019). CNNs consist of layers of filters and as the number of layers increases, the network extracts more complex information, and the networks with many layers are called "deep neural networks". Every sample in a layer is called a neuron and the output value of a neuron is obtained by

$$a_j^k = \sigma\Big(\sum_{i=1}^{N}(a_i^{(k-1)} w_{ij}^k) + b_j^k\Big) \ , \tag{5.1}$$

where N denotes the number of neurons in the previous layer, a_j^k denotes the value a to be calculated of the jth neuron in the kth layer, $a_i^{(k-1)}$ denotes the value a of the ith neuron in the $(k-1)$th layer, w_{ij}^k denotes the weight w of the ith neuron which is connected to the jth neuron in the kth layer, b_j^k denotes the bias b of the jth neuron in the kth layer, and σ denotes the activation function. The activation function is a nonlinear function so that the action of the neural network is nonlinear as well.

The process in which we adjust the weights, w, and biases, b, by an iterative data-fitting process is called training. During the training, once an output of a network is obtained, we define an error function to compare the output of the network to the desired output. The most commonly used error function is the Mean-Squared Error (MSE) metric that measures the difference between the predicted and the ground truth results,

$$MSE = \frac{1}{m}\sum_{i=1}^{m}(\hat{y}_i - y_i)^2 \ , \tag{5.2}$$

where $\hat{y}_i$ is the output of the network, y_i is the ground truth data for a given input data during the training, and m is the number of training examples. We minimize the MSE function by using the gradient descent method by calculating the gradient of the error function with respect to the weights. Training a CNN is commonly done by backpropagation where the choice loss function and gradient descent optimization algorithm are essential (Bishop, 2006; Ekman, 2021; Géron, 2019).

In this study, we use the network architecture proposed by Kiraz *et al.* (2023a) for simultaneous free-surface multiple attenuation and deghosting. Our CNN architecture consists of an input layer, nine one-dimensional (1-D) convolutional layers, and an output layer. The first and second, fourth and fifth, and seventh and eighth layers have 32 filters (Kiraz *et al.*, 2023a); the third, sixth, and ninth layers have eight filters. The length of the 1-D convolutional window for the first three layers is 1110 samples; for the following five layers is 101 samples; and for the last layer is 51 samples. Overall, the network consists of 1,754,017 total parameters all of which are trainable. For the input layer and the convolutional layers, we use the Rectified Linear Unit (ReLU) activation function, σ, (Ekman, 2021) which computes the function $\sigma(x) = max(0, x)$. We use a stochastic gradient descent optimization algorithm with momentum along with the mean-squared loss function as the objective function Géron (2019). To prevent the network from over-fitting, we split the input data into training data set, testing data set, and validation data set using 60%, 20%, and 20% ratios, respectively.

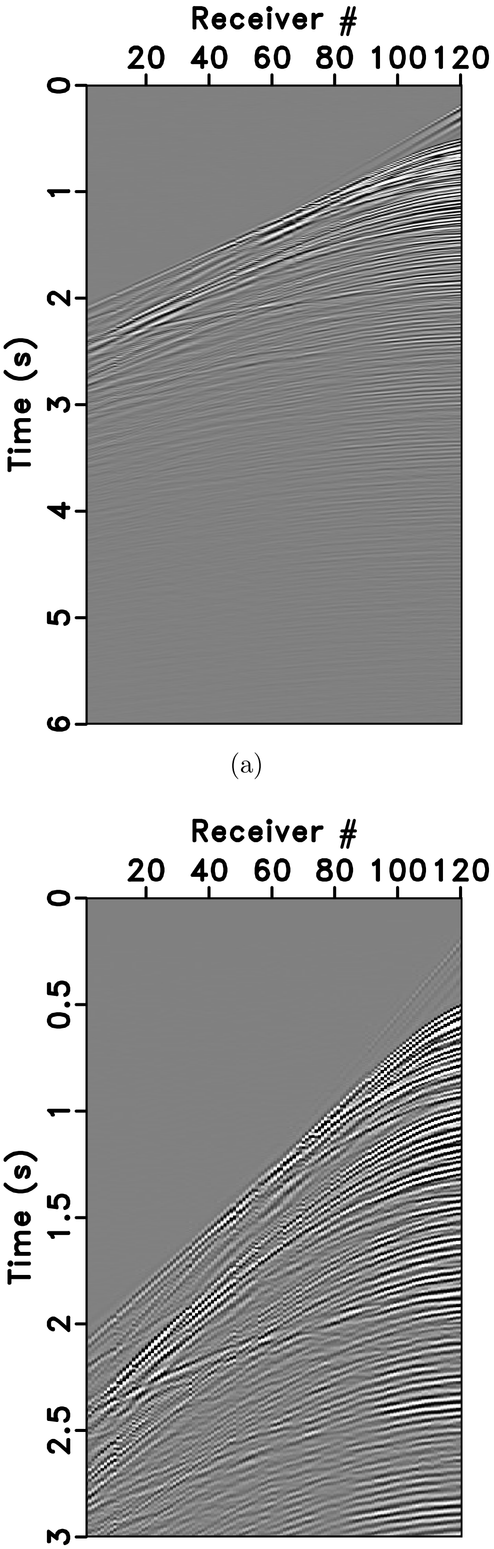
For the training model selection, we use the Marmousi (Versteeg, 1994) and Pluto (Stoughton *et al.*, 2001) velocity models. For the training source and receiver geometry, we use the source and receiver geometry of the field data set. We extract sub-models from the Marmousi and Pluto models and add a water layer to the top of each model using the water column depth of the field data set. For each velocity model, we use a density model that is a scaled version of the velocity model by a factor of two (and the units are converted from km/s to g/cm^3) and has the same water layer as the velocity models. Scaling the velocity model by two and adding the water layer into the density model creates a range of reflection coefficients for training purposes that helps generalize the network applicability. We use 52 input shot gathers with free-surface multiples and ghost reflections each of which consists of 100 traces (total of 5,200 input traces) using 2-D finite-difference forward modeling, and 52 output shot gathers without free-surface multiples and ghost reflections each of which consists of 100 traces (total of 5,200 output traces) using 2-D finite-difference forward modeling.

5.3 Field Data Experiment: The Mobil AVO Viking Graben Data Set From The North Viking Graben

The Mobil AVO Viking Graben data set was acquired in the North Sea over the Viking Graben (Keys & Foster, 1998). The initial purpose of this data set was to test different AVO methods using a single data set; however, when the results were presented at a workshop at the 64th SEG International Exposition and Annual Meeting in 1994, this data set was also used to cover problems related to geology, inversion, data processing, wave propagation, rock properties, and AVO (Keys & Foster, 1998; Lumley *et al.*, 1998).

The seismic line consists of 1,001 shot gathers each of which has a 6 s recording length with a 4 ms sampling rate. For each shot gather, 120 receivers are used, and the near-offset receiver is located at 262 m and the far-offset receiver is located at 3,237 m. Both the shot and receiver group interval is 25 m. The receivers are placed at a 10 m depth, and the air-gun array is placed at a 6 m depth (Keys & Foster, 1998).

The study area has a water depth of approximately 350 m. Strong water-bottom and interbed multiples contaminate the data and are a major source of noise (Keys & Foster, 1998; Madiba & McMechan, 2003). We use a CNN-based algorithm to attenuate free-surface multiples and ghost reflections, simultaneously, from this data set.



(a)

(b)

Figure 5.2 (a) Raw shot gather. (b) Shot gather in (a) after applying the pre-processing steps.

As part of the pre-processing of the seismic data for the CNN application, we first account for the geometric spreading in both the field data set and the synthetic data set used for training. The 3-D effects of geometric spreading (Liner, 2016) can be accounted for by

$$A_{3D} \propto \frac{1}{t} \ , \tag{5.3}$$

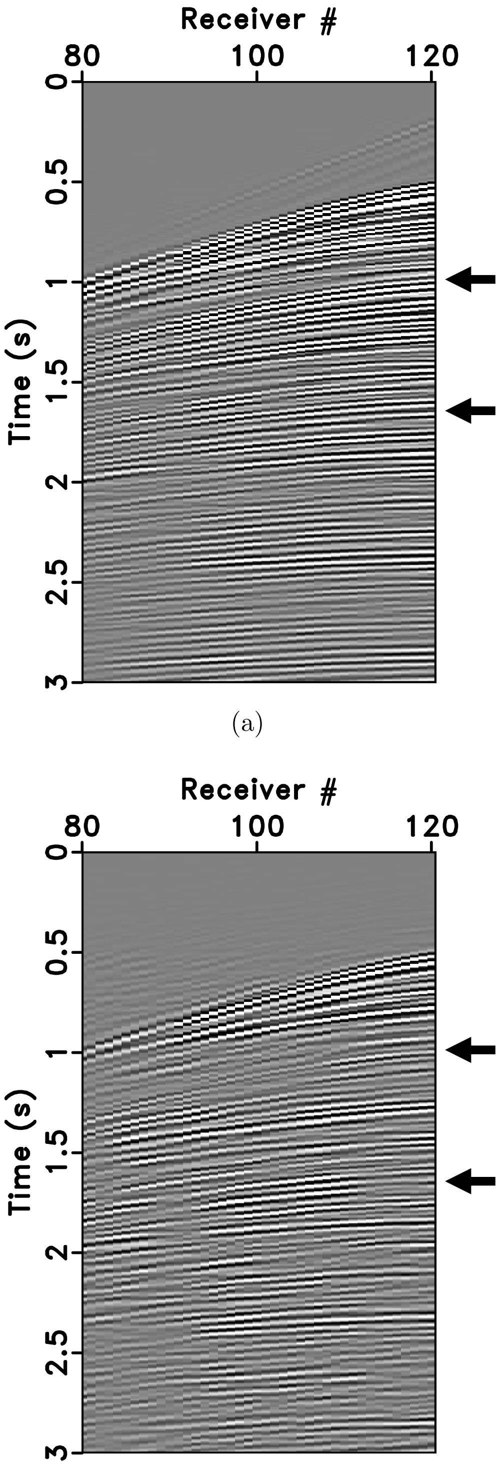
and similarly, the 2-D effects of geometric spreading (Liner, 2016) can be accounted for by

$$A_{2D} \propto \frac{1}{\sqrt{t}} \ , \tag{5.4}$$

where A denotes the amplitude and t denotes time. Therefore, we apply a t gain and $\sqrt{t}$ gain to the field data set and the synthetic data set, respectively. Because we use 1-D convolutional layers, we mimic the amplitudes in 1-D by applying the geometric spreading corrections given in equations (5.3) and (5.4). Next, we apply a band-pass filter with 15 Hz and 50 Hz cuts to the field data set to match the frequency content of the training data set and the field data set. We also normalize the data prior to using for the CNN application (Ekman, 2021; Géron, 2019). As a last step of the pre-processing, we only use the first 3.0 s of the seismic data for computational efficiency purposes.

Figure 5.2(a) shows the raw shot gather number 88 from the field data set and following the pre-processing steps described above, Figure 5.2(b) shows the shot gather shown in Figure 5.2(a) after applying t gain, band-pass filter, and keeping the data recorded from 0 s to 3.0 s only.

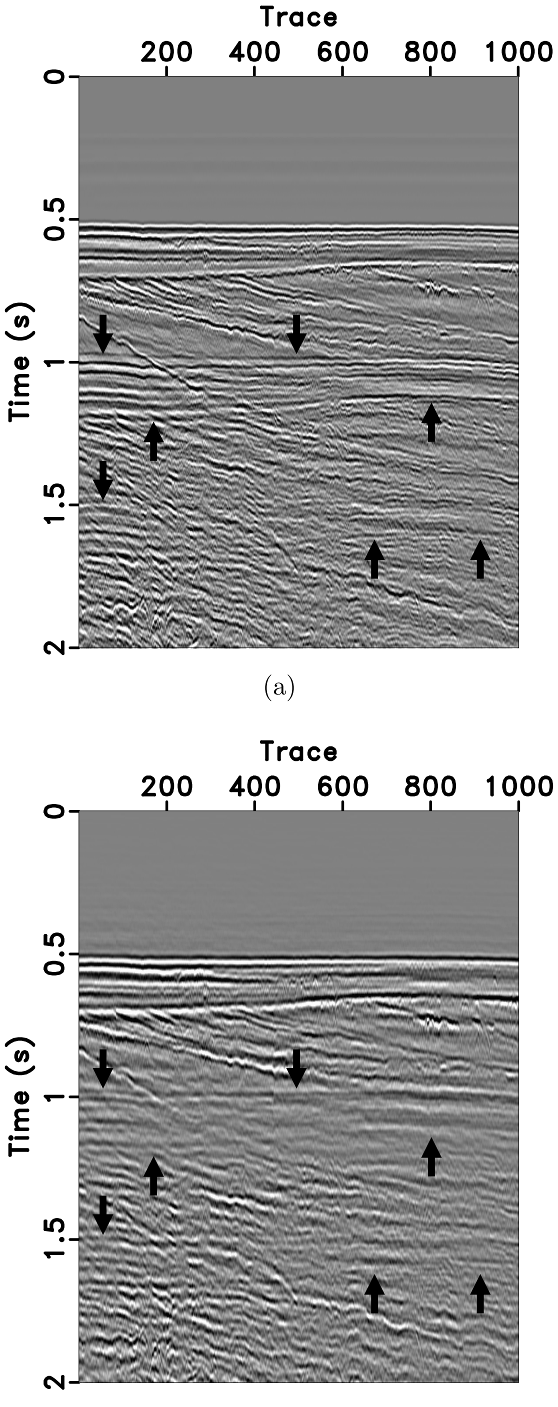
After applying the pre-processing steps to the entire field data set, we apply the CNN-based free-surface multiple attenuation and seismic deghosting. Figure 5.3(a) shows the shot gather number 88 before free-surface multiple attenuation and seismic deghosting. Figure 5.3(b) shows the shot gather number 88 after free-surface multiple attenuation and seismic deghosting using a CNN. The arrows in Figure 5.3(a) indicate some of the 1st- and 2nd-order free-surface multiples and the arrows in Figure 5.3(b) point to areas in the data where these multiples have been attenuated.



(a)

(b)

Figure 5.3 (a) Shot gather *before* free-surface multiple attenuation and deghosting. (b) Shot gather *after* free-surface multiple attenuation and deghosting using the CNN-based algorithm.



(a)

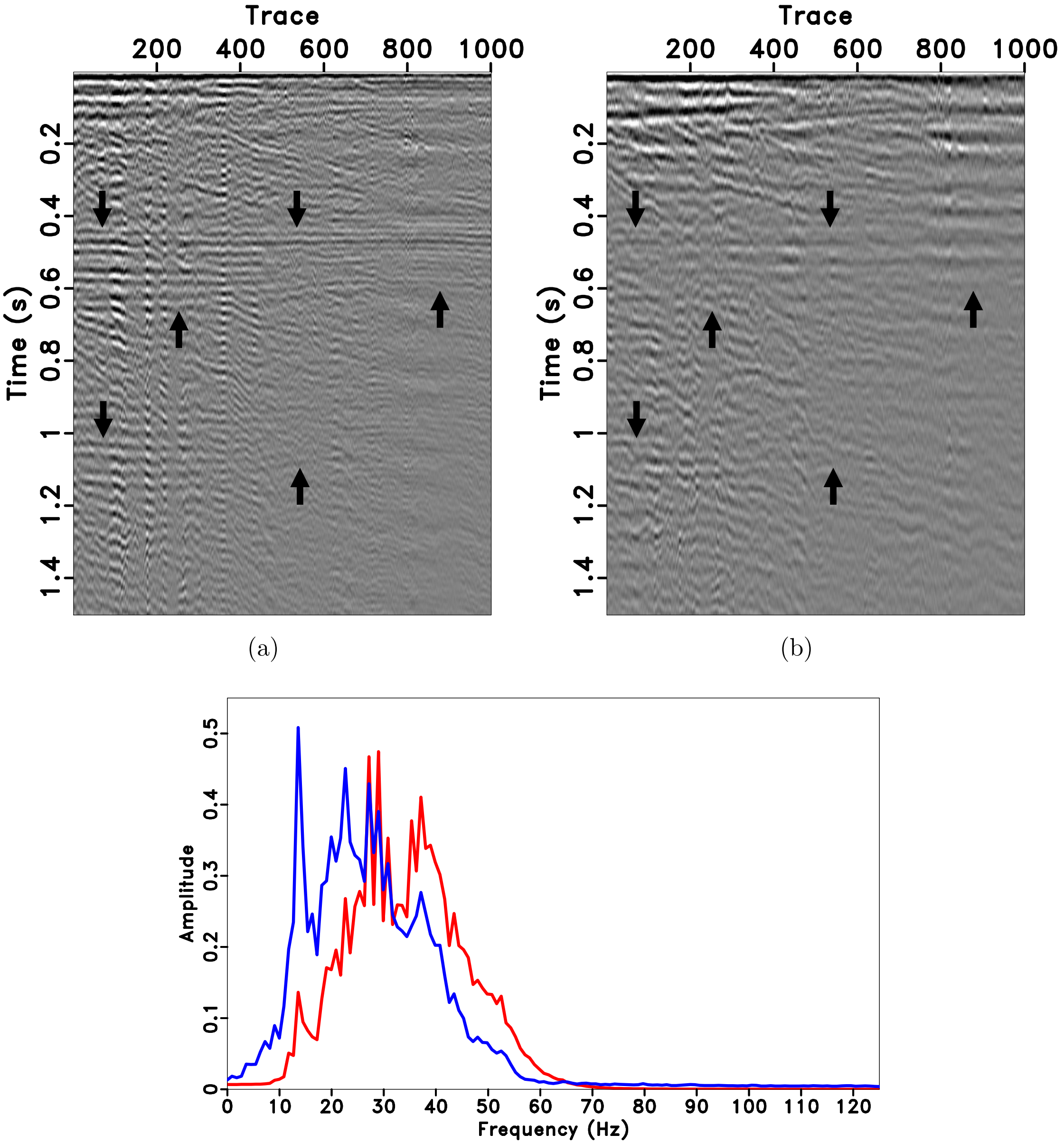
(b)

Figure 5.4 (a) Near-trace gather of the seismic line *before* free-surface multiple attenuation and deghosting shown only 0-2 s. (b) Near-trace gather of the seismic line *after* free-surface multiple attenuation and deghosting using a CNN shown only 0-2 s.

Figure 5.4(a) shows the near-offset trace of each shot gather (or the near-trace gather) of the entire data set before free-surface multiple attenuation and seismic deghosting, and Figure 5.4(b) shows the near-trace gather of the entire data set after free-surface multiple attenuation and seismic deghosting. The arrows between 1 s and 1.2 s on the left and right sides in Figure 5.4(a) point to the 1st-order free-surface multiples, and the three arrows between 1.5 s and 1.7 s on the left side in Figure 5.4(a) point to the 2nd-order free-surface multiples. Figure 5.4(b) shows that these multiples have been attenuated by our CNN-based algorithm.

To better evaluate the CNN prediction, we use auto-correlation plots as a diagnostic for periodicity in the records that is associated with multiples (Yilmaz, 2001). Figure 5.5(a) shows the auto-correlation of the near-trace gather shown in Figure 5.4(a) before free-surface multiple attenuation and seismic deghosting. Figure 5.5(b) shows the auto-correlation of the near-trace gather shown in Figure 5.4(b) after free-surface multiple attenuation and seismic deghosting. The arrows around 0.5 s and 1 s in Figure 5.5(a) indicate the periodicity of the 1st- and 2nd-order free-surface multiples, respectively. Figure 5.5(b) shows that both 1st- and 2nd-order free-surface multiples have been attenuated and their periodicity is not evident in the auto-correlation of the data after the application of our CNN-based algorithm.

Next, we evaluate the CNN-prediction results for seismic deghosting by comparing the average amplitude spectra before and after CNN-based free-surface multiple attenuation and seismic deghosting using the near-trace gathers shown in Figure 5.4. Figure 5.5(c) shows the average amplitude spectrum before free-surface multiple attenuation and seismic deghosting (red line) using the near-trace gather shown in Figure 5.4(a), and the average amplitude spectrum after free-surface multiple attenuation and seismic deghosting (blue line) using the near-trace gather shown in Figure 5.4(b). The average amplitude spectra before and after deghosting in Figure 5.5(c) show that the low frequencies have been recovered as a result of the CNN-based seismic deghosting.

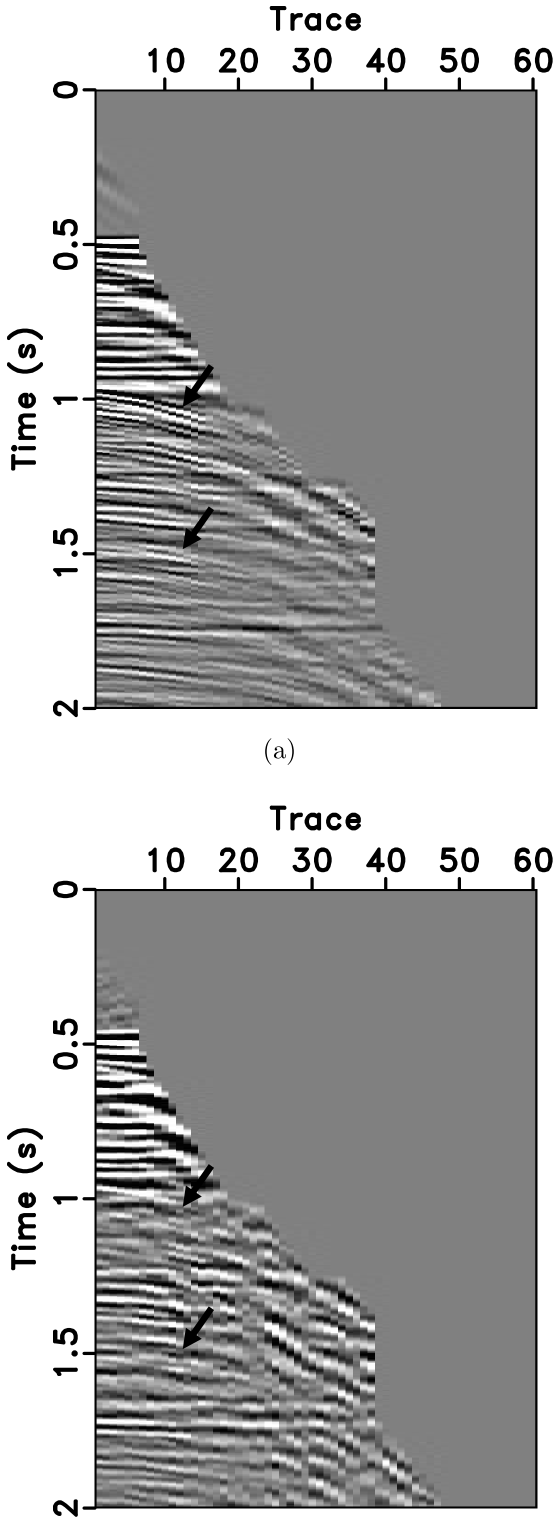


(a)

(b)

(c)

Figure 5.5 (a) Auto-correlation of the near-trace gather *before* free-surface multiple attenuation and deghosting shown in Figure 5.4(a). (b) Auto-correlation of the near-trace gather *after* free-surface multiple attenuation and deghosting shown in Figure 5.4(b). (c) Amplitude spectra of the near-trace gather *before* (red line) and *after* (blue line) free-surface multiple attenuation and deghosting.



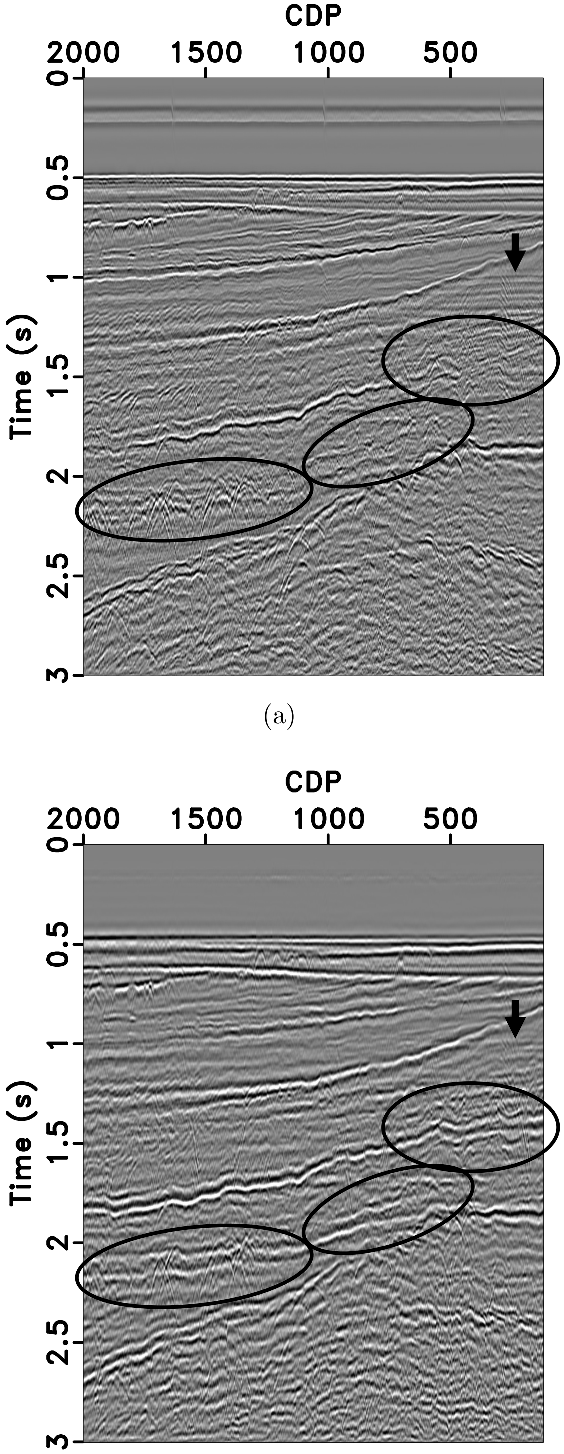
(a)

(b)

Figure 5.6 (a) NMO-corrected CDP gather *before* free-surface multiple attenuation and deghosting after muting the stretch zone. (b) NMO-corrected CDP gather *after* free-surface multiple attenuation and deghosting after muting the stretch zone.

Next, we use Common Depth Point (CDP) gathers to evaluate the effectiveness of our CNN-based free-surface multiple attenuation. We first apply a Normal-Moveout Correction (NMO) to the CDP gathers using a 1-D velocity profile (Keys & Foster, 1998). Because multiples have a slower velocity than the actual medium velocity, they appear on the NMO-corrected gathers as not flattened completely (or under-corrected), and the NMO-corrected CDP gathers help us distinguish between the primary and multiple reflections (Yilmaz, 2001). Figure 5.6(a) shows NMO-corrected 700th CDP before free-surface multiple attenuation and deghosting, and Figure 5.6(b) shows NMO-corrected 700th CDP after free-surface multiple attenuation and deghosting using the CNN. The arrows in Figure 5.6(a) indicate some of the under-corrected multiples, and Figure 5.6(b) shows that those multiples have been attenuated. We also mute the NMO-corrected gathers in Figure 5.6 for the stretch zones at large offsets.

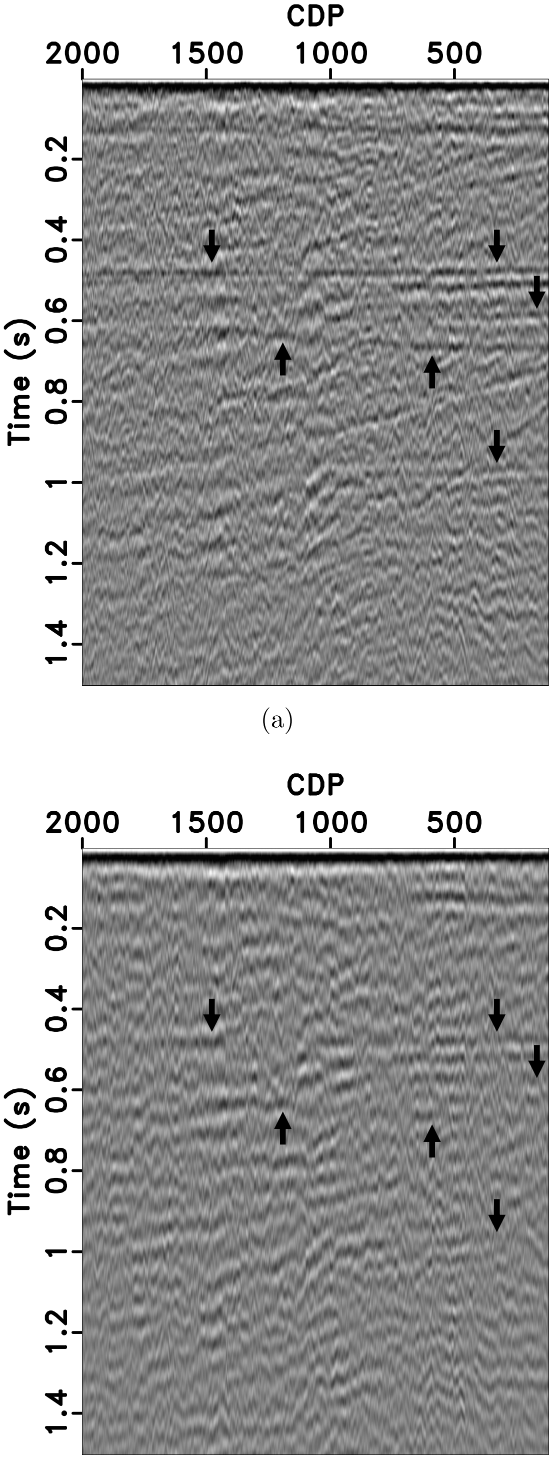
We sum the traces in the NMO-corrected CDP gathers, which is referred to as CDP stacking, to obtain a stack trace at the particular CDP location. The CDP stacking attenuates multiples to some extent as a result of residual moveout after applying NMO correction (Yilmaz, 2001). The CDP stack sections of the entire seismic data set after applying Automatic Gain Control (AGC) with 500 ms time gate (Yilmaz, 2001) for display purposes before and after the attenuation of surface multiples and deghosting with the CNN are shown in Figure 5.7(a) and Figure 5.7(b), respectively. The arrow in Figure 5.7(a) indicates free-surface multiples after CDP stacking using the data before free-surface multiple attenuation, and as shown in Figure 5.7(b), these multiples are attenuated by the application of the CNN. The events in Figure 5.7(a) are more continuous than the reflectors in Figure 5.7(b). Comparing the areas pointed out by the ellipses in Figure 5.7, the free-surface multiples that mask the primary reflection have been removed and the continuity of the primary has been recovered after attenuating the free-surface multiples and removing the ghost reflections using the CNN.



(a)

(b)

Figure 5.7 (a) CDP stack *before* free-surface multiple attenuation and deghosting after applying AGC. (b) CDP stack *after* free-surface multiple attenuation and deghosting after applying AGC.



(a)

(b)

Figure 5.8 (a) Auto-correlation of the near-trace gather *before* free-surface multiple attenuation and deghosting shown in Figure 5.7(a). (b) Auto-correlation of the near-trace gather *after* free-surface multiple attenuation and deghosting shown in Figure 5.7(b).

Lastly, Figure 5.8 shows the auto-correlation of the CDP stack section before and after free-surface multiple attenuation by the CNN in Figure 5.8(a) and Figure 5.8(b), respectively. The arrows in Figure 5.8(a) point out the remaining free-surface multiples after CDP stacking, and as shown in Figure 5.8(b), these remaining multiples have been attenuated by the application of our CNN-based algorithm.

5.4 Conclusions

We apply a CNN-based free-surface multiple attenuation and seismic deghosting to the Mobil AVO Viking Graben data set. Our CNN-based solution applies a trace-by-trace attenuation of free-surface multiples and ghost removal, and does not rely on a dense spatial distribution of sources and receivers. Therefore, our CNN-based method is suitable for acquisition geometries with irregular and sparse sources and receivers, and the accuracy of the prediction does not rely upon the availability of the near-offset and neighboring traces (which is one of the important requirements for the conventional SRME and deghosting algorithms). We show that while our CNN-based method removes free-surface effects (i.e., free-surface multiples and ghost reflections), it also preserves and increases the continuity of the primary reflections in the seismic data. We show that the 1st- and 2nd-order free-surface multiples have been successfully attenuated and the low-frequency content of the seismic data has been successfully recovered.

5.5 Acknowledgments

This work is supported by the Consortium Project on Seismic Inverse Methods for Complex Structures at the Colorado School of Mines. The finite-difference forward modeling examples in this paper were generated using the Madagascar software package (Fomel *et al.*, 2013). Machine learning implementations were completed using the TensorFlow source library (https://www.tensorflow.org/).

5.6 Data and Materials Availability

Data associated with this research are publicly available and can be obtained by visiting https://wiki.seg.org/wiki/Mobil_AVO_viking_graben_line_12.

CHAPTER 6

FREE-SURFACE MULTIPLE ATTENUATION AND SEISMIC DEGHOSTING FOR BLENDED DATA USING CONVOLUTIONAL NEURAL NETWORKS

A paper to be submitted to *Geophysics*

Mert Sinan Recep Kiraz [1,2,3], Roel Snieder[3] & Jon Sheiman

Simultaneous source acquisition has become common over the last decade for marine seismic surveys because of the increased efficiency of seismic acquisition by limiting the time, reducing the cost, and having less environmental impact than conventional single-source (or unblended) acquisition surveys. For simultaneous source acquisition, seismic sources at different locations are fired with time delays, and the recorded data are referred to as *blended data*. There are two different ways to process the blended data: (1) applying conventional processing algorithms after separating the blended data into individual data sets (also known as deblending), or (2) developing new algorithms for the blended data. The air-water interface (or free-surface) creates strong multiples and ghost reflections for blended seismic data. The multiples and/or ghost reflections caused by a source in blended data overlap with the primary reflections of another source, thus, creating a strong interference between the primary and multiple events of different sources. We develop a Convolutional Neural Network (CNN) method to attenuate free-surface multiples and remove ghost reflections simultaneously from the blended seismic data. The CNN-based solution we propose operates on single traces and is not sensitive to the missing near-offset traces, missing traces, and irregular/sparse acquisition parameters (e.g., for ocean-bottom node acquisition and time-lapse monitoring studies). We illustrate the efficacy of our free-surface multiple attenuation and seismic deghosting method by

[1] Primary researcher and author.

[2] Author for correspondence. Direct correspondence to mertkiraz@gmail.com.

[3] Center for Wave Phenomena, Colorado School of Mines, 1500 Illinois St., Golden, CO 80401, USA.

presenting synthetic and field data applications. The numerical experiments demonstrate that our CNN-based approach for simultaneously attenuating free-surface multiples and removing ghost reflections can be applied to the blended data without the deblending step. Although the interference of primaries and multiples from different shots in the blended data makes free-surface multiple attenuation harder than the unblended data, we demonstrate that our CNN-based method effectively attenuates free-surface multiples in the blended synthetic and field data even when the delay time for the blending is different in the training data than in the data to which the CNN is applied.

6.1 Introduction

Conventional single-source seismic acquisitions require shots to be recorded separately, and the time interval between shots is generally set to be large enough to avoid possible interference between two consecutive shots. Simultaneous source acquisition (also known as blended acquisition), on the other hand, allows two or more shots to be fired and recorded with a dithering delay allowing overlap between the reflected waves for different shots (Beasley *et al.*, 1998, 2012; Berkhout, 2008; Hampson *et al.*, 2008). Simultaneous source acquisition reduces the environmental impact of seismic data acquisition, the required time, and the cost of seismic surveys and creates better subsurface illumination with denser source distribution over the entire survey compared to conventional single-source seismic acquisitions. The recorded data using the simultaneous source acquisition are referred to as blended data. To handle the blended data, one can either separate blended data into individual data sets (also known as deblending) or process the blended data (Berkhout, 2008).

Blended data have been studied for various applications such as migration of blended data (Verschuur & Berkhout, 2011) and primary estimation using blended data (Groenestijn & Verschuur, 2011). Additionally, Ma *et al.* (2016) propose a surface-related multiple elimination based method and an inverse data processing based method for free-surface multiple attenution for blended data.

Regardless of the seismic acquisition type, the presence of the air-water interface (or free-surface) in marine seismic data causes free-surface multiples and ghost reflections, both of which need to be removed from seismic data to obtain an accurate image of the subsurface. Ghost reflections create notches in the frequency spectrum of seismic data which deteriorates the quality of the low-frequency component of seismic data (Amundsen & Zhou, 2013). There are different ghost reflections in seismic data namely source-side ghost, receiver-side ghost, and source and receiver-side ghosts. The source-side ghost propagates from a source as an up-going wave, the receiver-side ghost propagates to a receiver as a down-going wave, and the source and receiver-side ghosts is a combination of a source-side and receiver-side ghosts (Almuteri & Sava, 2023). Free-surface multiples have at least one downward reflection at the free surface and are one of the most challenging types of multiples to remove from seismic data. Surface-Related Multiple Elimination (SRME) (Dragoset *et al.*, 2010; Verschuur *et al.*, 1992) is a method that deals with free-surface multiples with at least one downward reflection at the free surface. The SRME method predicts the amplitude and arrival time of free-surface multiples, and has been the industry standard for attenuating free-surface multiples (Ma *et al.*, 2019). However, SRME depends on an adaptive subtraction step in which the amplitude discrepancy between the predicted and actual multiples is minimized. Adaptive subtraction may cause the removal of the primary reflections from the seismic data in addition to the multiples because of the overlapping primary and multiple reflections. Moreover, a significant requirement of SRME is that it requires a dense and regular spatial distribution of sources and receivers, and knowledge of the acquisition wavelet (Dragoset *et al.*, 2010). Reconstructing the missing near-offset traces and extending the offsets to the zero-offset is a computationally demanding and expensive requirement of the SRME method (Dragoset *et al.*, 2010).

In earlier work (Kiraz *et al.*, 2023a,b), we demonstrate successful implementations of the CNN-based free surface multiple attenuation and deghosting for unblended data. However, the interference of primary and multiple reflections of one shot with the primary and multiple reflections from another shot, and overlapping multiples with different periodicities blended over the same receiver location make the free-surface multiple attenuation and ghost reflection removal harder for the blended data. In this paper, we propose a CNN-based solution for attenuating the free-surface multiples and removing the source and receiver-side ghost reflections, simultaneously, for blended seismic data. Our CNN-based solution operates on individual traces and is not sensitive to the missing near-offset traces, missing traces, and irregular/sparse acquisition parameters (e.g., for ocean-bottom node acquisition and time-lapse monitoring studies). In Section 6.2, we give details about the CNN network. We train a network of one-dimensional (1-D) convolutional layers using two-dimensional (2-D) sub-models of Marmousi velocity model (Versteeg, 1994) and Pluto velocity model (Stoughton *et al.*, 2001). In Section 6.3, we present predictions using the Sigsbee velocity model (Paffenholz *et al.*, 2002) and the Mobil AVO Viking Graben data set (Keys & Foster, 1998). We show that our CNN-based algorithm attenuates the free-surface multiples and removes ghost reflections from previously unseen blended data and demonstrate that the solution generalizes well for field data application.

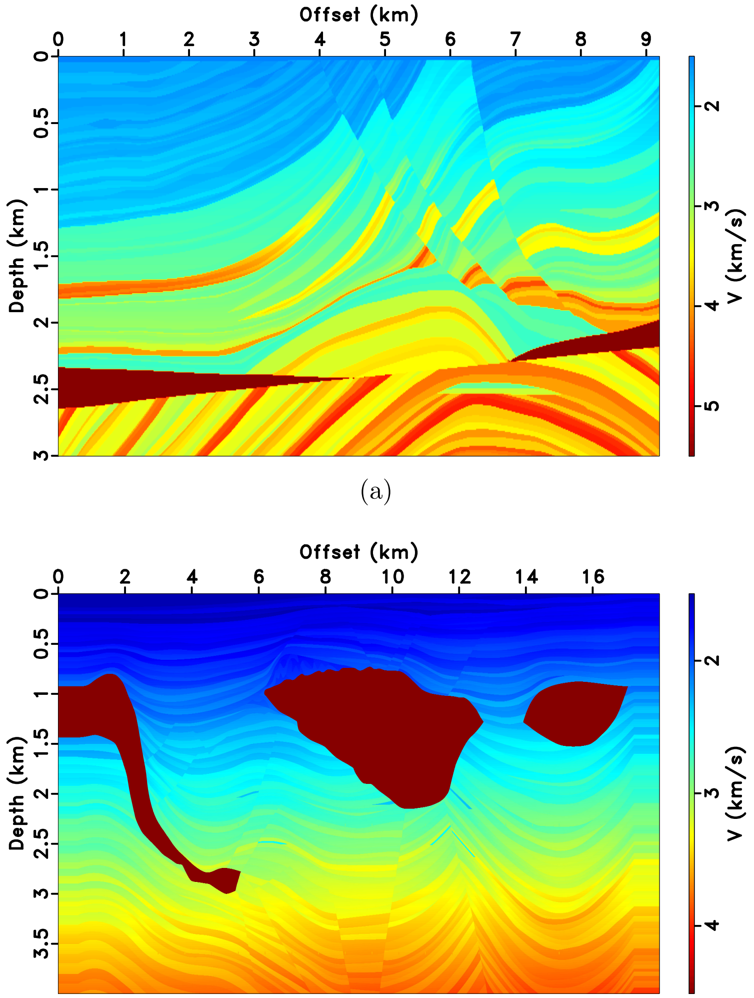
6.2 CNN Architecture and Training

Machine learning consists of a set of methods that can reveal patterns in data, and make predictions for new data by using previously learned patterns (Bishop, 2006; Murphy, 2012). Convolutional Neural Networks (CNNs) are important building blocks in deep learning methods and they have been widely incorporated in image recognition studies (Géron, 2019). CNNs can extract complex information and reveal complex patterns in the data (Bishop, 2006; Ekman, 2021; Géron, 2019).

CNNs have been used to address various problems in exploration seismology such as obtaining an elastic subsurface model using recorded normal-incidence seismic data (Das *et al.*, 2019), 3-D seismic fault segmentation (Wu *et al.*, 2019), seismic interpretation (Di *et al.*, 2020), seismic deghosting (Almuteri & Sava, 2023), 1-D wavefield focusing (Kiraz & Snieder, 2022), deblending simultaneous source seismic data (Cheng *et al.*, 2022), and attenuating free-surface multiples and removing ghost reflections from seismic data (Kiraz *et al.*, 2023a,b).

We use the CNN architecture proposed by Kiraz *et al.* (2023a) for attenuating free-surface multiples and removing source and receiver-side ghosts reflections from blended data. For the training model selection, we extract 15 sub-models from the Marmousi model which are shown in Figure 6.1(a). We also extract 37 sub-models from the Pluto model as shown in Figure 6.1(b). We add a water layer 500 m thick to the top of each model with 1500 m/s velocity and 1 g/cm^3 density. Each one of the 52 models used for training has 2.4 km horizontal extent and 2.4 km depth which includes the water column, 600 receivers, and two source locations for modeling the blended data. For each velocity model, we use a density model below the water bottom that is a scaled version of the velocity model by a factor two (and the units are converted from km/s to g/cm^3). The reflection coefficient at the water bottom thus obtained is comparable to the seafloor reflection coefficient given by Kim *et al.* (2004).

In simultaneous source acquisition, multiple sources are fired at different locations. Berkhout (2008) represents the blended seismic data as a linear combination of single sources using a blending operator that acts on one of the recorded waveforms. Following Berkhout (2008), for the synthetic data, we numerically create blended data using two unblended shot records fired at different locations and recorded at the same receiver locations with a blending operator. Figure 6.2 shows an illustration of the acquisition geometry for modeling the unblended shots used for the synthetic examples in this paper. In Figure 6.2, the red and blue explosion marks denote different source locations, and the white triangles denote the receiver locations. Each row in Figure 6.2 represents a blended shot record, the red mark being the source location 1 and the blue mark being the source location 2 (to which the blending operator is applied). The source and receiver depths stay the same at each row in Figure 6.2, and the receivers that record each of the unblended shot records are located at the same locations.



(a)

(b)

Figure 6.1 (a) Marmousi velocity model. (b) Pluto velocity model.

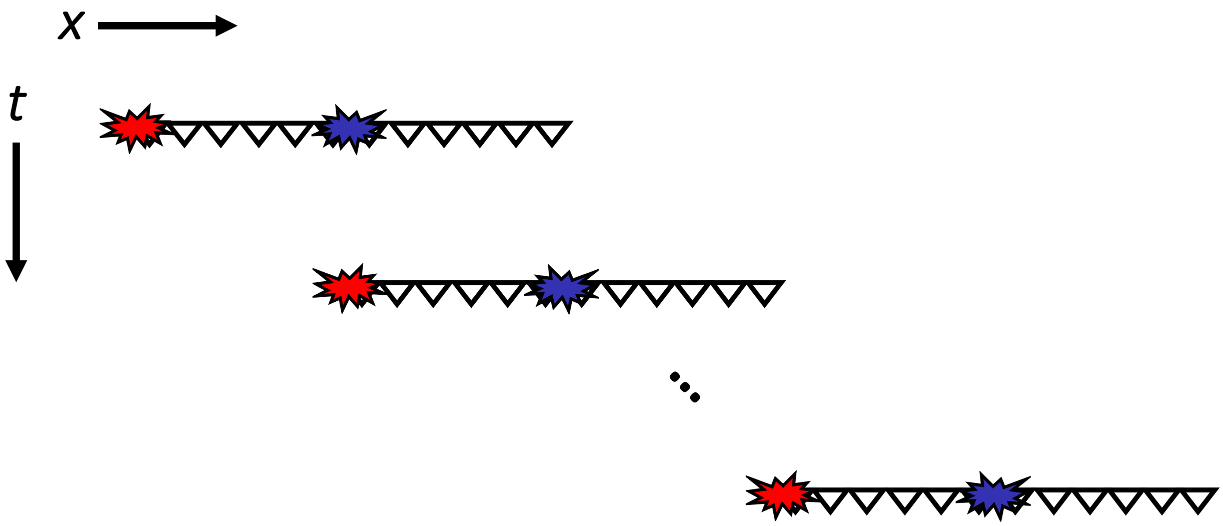
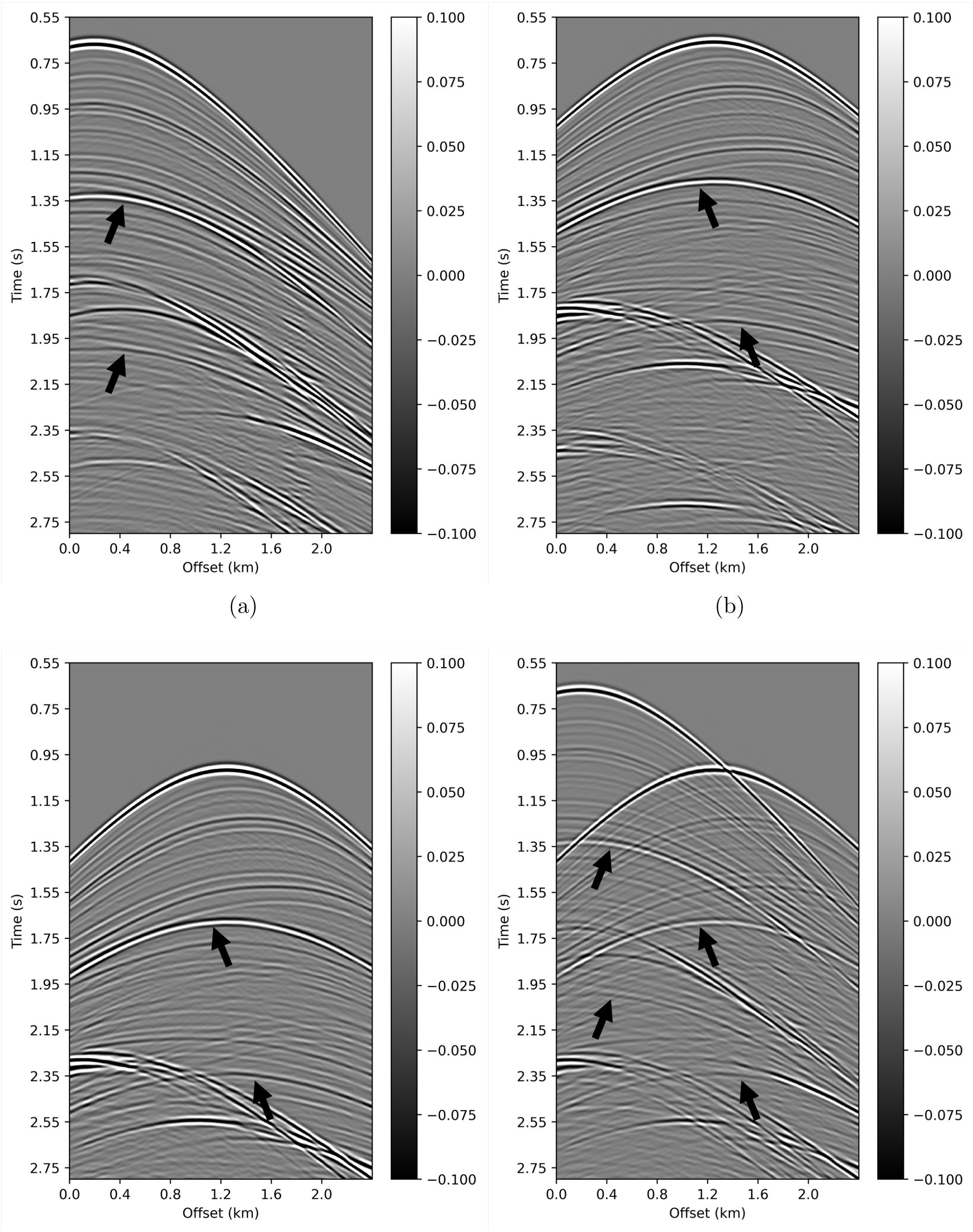


Figure 6.2 Illustration of the acquisition geometry for modeling the unblended shots for the synthetic data. Red and blue explosion marks denote different shot locations, and the white triangles denote the receiver locations. Each row represents a blended shot record for two source locations (red and blue explosion marks).

As an example of the blending process for the synthetic data, Figure 6.3(a) shows the shot record modeled at 0.2 km horizontal position at 7 m depth after removing the direct arrival. We remove the direct arrival from the recorded data during the training and prediction steps for both the synthetic and the field data. Figure 6.3(b) shows the shot record modeled for a source at 1.25 km horizontal position at 7 m depth after removing the direct arrival. Figure 6.3(c) shows the shot record in Figure 6.3(b) after adding a time-shift operator of 0.35 s which is constant for the training data. Figure 6.3(d) shows the blended shot record by combining two unblended shot records shown in Figure 6.3(a) and Figure 6.3(c). For the synthetic examples, the 600 receivers are placed along 2.4 km horizontal extent with the receiver sampling of 4 m at a depth of 8 m. The arrows in Figure 6.3 point to some of the free-surface multiples and indicate the presence of overlapping primaries and multiples in the blended data in Figure 6.3(d).



(a)

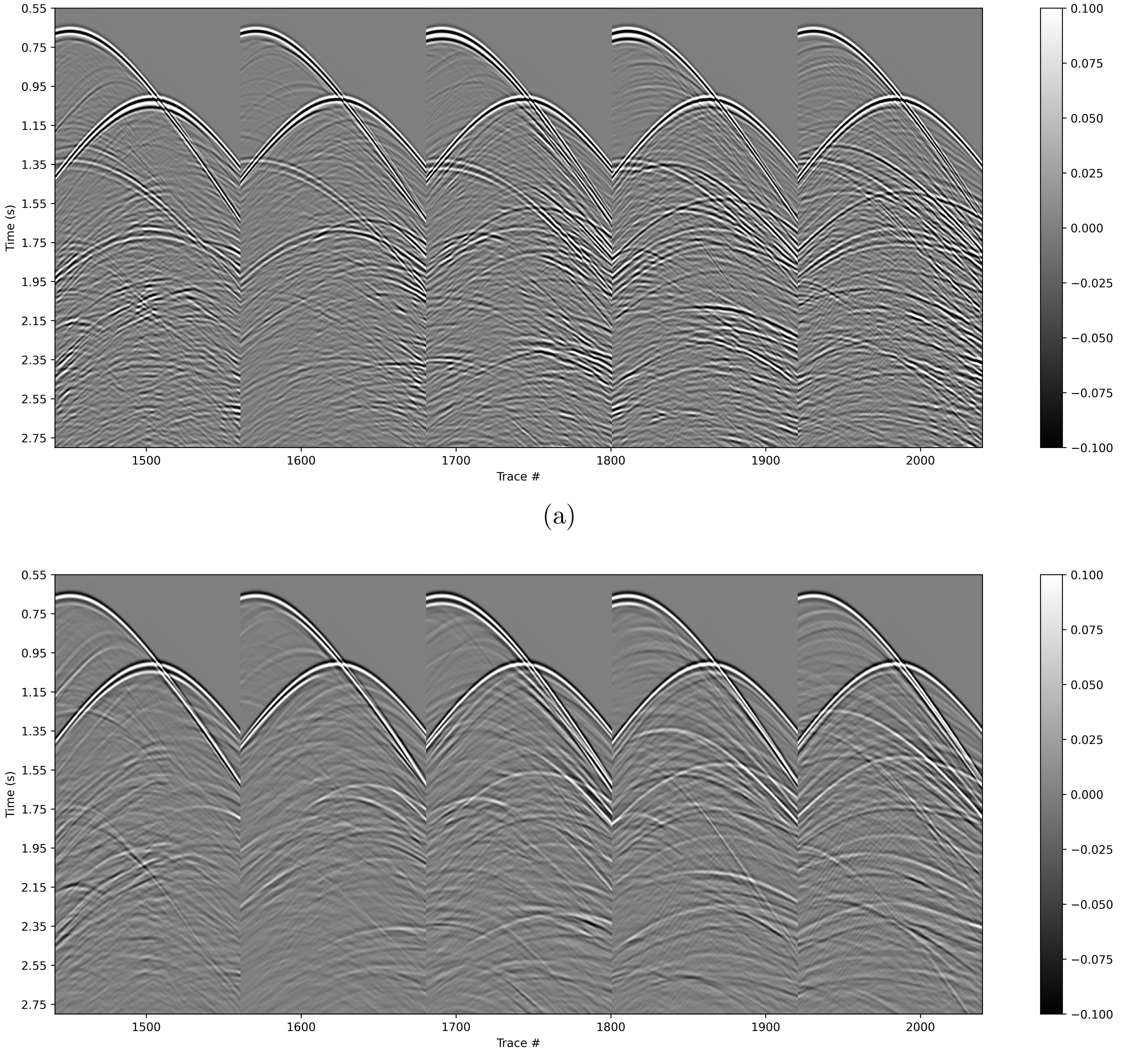
(b)

(c)

(d)

Figure 6.3 (a) Unblended shot record modeled for a source at 0.2 km. (b) Unblended shot record modeled for a source at 1.25 km. (c) Shot record in (b) after adding 0.35 s time-shift. (d) Blended shot record. Black arrows point to the 1st- and 2nd-order multiples.

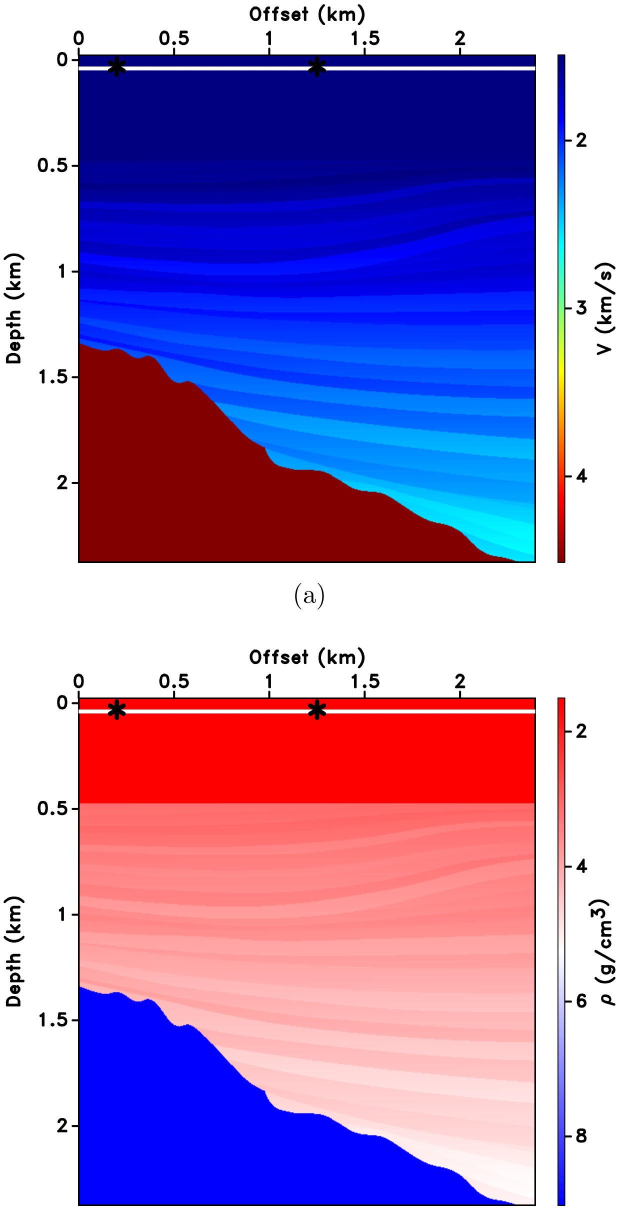
After creating the blended shot records using sub-models of the Marmousi and Pluto velocity models, Figure 6.4(a) shows 5 of 52 blended shot records of the data modeled with free-surface which are used as the training input after removing the direct arrival, and Figure 6.4(b) shows 5 of 52 blended shot records of the data modeled without free-surface which are used as the training output after removing the direct arrival.



(a)

(b)

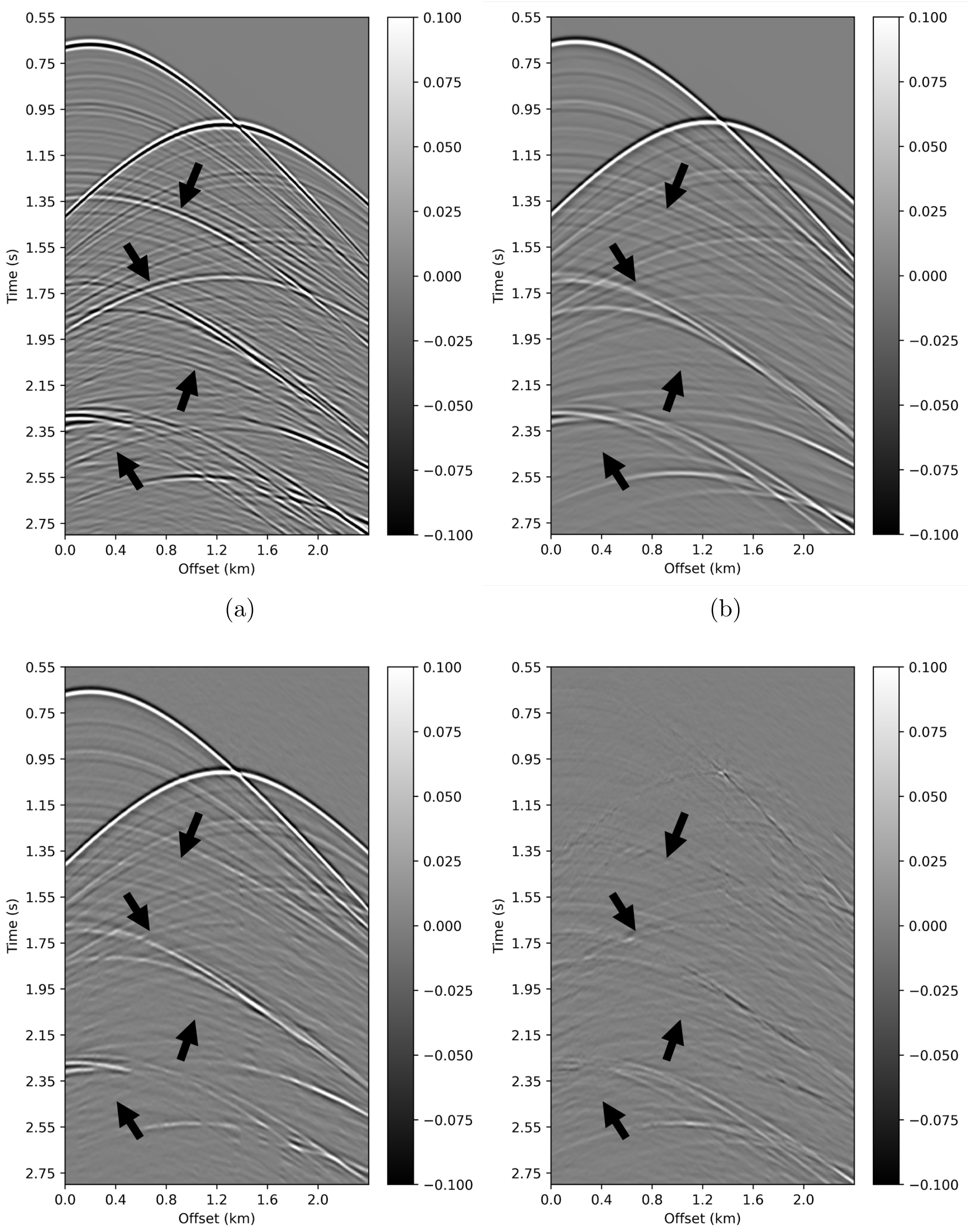
Figure 6.4 (a) 5 of 52 blended shot records of the input training data (*with* free-surface multiples and ghost reflections). (b) 5 of 52 blended shot records of the output training data (*without* free-surface multiples and ghost reflections).



(a)

(b)

Figure 6.5 (a) A sub-model of Sigsbee velocity model used for the prediction. (b) Corresponding density model of Sigsbee velocity model used for the prediction in (a).



(a)

(b)

(c)

(d)

Figure 6.6 (a) Numerically modeled shot record with the free-surface multiples (input data) obtained using the velocity model given in Figure 6.5. (b) Numerically modeled shot record without the free-surface multiples (output data/ground truth) obtained using the velocity model given in Figure 6.5. (c) Predicted shot record without the free-surface multiples using CNNs. (d) Difference between the ground truth data (Figure 6.6(b)) and the CNN prediction (Figure 6.6(c)). Black arrows point to the 1st- and 2nd-order multiples.

6.3 Numerical Examples

In this section, we demonstrate our CNN-based free-surface multiple attenuation and ghost removal for blended data with numerical examples. First, we illustrate the efficacy of our CNN-based algorithm using a synthetic example. For the synthetic example, we use two different shot records recorded at the same receiver locations to model the blended data. We then apply our CNN-based algorithm to a field data set. For the field data example, we use two different shot records recorded at different receiver locations to model the blended data. For the details of the CNN architecture used in this paper, we refer the reader to Kiraz *et al.* (2023a).

6.3.1 Synthetic Data Example

We first test our CNN algorithm using a sub-model of the Sigsbee velocity model, which was not used for the training, and Figure 6.5(a) shows the velocity model, and Figure 6.5(b) shows the corresponding density model used for modeling the prediction data. The white lines in Figure 6.5 denote the receiver locations at a depth of 8 m with a receiver sampling of 4 m, and the black asterisks denote the two different source locations used to create the blended shots with a time-shift of 0.35 s with free-surface multiples and ghost reflections at offsets 0.2 km and 1.25 km at a depth of 7 m. The water column is 500 m deep with a velocity of 1500 m/s and a density of 1 g/cm^3.

After the creation of synthetic data for prediction, we use our CNN-based algorithm to attenuate free surface multiples and source and receiver-side ghosts reflections.
Figure 6.6(a) shows the blended shot record modeled with free-surface multiples and ghost reflections using the velocity and density models shown in Figure 6.5, which is the input to the CNN network. Figure 6.6(b) shows the desired output, the numerically modeled blended shot record without free-surface multiples and ghost reflections using the velocity and density models shown in Figure 6.5. The black arrows in Figure 6.6 point to some of the free-surface multiples in the blended data. Figure 6.6(c) shows the blended data after

CNN-based free-surface multiple attenuation and ghost removal. Figure 6.6(d) shows the difference between the ground truth (Figure 6.6(b)) and the CNN prediction (Figure 6.6(c)). The black arrows in Figure 6.6(c) and Figure 6.6(d) point to the areas where the free-surface multiples have been suppressed in the CNN prediction. The difference plot in Figure 6.6(d) shows that the multiples caused by free-surface have been attenuated. However, Figure 6.6(d) also shows an amplitude mismatch between the desired and predicted waveforms, especially for times later than 2.35 s.

To investigate the efficacy of our CNN-based algorithm at various offsets for different subsurface models, we extract a velocity model from the Sigsbee velocity model which is shown in Figure 6.7. Following the steps described in Section 6.2 (also in Figure 6.3) for blended data creation, we use the velocity model shown in Figure 6.7 and model the blended shots starting from 0.2 km and 1.25 km positions with 80 m source intervals. We model 68 blended waveforms which are used as an input to the network, and Figure 6.8(a) shows the leftmost trace of each blended input shot record. Figure 6.8(b) shows the near-trace gathers from each blended shot record of the desired output (e.g., the leftmost trace from Figure 6.6(b)). Figure 6.8(c) shows the near-trace gathers from each blended shot record of the CNN prediction (e.g., the leftmost trace from Figure 6.6(c)). The arrows in Figure 6.8(a) point to the areas where the multiples caused by the free-surface such as 1st- and 2nd-order surface-related multiples and top-salt peg-leg multiples in the near-trace gather, and in Figure 6.8(c) these multiples have been attenuated by the application of the CNN. Figure 6.8 shows that the free-surface multiples at times around 1.35 s and 1.9 s, and the peg-leg multiples at times around 2.4 s have been suppressed at the near-offsets. The hyperbolic arrivals in Figure 6.8 correspond to diffractions from the salt model in Figure 6.7.

Figure 6.9(a) shows the traces extracted from the middle of each blended input shot record (e.g., the middle trace from Figure 6.6(a)) (which will be referred to as the mid-trace gather). Figure 6.9(b) shows the mid-trace gathers of the desired output (e.g., the middle trace from Figure 6.6(b)). Figure 6.9(c) shows the mid-trace gathers of the CNN prediction (e.g., the middle trace from Figure 6.6(c)). Figure 6.9(c) shows that the 1st- and 2nd-order surface-related multiples and top-salt peg-leg multiples indicated by the arrows in Figure 6.9(a) have been attenuated. Figure 6.9 shows that the free-surface multiples at times around 1.5 s and 1.65 s, and the peg-leg multiples at times around 2 s and 2.3 s have been suppressed at the mid-offsets.

Figure 6.10(a) shows the traces extracted from the rightmost of each blended input shot record (e.g., the rightmost trace from Figure 6.6(a)) (which will be referred to as the near-trace gather). Figure 6.10(b) shows the far-trace gathers of the desired output (e.g., the rightmost trace from Figure 6.6(b)). Figure 6.10(c) shows the far-trace gathers of the CNN prediction (e.g., the rightmost trace from Figure 6.6(c)) and the multiples indicated by the black arrows in Figure 6.10(a) have been attenuated in the far-offsets. Figure 6.10 shows that the free-surface multiples at times around 1.9 s and 2 s, and the peg-leg multiples at times around 2.5 s have been suppressed at the far-offsets. The black arrows in Figure 6.8, Figure 6.9, and Figure 6.10 point to the areas where the free-surface multiples have been attenuated in the CNN prediction at various offsets.

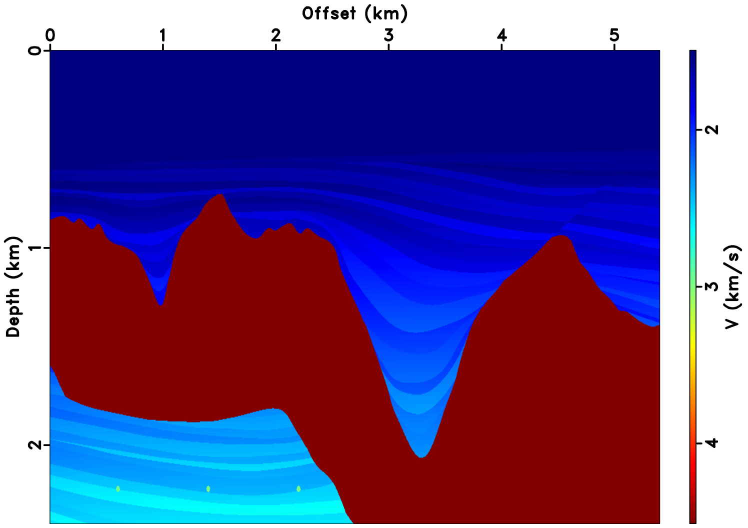
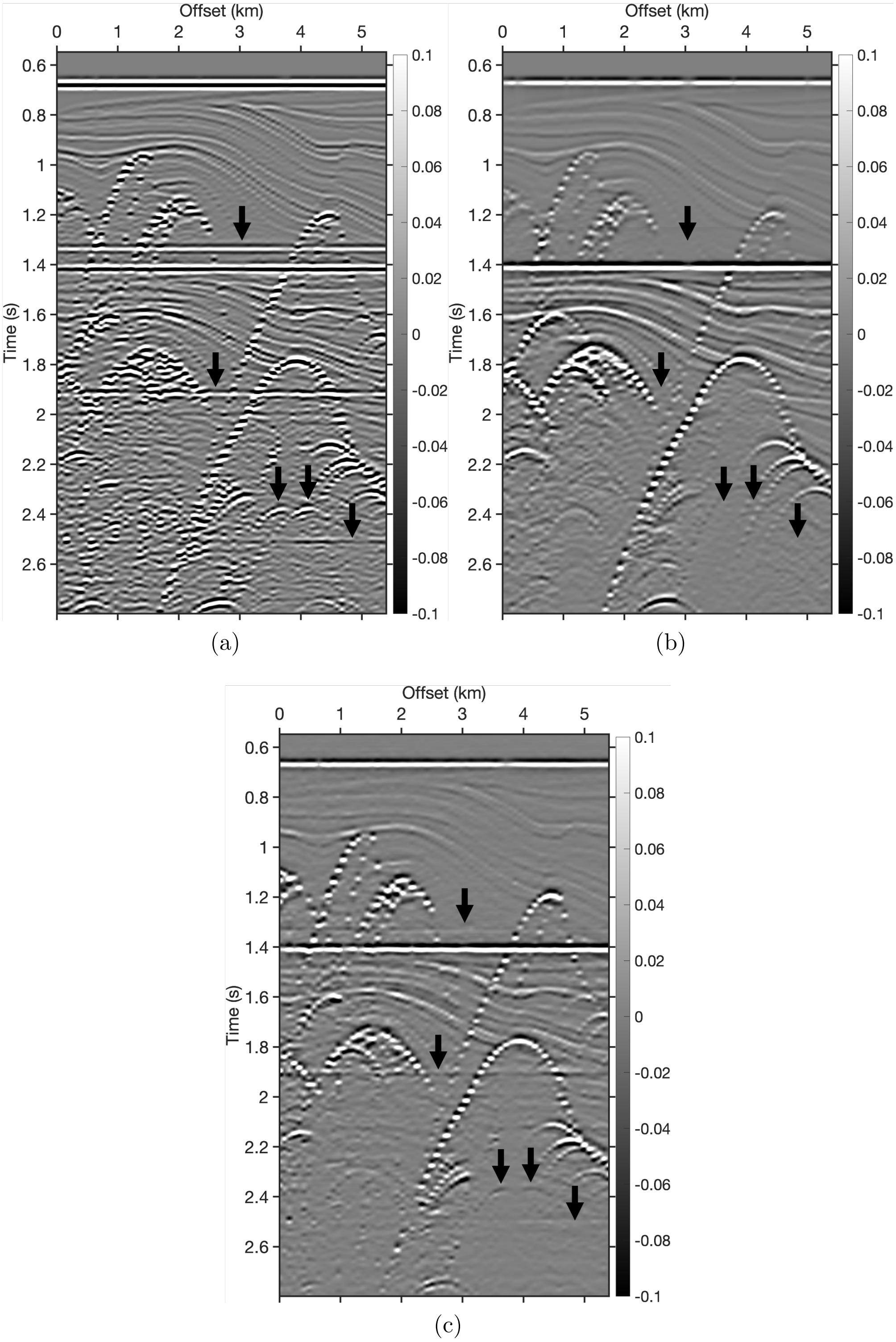


Figure 6.7 Extracted velocity model from the Sigsbee velocity model.

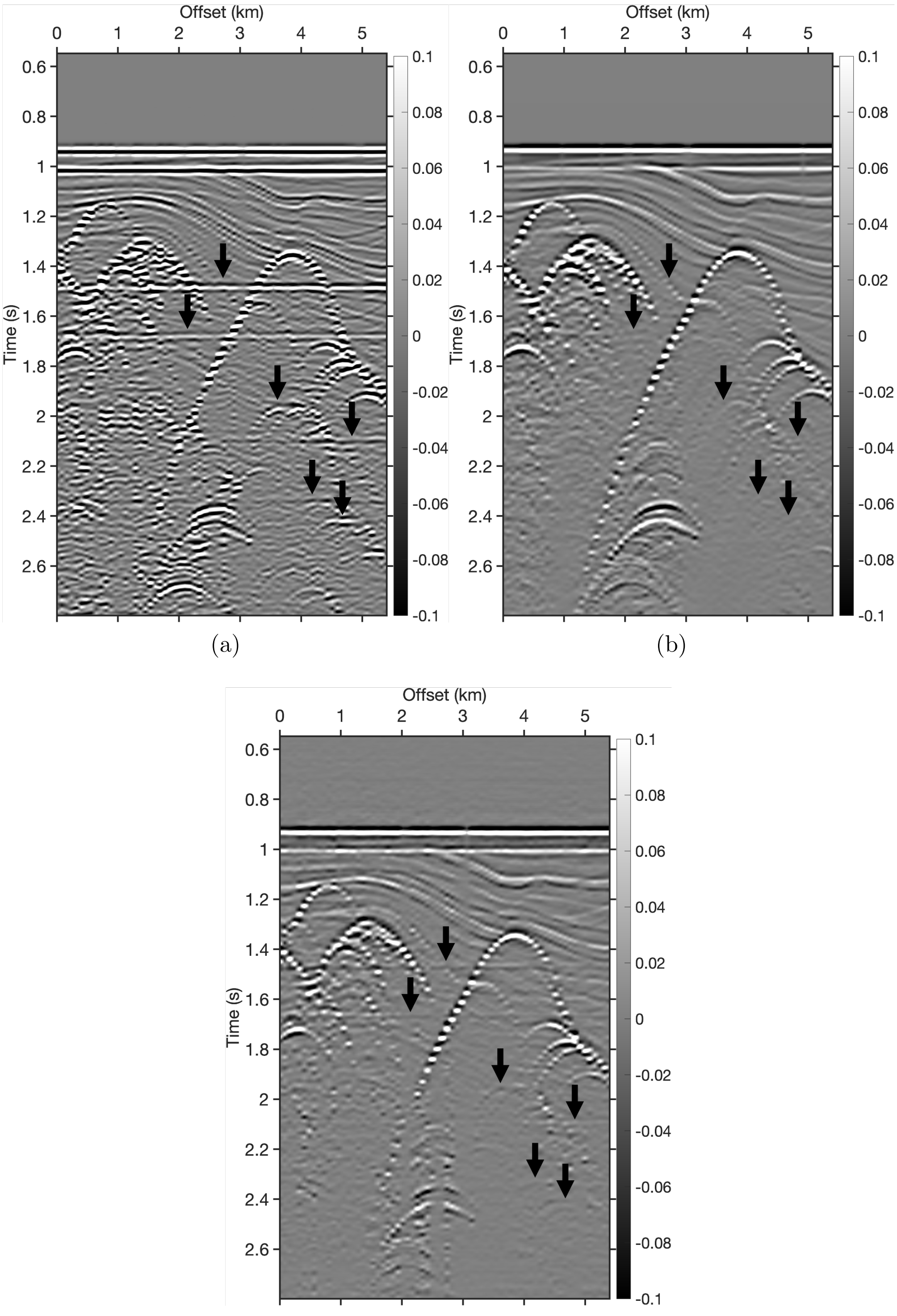


(a)

(b)

(c)

Figure 6.8 (a) Near-trace gather of the input blended data before free-surface multiple attenuation and deghosting. (b) Near-trace gather of the desired blended output. (c) Near-trace gather of the CNN prediction. Black arrows point to the 1st- and 2nd-order, and peg-leg multiples recorded at the near-offset section.

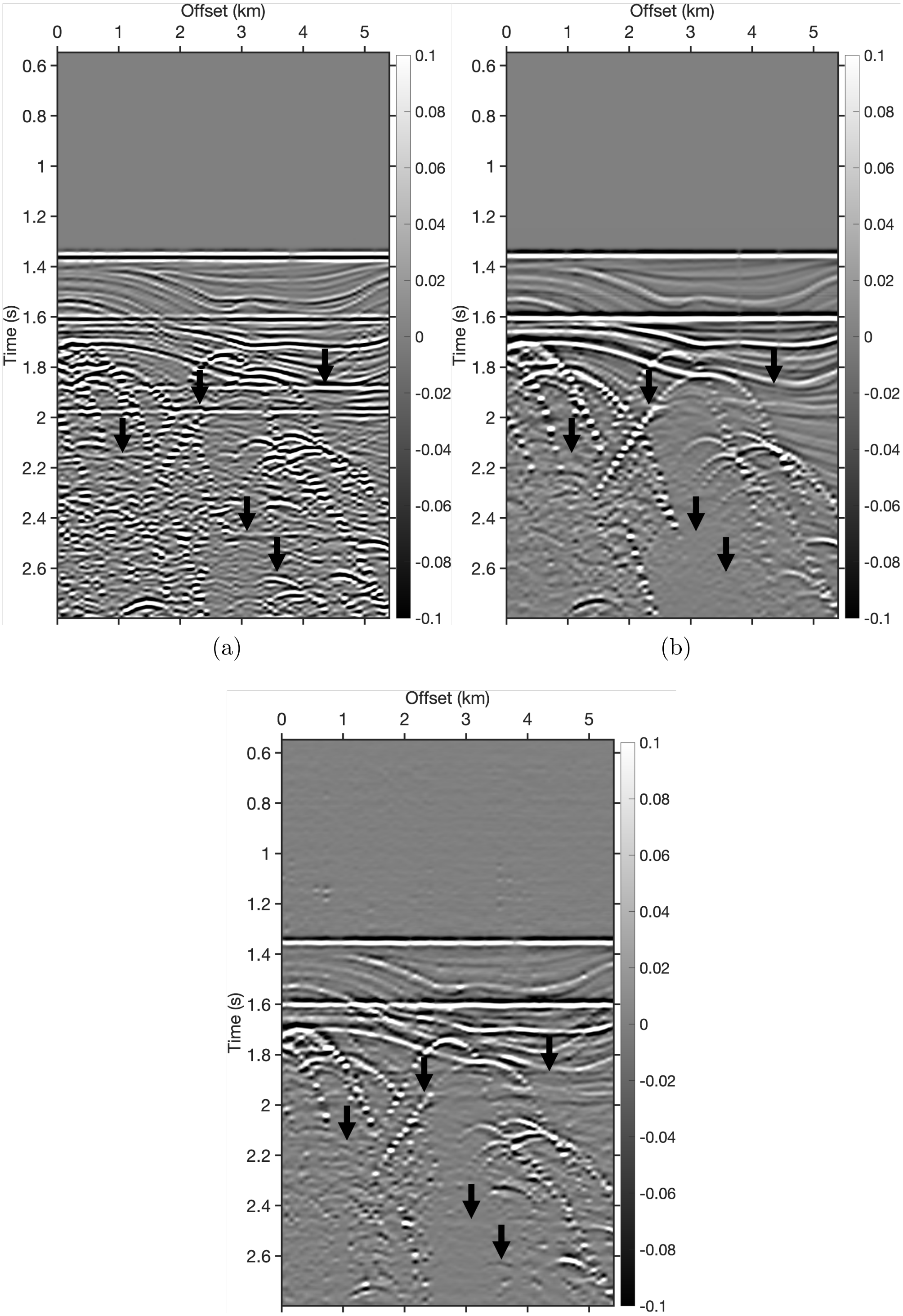


(a)

(b)

(c)

Figure 6.9 (a) Mid-trace gather of the input blended data before free-surface multiple attenuation and deghosting. (b) Mid-trace gather of the desired blended output. (c) Mid-trace gather of the CNN prediction. Black arrows point to the 1st- and 2nd-order, and peg-leg multiples recorded at the mid-offset section.



(a)

(b)

(c)

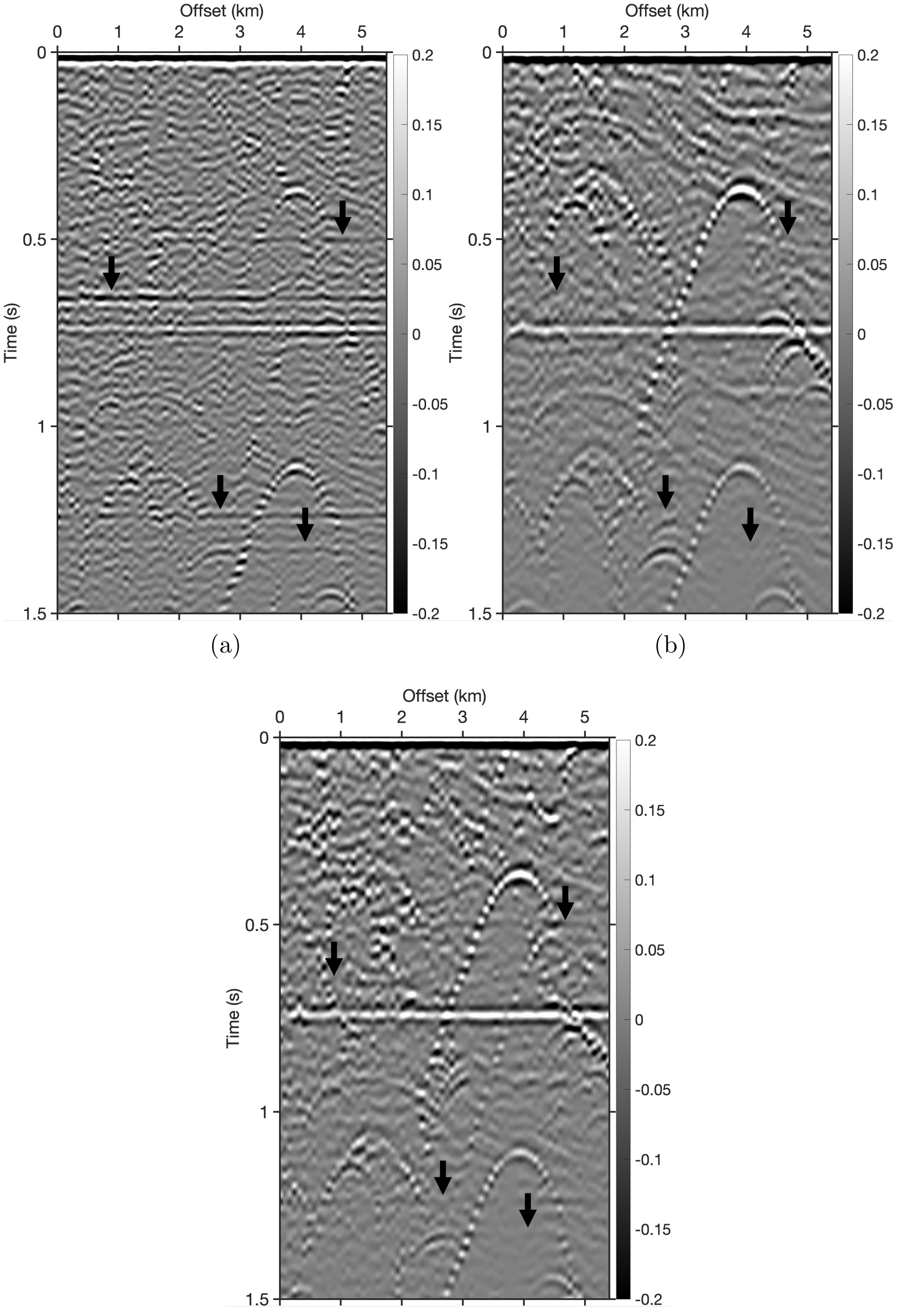
Figure 6.10 (a) Far-trace gather of the input blended data before free-surface multiple attenuation and deghosting. (b) Far-trace gather of the desired blended output. (c) Far-trace gather of the CNN prediction. Black arrows point to the 1st- and 2nd-order, and peg-leg multiples recorded at the far-offset section.

We use auto-correlation to investigate the periodicity associated with multiples (Yilmaz, 2001). Figure 6.11(a) shows the auto-correlation of the near-trace input data with free-surface multiples and ghost reflections shown in Figure 6.8(a), Figure 6.11(b) shows the auto-correlation of the near-trace desired data without free-surface multiples and ghost reflections shown in Figure 6.8(b), and Figure 6.11(c) shows the auto-correlation of the near-trace after application of the CNN-based algorithm shown in Figure 6.8(c). The periodicity of the water-bottom multiples is around 0.6 s, and the black arrows in Figure 6.11 point to the maxima that correspond to periodicities in the waveforms caused by the free-surface multiples that have been attenuated and their periodicity is not evident in the auto-correlation plot of the CNN prediction data.

We evaluate the CNN predictions for seismic deghosting by comparing the average amplitude spectra of the near-trace gathers shown in Figure 6.8. Figure 6.12 shows the average amplitude spectra on the log-log scale for the input near-trace gather shown in Figure 6.8(a) (red line) before free-surface multiple attenuation and ghost removal, the desired near-trace gather shown in Figure 6.8(b) (blue line) after free-surface multiple attenuation and ghost removal, and CNN prediction near-trace gather shown in Figure 6.8(c) (black line) after free-surface multiple attenuation and ghost removal. The average amplitude spectra in Figure 6.12 show that the CNN accurately recovers the low-frequency component of the spectrum, and the spectra of the numerically modeled and predicted match closely.

6.3.2 Field Data Example

For the field data application, we use the Mobil AVO Viking Graben data set acquired in the North Sea over the Viking Graben (Keys & Foster, 1998). This data set has been used to test different AVO methods and/or seismic data processing applications (Keys & Foster, 1998; Lumley *et al.*, 1998; Madiba & McMechan, 2003).



(a)

(b)

(c)

Figure 6.11 (a) Auto-correlation of the near-trace input blended data with free-surface multiples and ghost reflections shown in Figure 6.8(a). (b) Auto-correlation of the near-trace desired blended output without free-surface multiples and ghost reflections shown in Figure 6.8(b). (c) Auto-correlation of the near-trace CNN prediction shown in Figure 6.8(c). Black arrows point to the areas where the periodicity of free-surface multiples has been suppressed.

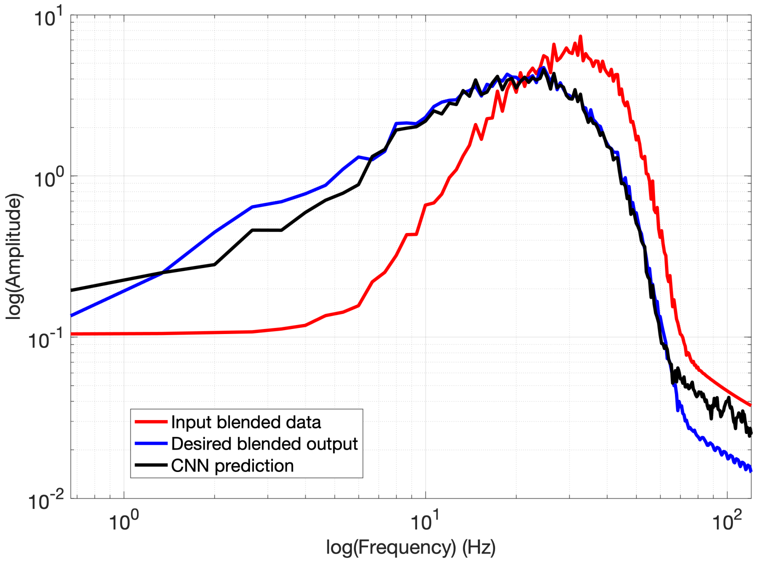


Figure 6.12 Average amplitude spectra in a log-log scale for the input near-trace gather shown in Figure 6.8(a) (red line), the desired near-trace gather shown in Figure 6.8(b) (blue line), and CNN prediction near-trace gather shown in Figure 6.8(c) (black line).

The Mobil AVO Viking Graben data set has 1001 shot records with 6 s recording length and a 4 ms sampling rate. There are 120 receivers for each show in a streamer that is towed along with the sources, and the near-offset receiver is located at 262 m and the far-offset receiver is located at 3237 m. The receivers are located at 10 m depth, and the air-gun array is placed at 6 m depth. Both the source and receiver group interval is 25 m, and the study area has a water depth of approximately 350 m (Keys & Foster, 1998; Madiba & McMechan, 2003).

For the CNN training for the field data application, we use the source and receiver geometry and the water depth of the Mobil AVO Viking Graben data set, and create synthetic data using sub-models of the Marmousi and Pluto velocity models. Different from the synthetic data examples, we first apply several pre-processing steps and blend the field data set. We then apply our CNN-based free-surface multiple attenuation and seismic deghosting algorithm to the blended field data set.

As a first pre-processing step, we remove the direct arrival, and account for the geometric spreading in both the synthetic and field data sets. For the training data, we account for the 2-D effects of geometric spreading of the synthetic training data set by applying a $\sqrt{t}$ gain, and the three-dimensional (3-D) effects of geometric spreading of the field data set by applying a t gain (Liner, 2016). We only use the first 3.5 s of the seismic data and apply a band-pass filter with 15 Hz and 50 Hz pass bands to the field data set to match the frequency content of the training data. Lastly, we normalize the synthetic data by the maximum amplitude prior to training, and the field data set by the maximum amplitude prior to prediction (Géron, 2019).

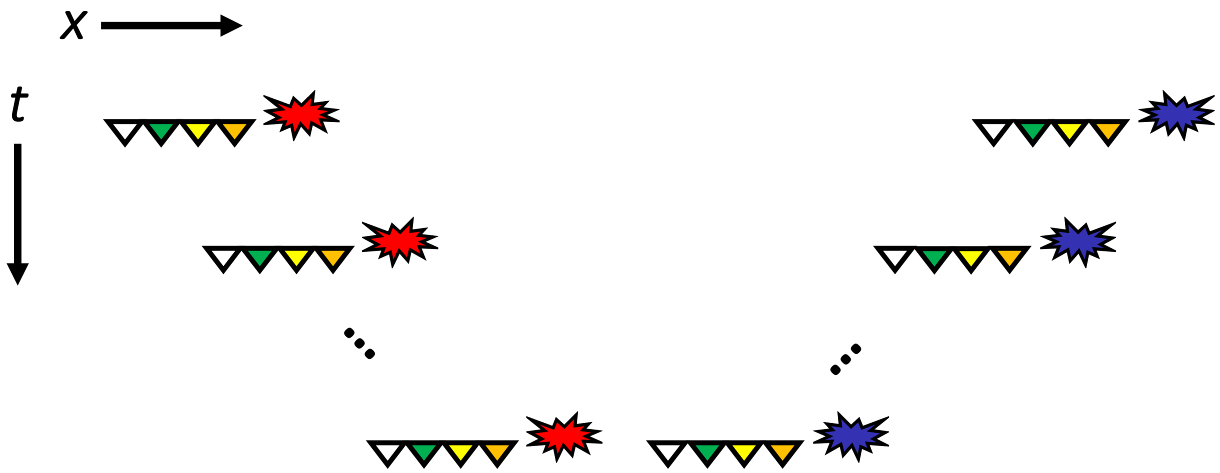


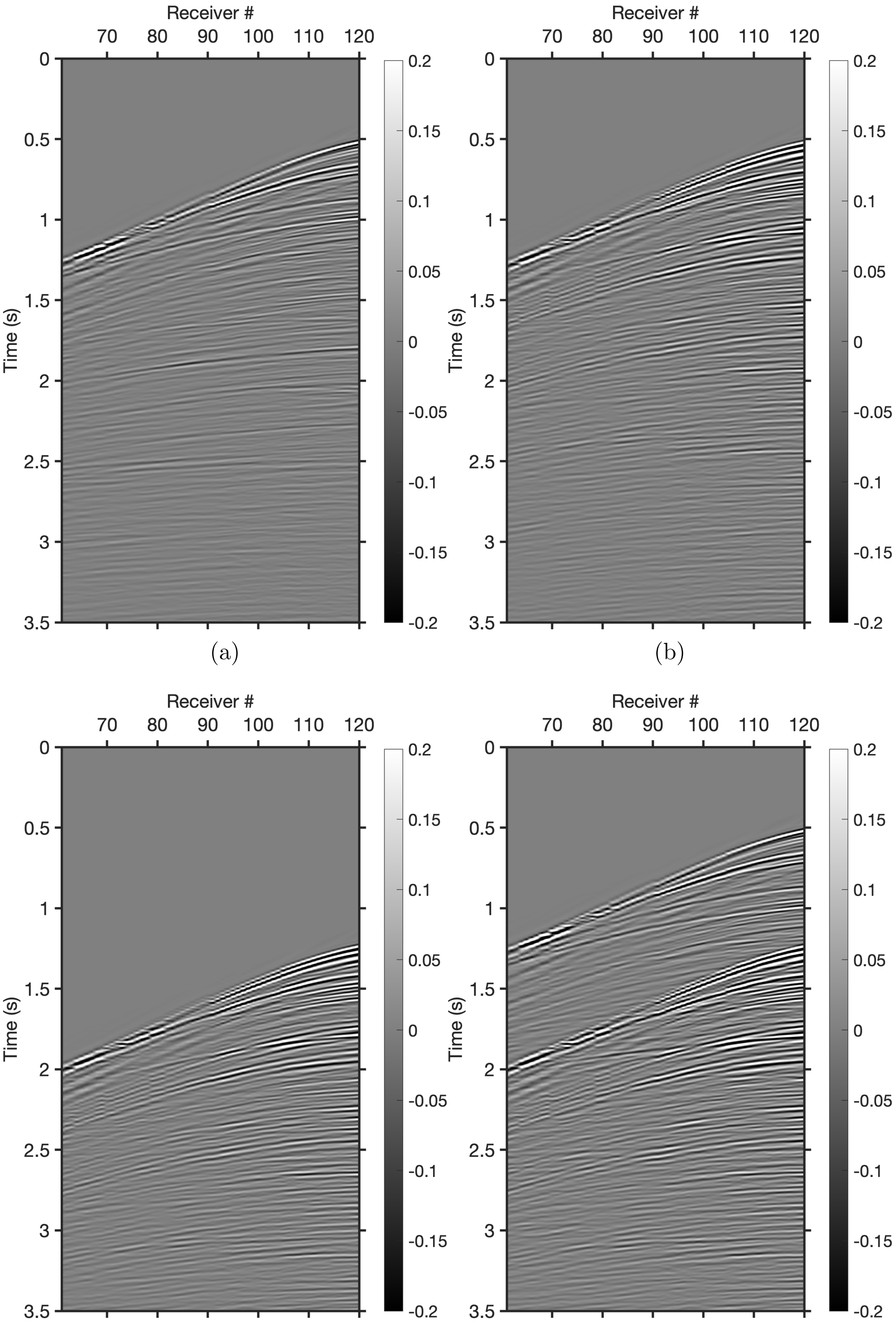
Figure 6.13 Illustration of the acquisition geometry for modeling the unblended shots for the Mobil data. Red and blue explosion marks denote the source locations and the triangles denote the receiver locations. Each row represents a blended shot record for two source locations (red and blue explosion marks). Data recorded by the same-colored receivers are blended.

The Mobil AVO Viking Graben data set is an unblended data set. After applying the pre-processing steps described above, we blend the Mobil AVO Viking Graben data set using two unblended shot records fired at different locations and recorded at different receiver locations with a blending operator. Figure 6.13 shows an illustration of the acquisition geometry for the unblended shots of the Mobil AVO Viking Graben data set. Similar to Figure 6.2, the red and blue explosion marks denote different source locations, and the triangles denote the receiver locations. The data recorded by the same-colored receivers are blended. Each row in Figure 6.13 represents a blended shot record, the red mark being the source location 1 and the blue mark being the source location 2 (to which the blending operator is applied). Although the source and receiver depths stay the same at each row in Figure 6.13, different from Figure 6.2, receivers that record each of the unblended shot records are located at different locations.

Figure 6.14 shows an example of the blending process for the field data set after applying the pre-processing steps described above. Figure 6.14(a) shows the shot record 750 of 1001 shot records from the field data. Figure 6.14(b) shows the shot record 252. Figure 6.14(c) shows the shot record in Figure 6.14(b) after 0.7 s time-shift. Figure 6.14(d) shows the modeled blended shot record 750 for the field data set.

After blending the field data, we use our CNN-based algorithm to attenuate free-surface multiples and ghost reflections. Figure 6.15(a) shows the blended shot record 750 (also shown in Figure 6.14(d)) before free-surface multiple attenuation and deghosting. We then use this blended data set as an input to the network and Figure 6.15(b) shows the CNN prediction after simultaneous attenuation of free-surface multiples and deghosting. The black arrows in Figure 6.15 point to the areas where free-surface multiples have been attenuated in the blended Mobil AVO Viking Graben data set.

Because the free-surface multiples in Figure 6.15 do not stand out, we present the near-offset sections where the multiples stand out more. Figure 6.16(a) shows the near-offset gather of the blended field data set before free-surface multiple attenuation and seismic deghosting. Figure 6.16(b) shows the near-trace gather of the blended data set after simultaneous free-surface multiple attenuation and seismic deghosting using CNNs. The black arrows in Figure 6.16 point to the suppression of multiples by the CNN-based algorithm. Figure 6.16 shows that the free-surface multiples have been suppressed by our CNN-based algorithm.



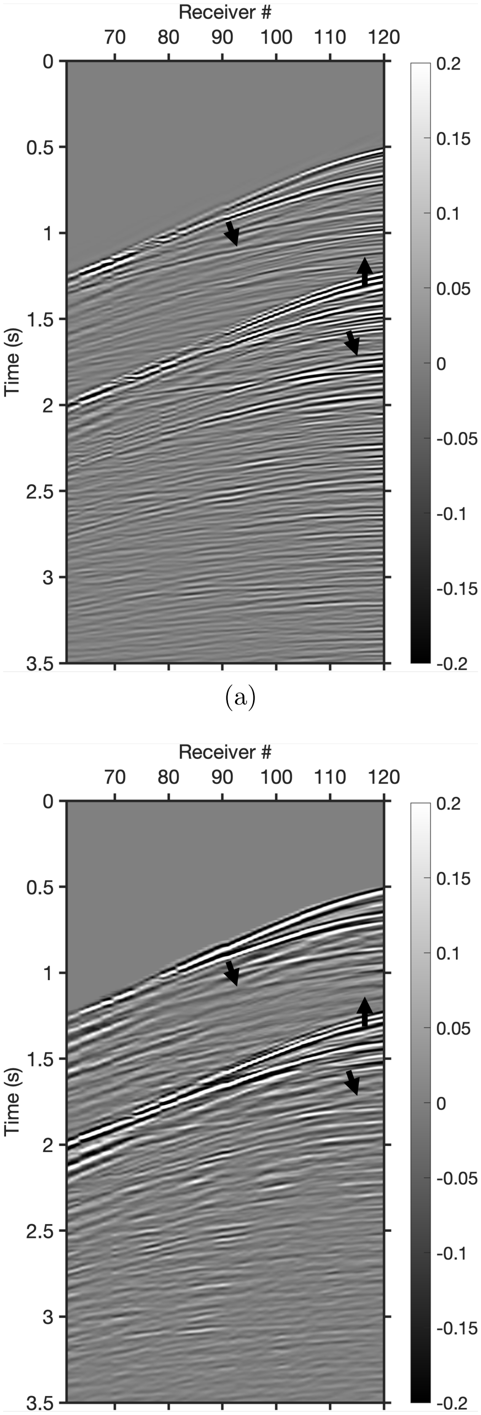
(a)

(b)

(c)

(d)

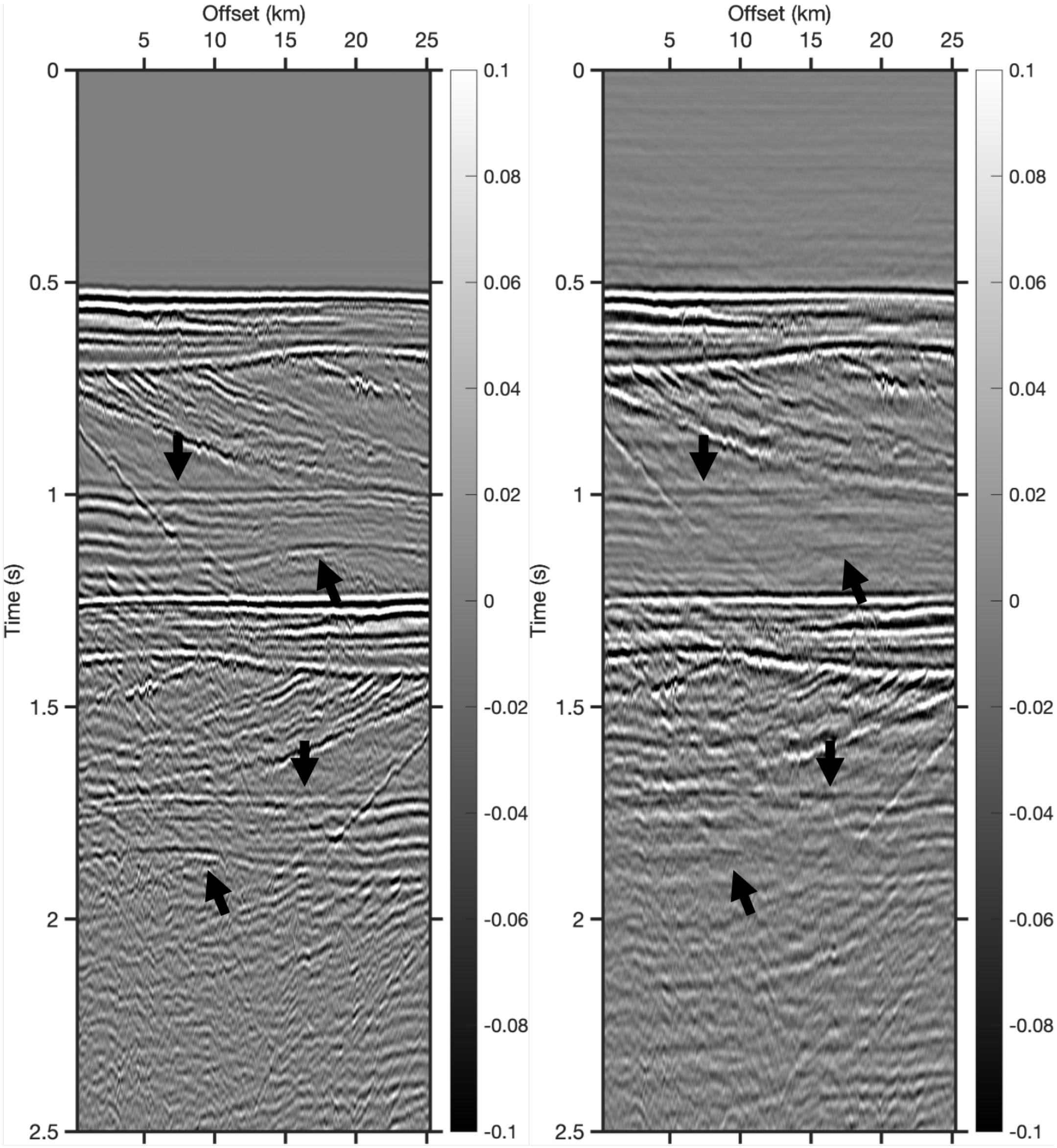
Figure 6.14 (a) Shot record 750 of the field data set. (b) Shot record 252 of the field data set. (c) Shot record 252 after 0.7 s time-shift. (d) Modeled blended data shot record 750 for the field data set.



(a)

(b)

Figure 6.15 (a) Shot record 750 of the blended field data set before free-surface multiple attenuation and deghosting. (b) Shot record 750 of the blended field data set after free-surface multiple attenuation and deghosting. Black arrows point to the areas where the free-surface multiples have been suppressed.



(a) (b)

Figure 6.16 (a) Near-trace gather of the blended field data before free-surface multiple attenuation and deghosting. (b) Near-trace gather of the blended field data after free-surface multiple attenuation and deghosting using CNNs. Black arrows point to the 1st- and 2nd-order free-surface multiples.

We start the blending process for the Mobil data set by blending the shot records from the leftmost (e.g., shot record 1) and rightmost (e.g., shot record 1001) of the Mobil AVO seismic line which is illustrated on the first row of Figure 6.13. Then we use shot records 2 and 1000, 3 and 999, and so on to create the blended shot records. The flipped appearance of the reflectors in Figure 6.16 around 1.2 s and 1.7 s of the reflections around 0.5 s and 1 s, respectively, is because of the blending process we use.

We evaluate the CNN predictions for seismic deghosting by comparing the average amplitude spectra before and after CNN-based free-surface multiple attenuation and seismic deghosting using the near-trace gathers shown in Figure 6.16. Figure 6.17 shows the average amplitude spectrum before free-surface multiple attenuation and seismic deghosting (red line) of the near-trace gather shown in Figure 6.16(a), and the average amplitude spectrum after free-surface multiple attenuation and seismic deghosting (blue line) using CNNs of the near-trace gather shown in Figure 6.16(b). The average amplitude spectra before and after deghosting in Figure 6.17 illustrate the recovery in the low-frequency component of the field blended data set as a result of the CNN-based seismic deghosting.

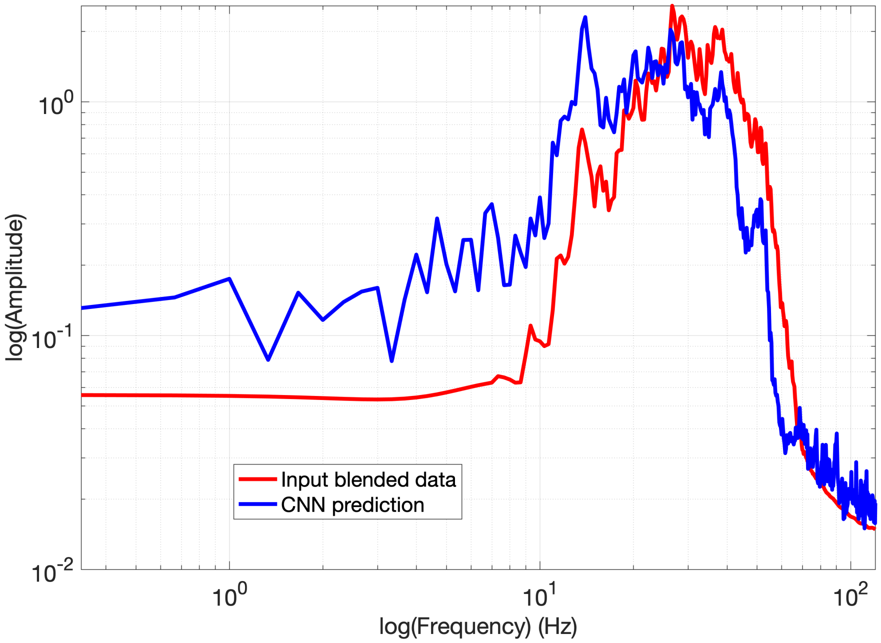


Figure 6.17 Average amplitude spectra in a log-log scale for the input near-trace gather shown in Figure 6.16(a) (red line) before free-surface multiple attenuation and deghosting, and CNN prediction near-trace gather shown in Figure 6.16(b) (blue line) after free-surface multiple attenuation and deghosting.

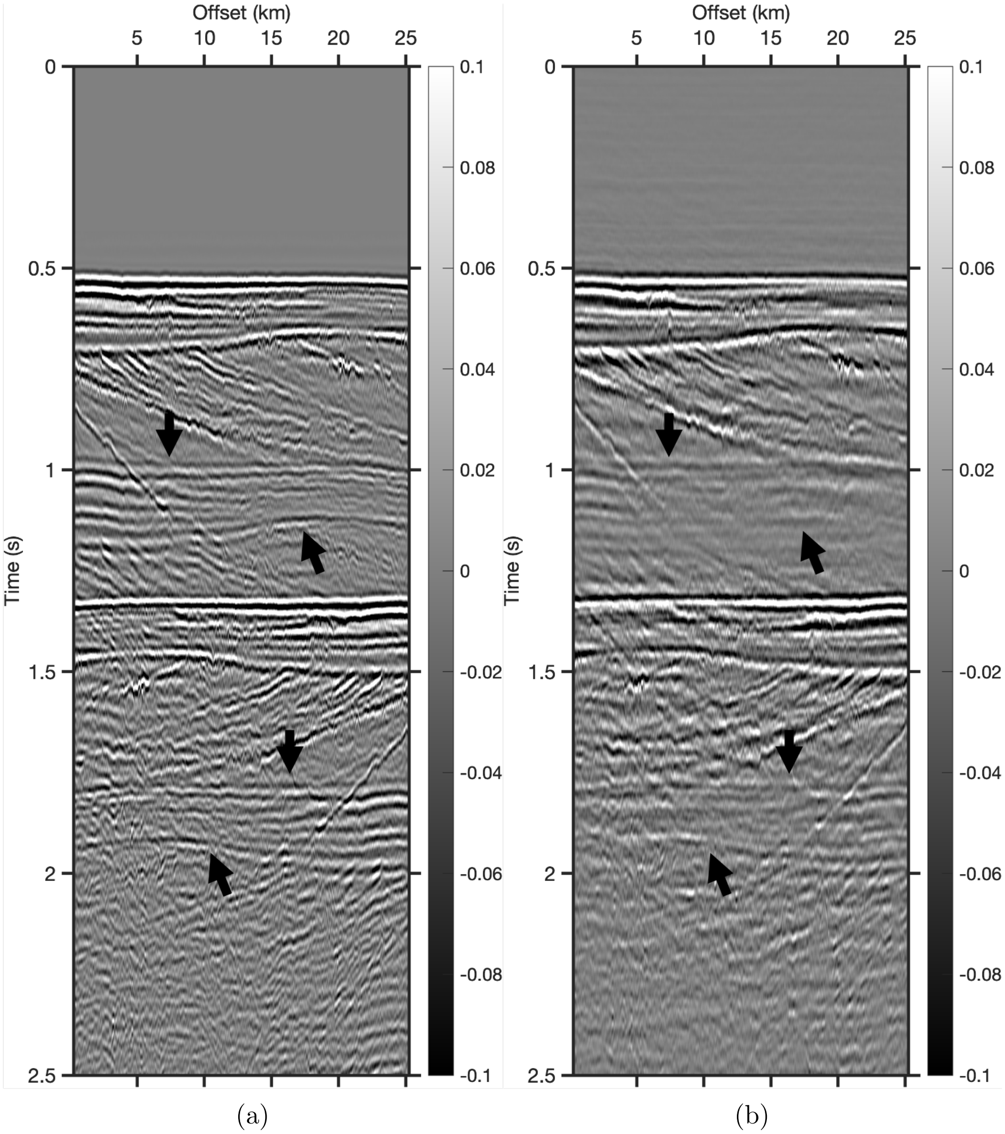


Figure 6.18 (a) Near-trace gather of the blended field data using a time-shift of 0.8 s before free-surface multiple attenuation and deghosting. (b) Near-trace gather of the blended field data using a time-shift of 0.8 s after free-surface multiple attenuation and deghosting using CNNs Black arrows point to the 1st- and 2nd-order free-surface multiples. Training of the network was based on 0.7 s and 1 s time-shifts.

To test the robustness of the CNN multiple suppression to variations in the dithering time, we use the blended data modeled using 0.7 s and 1 s delay times as the blending operator for training data, and we use the blended field data set using 0.8 s delay time for the CNN prediction. Figure 6.18(a) shows the near-offset gather of the blended field data set using 0.8 s delay time before free-surface multiple attenuation and seismic deghosting. Figure 6.18(b) shows the near-trace gather of the blended data set using 0.8 s delay time after simultaneous free-surface multiple attenuation and seismic deghosting using CNNs. The black arrows in Figure 6.18 point to the suppression of multiples by the CNN-based algorithm. This example shows that our CNN-based method attenuates the free-surface multiples and ghost reflections in the blended data created with different a time-shift that is unknown to the network.

6.4 Conclusions

We propose a CNN-based simultaneous free-surface multiple attenuation and seismic deghosting for blended data. The numerical examples we present show that our CNN-based approach removes free-surface multiples and ghost reflections from the blended data when two shots fired from different locations are used for blending. Our CNN-based algorithm does not suffer from the strong interference between the primary and multiple reflections of different shots used for blending. The field data application we present shows that the CNN-based free-surface multiple attenuation and deghosting can be generalized to the field data set applications when the CNN is trained with synthetic data. We show both in the synthetic and field blended data applications that the free-surface multiples have been attenuated, the low-frequency content of the blended seismic data has been recovered, and source and receiver-side deghosting has been successfully applied. We show that by using different delay times for training, the CNN-based algorithm we propose can be generalized further for delay times for the blended field data applications that have not been used in the training.

6.4.1 Acknowledgments

We thank Khalid Almuteri and Alejandro Jaimes from the Center for Wave Phenomena (CWP) for fruitful discussions. This work is supported by the Consortium Project on Seismic Inverse Methods for Complex Structures at the Colorado School of Mines. The

finite-difference forward modeling examples in this paper were generated using the Madagascar software package (Fomel *et al.*, 2013). Machine learning implementations were completed using the TensorFlow source library (https://www.tensorflow.org/).

CHAPTER 7
CONCLUSIONS

In this thesis, I propose algorithms to remove internal and free-surface multiples, and ghost reflections from seismic data. In Chapter 2, I introduce a two-dimensional (2-D) Marchenko equation that successfully retrieves the Green's function for a virtual source location in the subsurface. I show that the proposed Marchenko method does not require component separation for the Green's function retrieval, which forms the basis for obtaining multiple-free images of the subsurface.

In Chapter 3, I present a background velocity model dependence analysis of the Marchenko method on the reflected and refracted waves. The synthetic numerical tests demonstrate that in order to retrieve the Green's function using the Marchenko equation with high accuracy, a constant background model for the Marchenko algorithm suffices as long as the average slowness between the virtual source and receiver locations is known. I also show that the presence of refracted waves in the retrieved Green's function depends on the smoothness of the background velocity model.

In Chapter 4, I develop a Convolutional Neural Network (CNN)-based algorithm that simultaneously attenuates free-surface multiples and ghost reflections in seismic data. The developed algorithm operates on a single trace and does not require trace interpolation, extrapolation, or regularization for data gaps in seismic data, which can be expensive and time-consuming to fulfill. The proposed algorithm, therefore, can be applied to surveys with irregular acquisition such as ocean-bottom node seismic data or missing near-offset recordings for streamer surveys. Using the CNN-based algorithm, the attenuation of free-surface multiples does not require amplitude matching and/or subtraction steps unlike the industry-standard techniques (e.g., Surface-Related Multiple Elimination (SRME)). The synthetic numerical examples demonstrate that the CNN-based algorithm successfully removes free-surface multiples and ghost reflections at various offsets in seismic data.

In Chapter 5, I apply the algorithm developed in Chapter 4 to a field data set from the North Sea. Using only synthetic data to train the CNN, I show that the application of simultaneous free-surface multiple attenuation and deghosting can be generalized to field data, and the algorithm significantly attenuates free-surface multiples and applies deghosting, simultaneously, in the field data set. The results also show that the CNN-based algorithm preserves the primary reflections and increases their continuity while removing free-surface effects from the field data set.

In Chapter 6, I extend the CNN-based algorithm presented in Chapter 4 to the simultaneous source acquisition seismic data (or blended data). The effects of free-surface multiples in blended data are more significant than for unblended data because in the blended data there are more overlapping multiple and primary reflections from various offsets than in single-shot data. I apply the CNN-based algorithm to the field data set from the North Sea and show that the network trained with synthetic data can be applied to the blended field data set to remove free-surface effects. The numerical examples in this chapter demonstrate that the CNN-based algorithm successfully removes free-surface effects from blended data.

7.1 Future Research

The proposed algorithms in this thesis can be further generalized for different applications. Below I make recommendations for future research.

7.1.1 Imaging Through The Skull Using The Marchenko Equation

The Marchenko equation I develop in Chapter 2 enables wavefield focusing in an unknown complicated structure where the object is surrounded by sources/receivers where the up/down separation of the focusing function does not work. A possible application for this type source/receiver geometry can be imaging of the human brain. Guasch *et al.* (2020) present challenges related to successfully imaging inside the adult human skull, and use FWI for brain imaging. The human skull causes scattering of the acoustic waves and this limits the successful imaging of the human brain (Guasch *et al.*, 2020). The Marchenko equation I propose has the potential to create virtual sources within the brain so that the strong scattering caused by the skull can be eliminated. Propagating acoustic waves through the skull can then be achieved using the Marchenko equation I develop in Chapter 2.

7.1.2 Non-Destructive Testing Using The Marchenko Equation

Non-Destructive Testing (NDT) in civil engineering is used for assessing the quality and damages of structures like buildings, bridges, and dams (Müller *et al.*, 2012). Seismic imaging techniques like reverse time migration has been successfully implemented as an imaging technique for NDT (Müller *et al.*, 2012). One may apply the Marchenko equation I propose in Chapter 2 to retrieve the Green's function for virtual source locations, and by developing an imaging condition, imaging can be performed for NDT purposes when the object under investigation is surrounded by sources and receivers. However, developing an imaging condition for the Marchenko equation I propose needs further investigation.

7.1.3 Velocity Model Dependence Of Marchenko Imaging

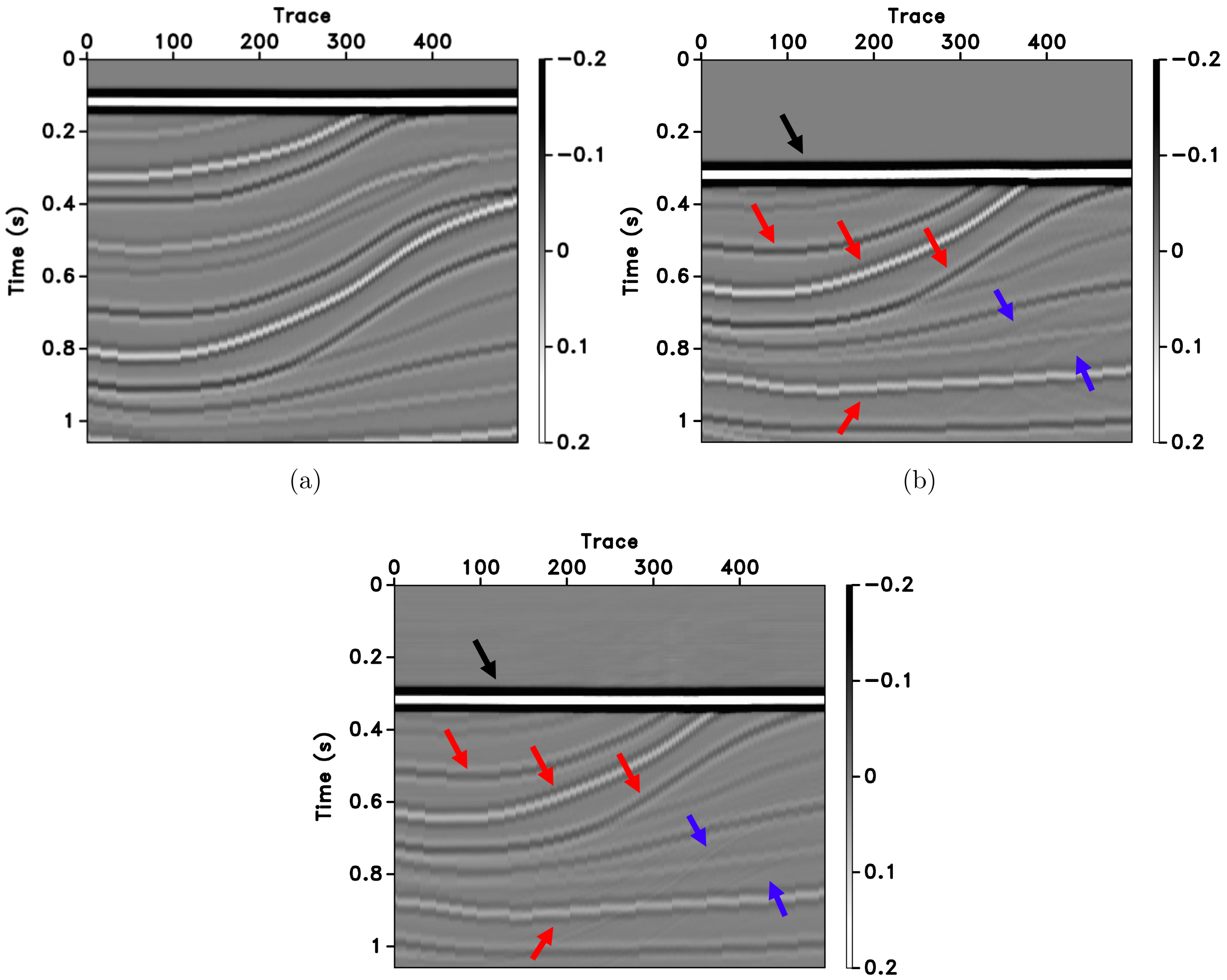
An initial velocity model is an important requirement of the conventional imaging algorithms (Baysal *et al.*, 1983), full-waveform inversion algorithms (Virieux & Operto, 2009), and Marchenko imaging algorithms (Wapenaar *et al.*, 2014). However, obtaining an accurate initial velocity model is not a trivial task. In Chapter 3, I show that a constant velocity model used for the Marchenko algorithm enables the retrieval of the Green's function with high accuracy if the average slowness is known between the receiver and the virtual source location. The proposed algorithm in Chapter 2 and examples in Chapter 3 have the potential to provide useful information about the velocity model dependence of the Marchenko imaging.

7.1.4 Focusing Waves In An Unknown 3-D Medium Without Wavefield Decomposition

In Chapter 2, I propose a Marchenko equation for wavefield focusing in an unknown medium and show that the Green's function for a virtual source location can be retrieved. The equation I propose is valid for 2-D media and holds equally well in three-dimensional (3-D) media. However, strategies for addressing implementation issues, in particular sampling and computational aspects, need further investigation.

7.1.5 Wavefield Focusing Using CNNs In Multi-Dimensional Media

Although not presented as a technical chapter in this thesis, I propose a CNN-based solution for wavefield focusing in one-dimensional (1-D) media (Kiraz & Snieder, 2022). By training a CNN using the surface-recorded data as the training input data, and the focused wavefield as the training output data, the proposed algorithm retrieves the Green's function for a virtual source location in 1-D media without the need to solve the Marchenko equation. Figure 7.1(a) shows the traces modeled for sources located at 0.1 km and receivers located at the surface (input data to the network), Figure 7.1(b) shows the desired output, the traces modeled for virtual sources located at 0.43 km in the subsurface and receivers located at the surface. Figure 7.1(c) shows the CNN prediction, all of which are presented in Kiraz & Snieder (2022). In Figure 7.1(b) and Figure 7.1(c), the black arrows point to the direct arrivals, the red arrows point to some of the primary reflections, and the blue arrows point to some of the internal multiples (although not as strong as the primaries) caused by the virtual source location in the subsurface. I show in Figure 7.1 that I can predict the Green's function for an arbitrary depth location and our algorithm only requires the surface-recorded data as the input. A further investigation on extending this CNN-based method into 2-D and 3-D media can be performed.



(a)

(b)

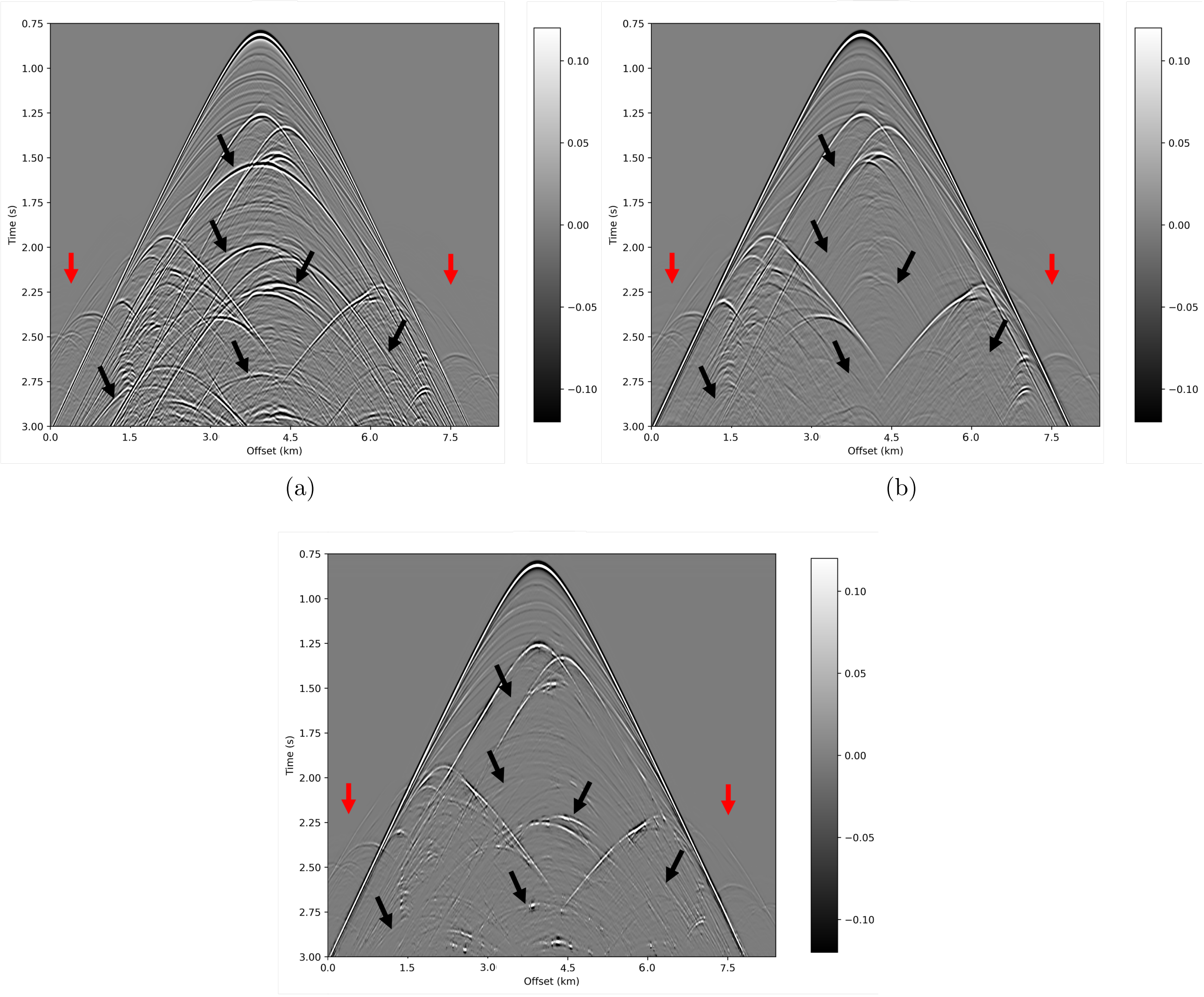
(c)

Figure 7.1 (a) Input data for the network, traces modeled for sources located at 0.1 km and receivers located at the surface. (b) Desired output, traces modeled for virtual sources located at 0.43 km in the subsurface and receivers located at the surface. (c) CNN prediction.

Additionally, Physics-Informed Neural Networks (PINNs) can also be used for solving the Marchenko equation. PINNs provide a neural network approach that satisfies any physical law described by general nonlinear equations (Raissi *et al.*, 2019). PINNs have successfully addressed problems like solving the wave equation (Moseley *et al.*, 2020; Song *et al.*, 2021), velocity inversion (Xu *et al.*, 2019), and anisotropic eikonal solution (Waheed *et al.*, 2020). One can also incorporate PINNs for solving the Marchenko equation and retrieving the Green's function in multi-dimensional media.

7.1.6 Ocean-Bottom Node Seismic Data

The use of Ocean-Bottom Node (OBN) for seismic surveys has become more popular over the last decade in seismic data acquisition, especially for time-lapse monitoring studies. The nodes in the OBN acquisition are placed on the sea bottom very sparsely (e.g., 1000 m apart) and in order to apply the SRME, one has to depend on the availability of a streamer data set which is not always the case. The CNN-based free-surface multiple attenuation I propose operates on a single trace and is not sensitive to the irregular and/or sparse receiver sampling of seismic data. Therefore, there is a potential to apply the CNN-based method on OBN data for free-surface multiple attenuation. Additionally, the OBN surveys can have large offsets where interference with refracted waves can be dominant. For instance, Figure 7.2 shows a test I did for removing free-surface effects from an OBN data set with 8 km offset. Figure 7.2(a) shows the input data to the CNN which is the numerically modeled data that have free-surface multiples and ghost reflections with receivers placed at the sea bottom with an 8 km horizontal extent. Figure 7.2(b) shows the desired output which is the numerically modeled data without a free surface. Figure 7.2(c) shows the CNN prediction. The black arrows in Figure 7.2 point to some of the free-surface multiples and the red arrows point to the areas where the refracted waves are present. Figure 7.2(c) shows that the free-surface multiples at around offset 4.5 km and 2.25 s and 2.75 s have not been eliminated in the CNN prediction. The refracted waves pointed with the red arrows have been preserved in Figure 7.2(c) and deghosting has been successfully applied. Although the CNN prediction in Figure 7.2(c) does not completely match the ground truth shown in Figure 7.2(b), the results can be further improved with hyperparameter tuning considering these are just initial test results. One can utilize CNN-based solutions for attenuating free-surface multiples at large offsets where reflections interfere with refracted waves.

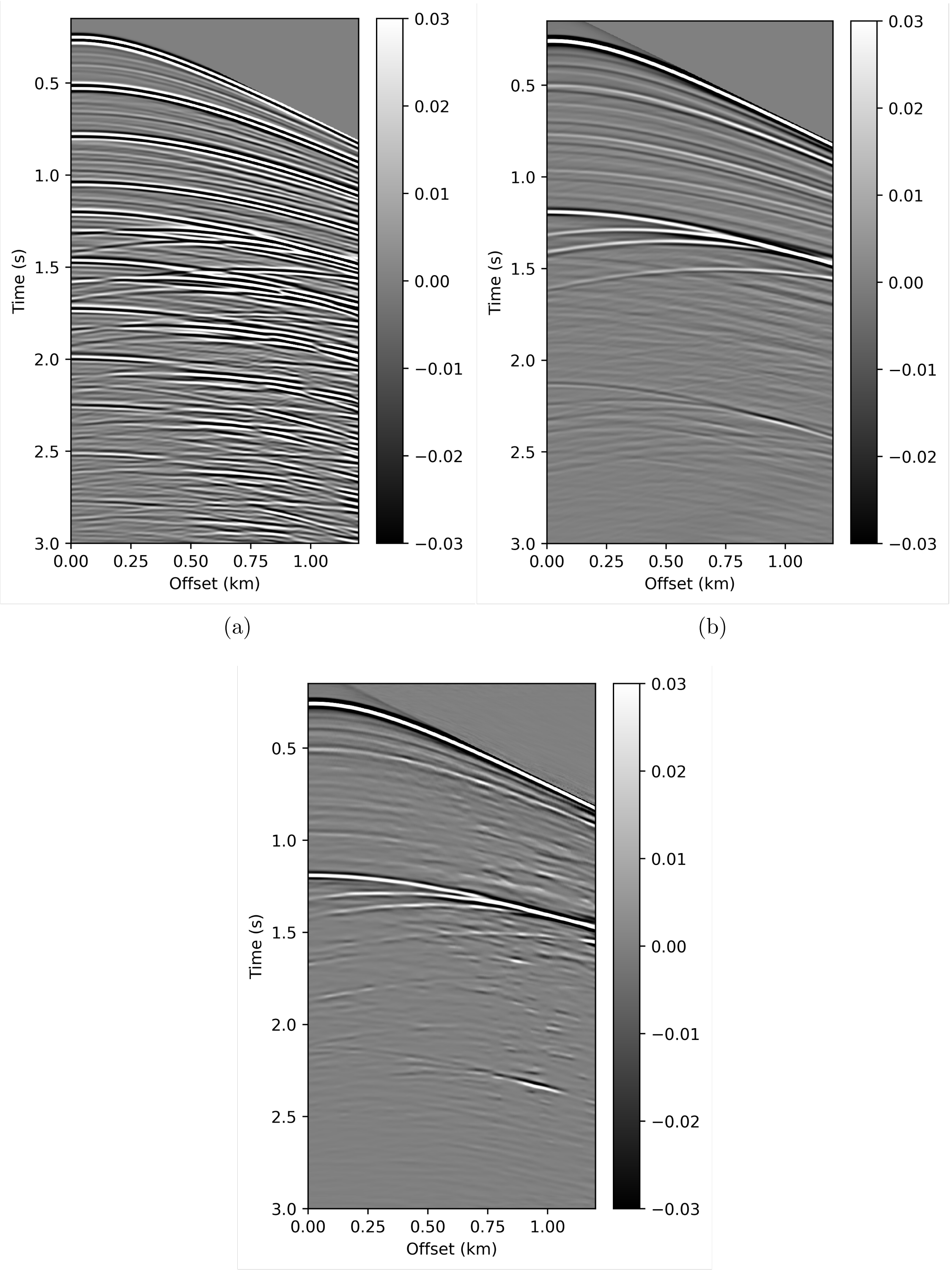


(a)

(b)

(c)

Figure 7.2 (a) Numerically modeled OBN data with 8 km offset *with* a free surface (input to the network). (b) Numerically modeled OBN data with 8 km offset *without* a free surface (desired output). (c) CNN prediction *without* a free surface.



(a)

(b)

(c)

Figure 7.3 (a) Numerically modeled data with 200 m water depth *with* a free surface (input to the network). (b) Numerically modeled data with 200 m water depth *without* a free surface (desired output). (c) CNN prediction *without* a free surface.

7.1.7 Short-Period Multiple Seismic Data

The elimination of free-surface multiples for short-period multiples from very shallow water depths (approximately 0-200 m) is another limitation of the SRME. The near-offset reconstruction and extending the offsets to zero offset in shallow-water data is significantly more important compared to the water depths larger than approximately 200 m (Dragoset *et al.*, 2010). Additionally, obtaining correct amplitudes in shallow-water data requires additional iterations of the SRME method (Dragoset *et al.*, 2010).

As an example, Figure 7.3 shows a test for removing free-surface effects from a data set with 200 m water depth. Figure 7.3(a) shows the input data to the CNN which is the numerically modeled data with free-surface multiples and ghost reflections with 200 m water depth. Figure 7.3(b) shows the desired output which is the numerically modeled data without a free surface. Figure 7.3(c) shows the CNN prediction. The CNN prediction shown in Figure 7.3(c) is unable to recover the amplitude of primary reflections for times later than 2 s. Although for the earlier times the free-surface multiples have been attenuated, the amplitude recovery in the CNN prediction does not match the ground truth shown in Figure 7.3(b). However, these are preliminary results and I have not worked on tuning the hyperparameters.

Unlike the SRME method, the CNN-based free-surface multiple attenuation algorithm I propose does not require near-offset reconstruction and has the potential to address some of the aforementioned limitations. However, strategies for attenuating short-period multiples using CNNs need further investigation.

7.1.8 Deblending Seismic Data

With the recent advancements of machine learning techniques, different successful machine learning applications of separating blended data into individual data sets (or deblending) have been performed (Cheng *et al.*, 2022; Sun *et al.*, 2020, 2022; Wang *et al.*, 2022; Zu *et al.*, 2020). The CNN architecture I propose can be generalized for deblending applications through training.

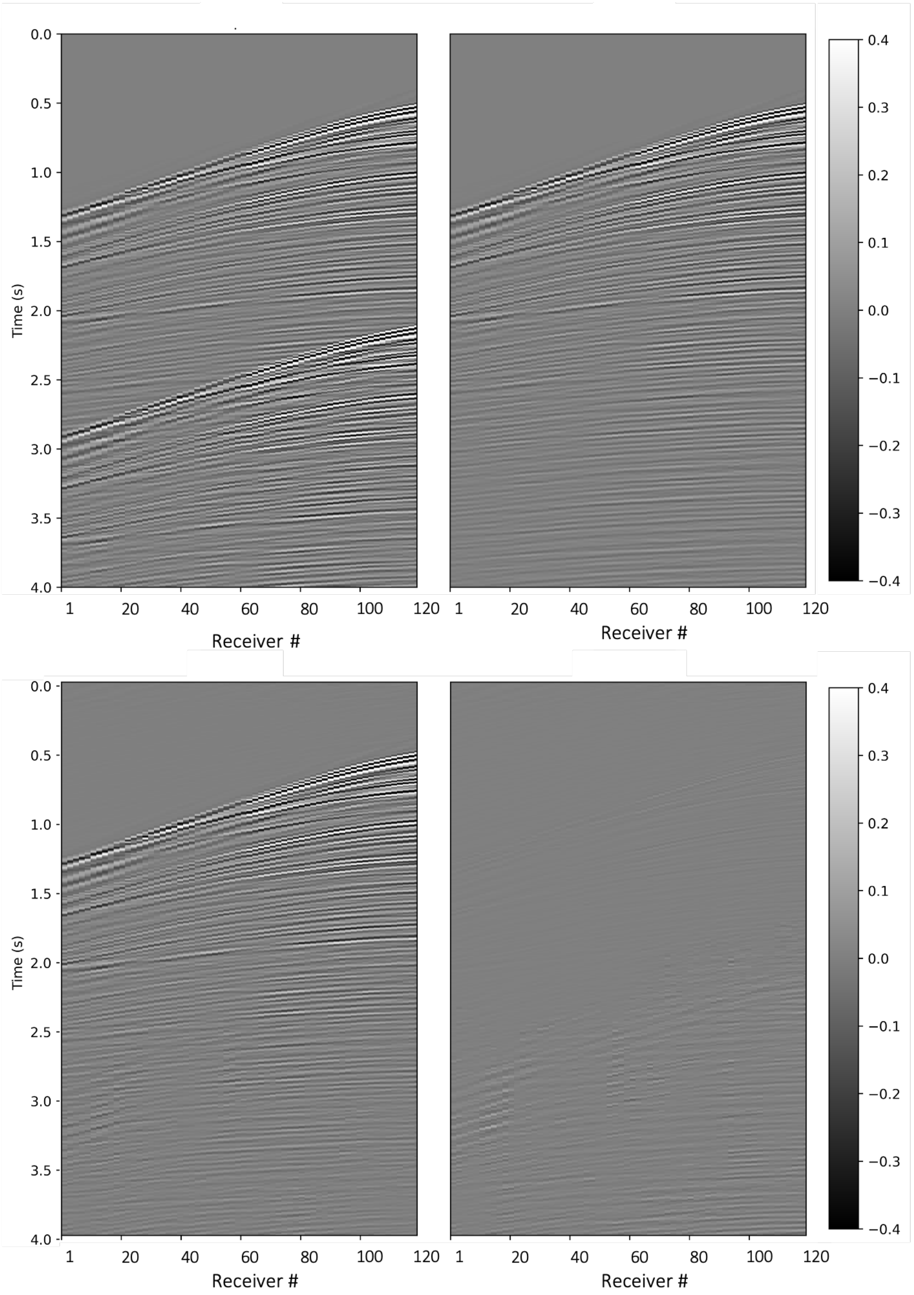


Figure 7.4 Blended field shot gather (input to the CNN) with a 1 s time-shift operator (upper left), unblended shot gather (desired output) (upper right), CNN prediction (lower left), and the difference between the desired output and CNN prediction (lower right).

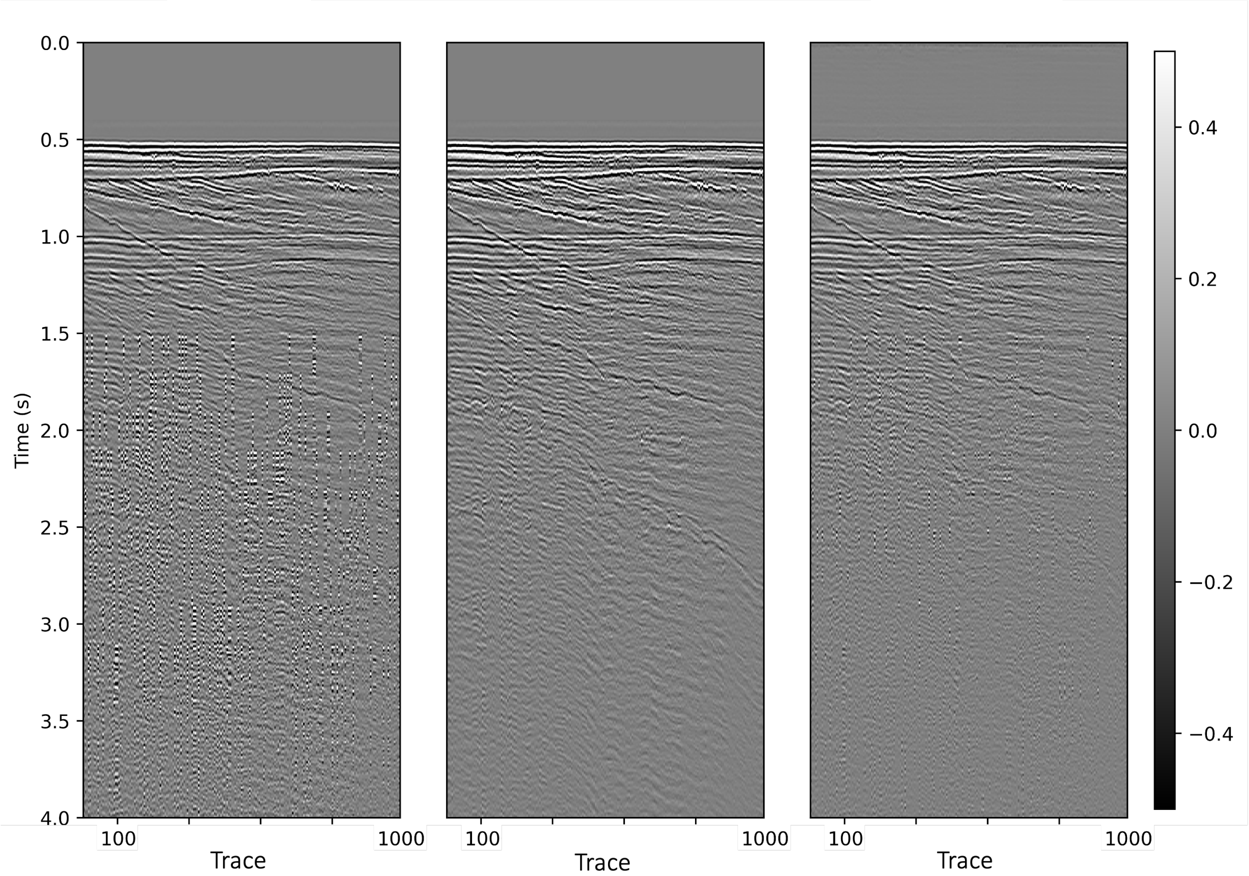


Figure 7.5 From left to right, traces corresponding to receiver number 120 from every blended field shot gather created with randomly chosen time-shifts between 1 s and 2.5 s, from the unblended shot gather (desired output), from the CNN prediction, respectively.

To illustrate, I blend the Mobil AVO Viking Graben data (Keys & Foster, 1998) and apply the CNN-based deblending. I create the blended data by blending a shot gather with itself and applying a time-shift operator. For the time-shift operator, I use randomly chosen time delays between 1 s and 2.5 s with a 0.5 s increment. As an example for the 1 s time-shift operator, Figure 7.4 shows the blended field shot gather (input to the CNN) (upper left), unblended shot gather (desired output) (upper right), the CNN prediction (lower left), and the difference between the desired output and CNN prediction (lower right). The difference waveform in Figure 7.4 (lower right) shows that although there is a small amplitude discrepancy at times later than 2.5 s, the effects of the blended shot gather have been greatly eliminated.

I extract the traces corresponding to receiver number 120 (also referred to as near-trace gather) from every blended field shot gather created with randomly chosen time-shifts between 1 s and 2.5 s, from the unblended shot gather (desired output), from the CNN prediction, and they are shown in Figure 7.5 from left to right, respectively. The random time shifts appear as incoherent noise between 1.5 s and 4 s on the left in Figure 7.5. The CNN prediction near-trace gather in Figure 7.5 (figure on the right) shows that the incoherent noise has been greatly suppressed and deblending has been applied.

For creating the blended data shown in Figure 7.4 and Figure 7.5, I blend a shot gather with itself. However, one can use different shot gathers to create the blended data and this approach can be generalized. Moreover, different time-shift operators can be included in the training and the CNN-based deblending can be further generalized for different data sets.

7.1.9 Incorporating Variations In Acquisition Into CNN

For obtaining the best prediction results in CNN-based solutions, training data need to contain parameters close to that of the prediction data. However, generalizing the CNN for variations in acquisitions such as source directionality, source signature, sea depth, sea-bottom topography, receiver spacing, frequency content, 2-D vs. 3-D data, and near-and far-offset receiver spacing would be of great use for seismic exploration problems. Seismic surveys typically have variations between the consecutive shots and generalizing the CNN-based algorithms for such variations address serious limitations and eliminate the need for training the CNN for specific problems.

7.1.10 Unboxing CNNs

The use of CNNs for seismic problems has become popular over the last decade. One of the advantages of utilizing CNN-based solutions for problems such as attenuating free-surface multiples is that they do not require an in-depth understanding of the CNNs. However, once we fully understand the “black box” behind the CNNs and how CNN tackles problems like free-surface multiple attenuation, it will help us utilize CNNs for other problems in exploration seismology as well. For example, the effects of different activation functions on the predictions can be investigated for free-surface multiple attenuation. Throughout this thesis, I use 1-D convolutional layers but a comparison of 2-D and 3-D convolutional layer predictions can be performed. Strategies for understanding how the CNN works need further investigation. This, however, is not a trivial question, but one approach might be to consider the state of the data at the intermediate levels in the network. We do not know, though, if that would actually be informative.

7.1.11 Addressing 3-D Free-Surface Multiples and Internal Multiples

I present a CNN-based approach for removing the free-surface multiples (1st- and 2nd-order, and peg-leg multiples) in 2-D in this thesis. One can also extend this approach to free-surface multiple removal in 3-D once the training in 3-D is done. Free-surface multiples in the in-line and cross-line directions need to be included in the network by creating input and output data pairs for training. Because our CNN-based approach can be used for irregular surveys, the requirements of the SRME method in 3-D can be circumvented. Moreover, the CNN-based approach can be generalized to incorporate the internal multiples. Similar to the removal of free-surface multiples, internal multiple removal workflows include (1) predicting the multiples, and (2) subtracting the multiples from the seismic data (Berkhout & Verschuur, 1997). The CNN-based approach I propose can be further generalized to mitigate the effects of primary removal when the primary and multiple reflections overlap during the subtraction step during the internal multiple elimination.

REFERENCES

Agranovich, Z. S., & Marchenko, V. A. 1963. *The inverse problem of scattering theory.* New York: Gordon and Breach.

Almuteri, K., & Sava, P. 2023. Seismic deghosting using convolutional neural networks. *Geophysics*, **88**(3), V113–V125.

Alzahrani, H., & Shragge, J. 2022. Seismic velocity model building using neural networks: Training data design and learning generalization. *Geophysics*, **87**(2), R193–R211.

Amundsen, L., & Zhou, H. 2013. Low-frequency seismic deghosting. *Geophysics*, **78**(2), WA15–WA20.

Araújo, F. V., Weglein, A. B., Carvalho, P. M., & Stolt, R. H. 1994. Inverse scattering series for multiple attenuation: An example with surface and internal multiples. *SEG Technical Program Expanded Abstracts*, 1039–1041.

Backus, M. M. 1959. Water reverberations - their nature and elimination. *Geophysics*, **24**(2), 233–261.

Baysal, E., Kosloff, D. D., & Sherwood, J. W. C. 1983. Reverse time migration. *Geophysics*, **48**(11), 1514–1524.

Beasley, C. J., Chambers, R. E., & Jiang, Z. 1998. A new look at simultaneous sources. *SEG Technical Program Expanded Abstracts*, 133–135.

Beasley, C. J., Moore, I., Monk, D., & Hansen, L. 2012. Simultaneous sources: The inaugural full-field, marine seismic case history. *SEG Technical Program Expanded Abstracts*, 1–5.

Behura, J., Wapenaar, K., & Snieder, R. 2014. Autofocus Imaging: Image reconstruction based on inverse scattering theory. *Geophysics*, **79**(3), A19–A26.

Berkhout, A. J. 2008. Changing the mindset in seismic data acquisition. *The Leading Edge*, **27**(7), 924–938.

Berkhout, A. J., & Verschuur, D. J. 1997. Estimation of multiple scattering by iterative inversion, Part I: Theoretical considerations. *Geophysics*, **62**(5), 1586–1595.

Bishop, C. M. 2006. Pattern recognition and machine learning. *Springer.*

Bleistein, N., & Cohen, J. K. 1977. Nonuniqueness in the inverse source problem in acoustics and electromagnetics. *J. Math. Phys.*, **18**, 194–201.

Brackenhoff, J., Thorbecke, J., & Wapenaar, K. 2019. Virtual sources and receivers in the real Earth: Considerations for practical applications. *Journal of Geophysical Research: Solid Earth*, **124**(11), 11802–11821.

Broggini, F., & Snieder, R. 2012. Connection of scattering principles: A visual and mathematical tour. *European Journal of Physics*, **33**(3), 593–613.

Burridge, R. 1980. The Gelfand-Levitan, the Marchenko, and the Gopinath-Sondhi integral equations of inverse scattering theory, regarded in the context of inverse impulse-response problems. *Wave Motion*, **2**(4), 305–323.

Campillo, M., & Paul, A. 2003. Long-range correlations in the diffuse seismic coda. *Science*, **299**(5606), 547–549.

Carlson, D. H., Long, A., Sollner, W., Tabti, H., Tenghamn, R., & Lunde, N. 2007. Increased resolution and penetration from a towed dual-sensor streamer. *First Break*, **25**, 71–77.

Chadan, K., & Sabatier, P. C. 1989. *Inverse problems in quantum scattering theory.* Springer.

Cheng, J., Liu, C., Zhou, L., Chen, W., & Gu, H. 2022. Deblending of simultaneous-source seismic data based on deep convolutional neural network. *IEEE Transactions on Geoscience and Remote Sensing*, **60**, 1–13.

Colton, D., & Kress, R. 1998. *Inverse acoustic and electromagnetic scattering theory.* Springer.

Curtis, A., Gerstoft, P., Sato, H., K., R. Snieder, & Wapenaar. 2006. Seismic interferometry - turning noise into signal. *The Leading Edge*, **25**, 1082–1092.

da Costa Filho, C. A., Ravasi, M., Curtis, A., & Meles, G. A. 2014. Elastodynamic Green's function retrieval through single-sided Marchenko inverse scattering. *Phys. Rev. E*, **90**, 063201.

da Costa Filho, C. A., Ravasi, M., & Curtis, A. 2015. Elastic P- and S-wave autofocus imaging with primaries and internal multiples. *Geophysics*, **80**(5), S187–S202.

Das, V., Pollack, A., Wollner, U., & Mukerji, T. 2019. Convolutional neural network for seismic impedance inversion. *Geophysics*, **84**(6), R869–R880.

Derode, A., Larose, E., Campillo, M., & Fink, M. 2003. How to estimate the Green's function for a heterogeneous medium between two passive sensors? Application to acoustic waves. *Appl. Phys. Lett.*, **83**, 3054–3056.

Devaney, A. J., & Sherman, G. C. 1982. Nonuniqueness in inverse source and scattering problems. *IEEE Trans. on Antennas and Propagation*, **AP-30**, 1034–1037.

Di, H., Truelove, L., Li, C., & Abubakar, A. 2020. Accelerating seismic fault and stratigraphy interpretation with deep CNNs: A case study of the Taranaki Basin, New Zealand. *The Leading Edge*, **39**(10), 727–733.

Diebold, J. B., & Stoffa, P. L. 1981. The traveltime equation, tau-p mapping, and inversion of common midpoint data. *Geophysics*, **46**(3), 238–254.

Diekmann, L., & Vasconcelos, I. 2021. Focusing and Green's function retrieval in three-dimensional inverse scattering revisited: A single-sided Marchenko integral for the full wave field. *Phys. Rev. Research*, **3**, 013206.

Dix, C. H. 1948. The existence of multiple reflections. *Geophysics*, **13**(1), 49–50.

Dondurur, D. 2018. Acquisition and processing of marine seismic data. *Elsevier.*

Dragoset, B., & MacKay, S. 1993. Surface multiple attenuation and sub-salt imaging. *Exploration Geophysics*, **24**(3-4), 463–472.

Dragoset, B., Verschuur, E., Moore, I., & Bisley, R. 2010. A perspective on 3D surface-related multiple elimination. *Geophysics*, **75**(5), 75A245–75A261.

Dragoset, W. H., & Jeričević, Z. 1998. Some remarks on surface multiple attenuation. *Geophysics*, **63**(2), 772–789.

Duroux, A., Sabra, K. G., Ayers, J., & Ruzzene, M. 2010. Using cross-correlations of elastic diffuse fields for attenuation tomography of structural damage. *J. Acoust. Soc. Am.*, **127**, 3311–3314.

Ekman, M. 2021. *Learning dDeep learning: Theory and practice of neural networks, computer vision, NLP, and transformers using TensorFlow.* Addison-Wesley Professional.

Ellsworth, T. P. 1948. Multiple reflections. *Geophysics*, **13**(1), 1–18.

Fomel, S., Sava, P., Vlad, I., Liu, Y., & Bashkardin, V. 2013. Madagascar: Open-source software project for multidimensional data analysis and reproducible computational experiments. *Journal of Open Research Software*, **1**(1), e8.

Gel'fand, I. M., & Levitan, B. M. 1955. On the determination of the differential equation from its spectral function. *Amer. Math. Soc. Transl.*, **1**, 253–304.

Géron, A. 2019. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow, 2nd Edition.* O'Reilly Media, Inc.

Gladwell, G. M. L. 1993. *Inverse problems in scattering.* Kluwer Academic Publishing.

Groenestijn, V., & Verschuur, A. J. 2011. Using surface multiples to estimate primaries by sparse inversion from blended data. *Geophysical Prospecting*, **59**(1), 10–23.

Guasch, L., Agudo, O. C., Tang, M., Nachev, P., & Warner, M. 2020. Full-waveform inversion imaging of the human brain. *npj Digital Medicine*, **28**(3).

Hampson, G., Stefani, J., & Herkenhoff, F. 2008. Acquisition using simultaneous sources. *The Leading Edge*, **27**(7), 918–923.

Higham, C. F., & Higham, D. J. 2019. Deep learning: An introduction for applied mathematicians. *SIAM Review*, **61**(4), 860–891.

Hu, Y., Zhen, R., & Sheikh, H. 2019. CNN-dased deghosting in high dynamic range imaging. *2019 IEEE International Conference on Image Processing (ICIP)*, 4360–4364.

Jakubowicz, H. 1998. Wave equation prediction and removal of interbed multiples. *SEG Technical Program Expanded Abstracts*, 1527–1530.

Jia, X., Guitton, A., & Snieder, R. 2018. A practical implementation of subsalt Marchenko imaging with a Gulf of Mexico data set. *Geophysics*, **83**(5), S409–S419.

Jia, X., Baumstein, A., Jing, C., Neumann, E., & Snieder, R. 2021. Subbasalt Marchenko imaging with offshore Brazil field data. *Geophysics*, **86**, WC31–WC40.

Kelamis, P. G., & Verschuur, D. J. 2000. Surface-related multiple elimination on land seismic data—Strategies via case studies. *Geophysics*, **65**(3), 719–734.

Keys, R. G., & Foster, D. J. 1998. *Comparison of seismic inversion methods on a single real data set: 1. A data set for evaluating and comparing seismic inversion methods.* Society of Exploration Geophysicists.

Kim, G. Y., Richardson, M. D., & Bibee, D. L. 2004. Sediment types determination using acoustic techniques in the Northeastern Gulf of Mexico. *Geosciences Journal*, **8**(1), 95–103.

Kiraz, M. S. R., & Nowack, R. L. 2018. Marchenko redatuming and imaging with application to the Frio carbon sequestration experiment. *Geophysical Journal International*, **215**(3), 1633–1643.

Kiraz, M. S. R., & Snieder, R. 2022. Marchenko focusing using convolutional neural networks. *Second International Meeting for Applied Geoscience & Energy*, 1930–1934.

Kiraz, M. S. R., Snieder, R., & Wapenaar, K. 2020. Marchenko focusing without up/down decomposition. *SEG Technical Program Expanded Abstracts*, 3593–3597.

Kiraz, M. S. R., Snieder, R., & Wapenaar, K. 2021. Focusing waves in an unknown medium without wavefield decomposition. *JASA Express Letters*, **1**(5), 055602.

Kiraz, M. S. R., Snieder, R., & Sheiman, J. 2023a. Removing free-surface effects from seismic data using convolutional neural networks – Part 1: Theory and sensitivity analysis. *submitted to Geophysical Prospecting.*

Kiraz, M. S. R., Snieder, R., & Sheiman, J. 2023b. Removing free-surface effects from seismic data using convolutional neural networks – Part 2: An application to the Mobil AVO Viking Graben data set. *submitted to Geophysical Prospecting.*

Kragh, E., Muyzert, E., Curtis, T., Svendsen, M., & Kapadia, D. 2010. Efficient broadband marine acquisition and processing for improved resolution and deep imaging. *The Leading Edge*, **29**(4), 464–469.

Li, J., Liu, Y., Tang, J., & Ma, F. 2023. Magnetotelluric noise suppression via convolutional neural network. *Geophysics*, **88**(1), WA361–WA375.

Liner, C. L. 2016. *Elements of 3D seismology, Third edition.* Society of Exploration Geophysicists.

Liu, Z., Chen, Y., & Schuster, G. 2020. Deep convolutional neural network and sparse least-squares migration. *Geophysics*, **85**(4), WA241–WA253.

Lomas, A., & Curtis, A. 2019. An introduction to Marchenko methods for imaging. *Geophysics*, **84**(2), F35–F45.

Lumley, D. E., Nichols, D., Ecker, C., Rekdal, T., & Berlioux, A. 1998. *Comparison of seismic inversion methods on a single real data set: 5. Amplitude-preserved processing and analysis of the Mobil AVO data set.* Society of Exploration Geophysicists.

Ma, C., Fu, Q., & Weglein, A. B. 2019. Comparison of the inverse scattering series free-surface multiple elimination (ISS FSME) algorithm with the industry-standard surface-related multiple elimination (SRME): Defining the circumstances in which each method is the appropriate toolbox choice. *Geophysics*, **84**(5), S459–S478.

Ma, J., Chen, X., Sen, M. K., & Xue, Y. 2016. Free-surface multiple attenuation for blended data. *Geophysics*, **81**(3), V227–V233.

Madiba, G. B., & McMechan, G. A. 2003. Processing, inversion, and interpretation of a 2D seismic data set from the North Viking Graben, North Sea. *Geophysics*, **68**(3), 837–848.

Marchenko, V. A. 1955. The construction of the potential energy from the phases of scattered waves. *Dokl. Akad. Nauk*, **104**, 695–698.

Mayne, W. H. 1962. Common reflection point horizontal data stacking techniques. *Geophysics*, **27**(6), 927–938.

McMechan, G. A. 1983. Migration by extrapolation of time-dependent boundary values. *Geophysical Prospecting*, **31**(3), 413–420.

Meles, G. A., Löer, K., Ravasi, M., Curtis, A., & da Costa Filho, C. A. 2015. Internal multiple prediction and removal using Marchenko autofocusing and seismic interferometry. *Geophysics*, **80**(1), A7–A11.

Meles, G. A., Wapenaar, K., & Curtis, A. 2016. Reconstructing the primary reflections in seismic data by Marchenko redatuming and convolutional interferometry. *Geophysics*, **81**(2), Q15–Q26.

Moldoveanu, N., Combee, L., Egan, M., Hampson, G., Sydora, L., & Abriel, W. 2007. Over/under towed-streamer acquisition: A method to extend seismic bandwidth to both higher and lower frequencies. *The Leading Edge*, **26**(1), 41–58.

Moseley, B., Markham, A., & Nissen-Meyer, T. 2020. Solving the wave equation with physics-informed deep learning. *arXiv:2006.11894*.

Murphy, K. P. 2012. *Machine learning: A probabilistic perspective*. Cambridge, MA: The MIT Press.

Müller, S., Niederleithinger, E., & Bohlen, T. 2012. Reverse time migration: A seismic imaging technique applied to synthetic ultrasonic data. *International Journal of Geophysics*.

Newton, R. G. 1980a. Inverse scattering. I. One dimension. *Journal of Mathematical Physics*, **21**(3), 493–505.

Newton, R. G. 1980b. Inverse scattering. II. Three dimensions. *Journal of Mathematical Physics*, **21**(7), 1698–1715.

Newton, R. G. 1981. Inverse scattering. III. Three dimensions, continued. *Journal of Mathematical Physics*, **22**(10), 2191–2200.

Newton, R. G. 1982. Inverse scattering. IV. Three dimensions: Generalized Marchenko construction with bound states, and generalized Gel'fand–Levitan equations. *Journal of Mathematical Physics*, **23**(4), 594–604.

Oristaglio, M. L. 1989. An inverse scattering formula that uses all the data. *Inverse Problems*, **5**(6), 1097–1105.

Ovcharenko, O., Baumstein, A., & Neumann, E. 2021. Surface-related multiple elimination through orthogonal encoding in the latent space of convolutional autoencoder. *First International Meeting for Applied Geoscience & Energy Expanded Abstracts*, 1355–1359.

Paffenholz, J., Stefani, J., McLain, B., & Bishop, K. 2002. Sigsbee_2A synthetic subsalt dataset - Image quality as function of migration algorithm and velocity model error. *European Association of Geoscientists & Engineers*, 2214–4609.

Raissi, M., Perdikaris, P., & Karniadakis, G. E. 2019. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, **378**, 686–707.

Ravasi, M., Vasconcelos, I., Kritski, A., Curtis, A., da Costa Filho, C. A., & Meles, G. A. 2016. Target-oriented Marchenko imaging of a North Sea field. *Geophysical Journal International*, **205**(1), 99–104.

Ricker, N. 1953. Wavelet contraction, wavelet expansion, and the control of seismic resolution. *Geophysics*, **18**(4), 769–792.

Riley, D., & Claerbout, J. 1976. 2-D multiple reflections. *Geophysics*, **41**(4), 1942.

Rose, J. H. 2001. "Single-sided" focusing of the time-dependent Schrödinger equation. *Physical Review A*, **65**, 012707.

Rose, J. H. 2002a. 'Single-sided' autofocusing of sound in layered materials. *Inverse Problems*, **18**(6), 1923–1934.

Rose, J. H. 2002b. *Time reversal, focusing and exact inverse scattering. In: Fink M., Kuperman W.A., Montagner J. P., Tourin A. (eds) Imaging of complex media with acoustic and seismic waves, topics in applied physics*. Springer, Berlin, Heidelberg.

Roux, P., Kuperman, W. A., & NPAL Group. 2004. Extracting coherent wave fronts from acoustic ambient noise in the ocean. *J. Acoust. Soc. Am.*, **116**, 1995–2003.

Sabra, K. G., Gerstoft, P., Roux, P., Kuperman, W. A., & Fehler, M. C. 2005. Extracting time-domain Green's function estimates from ambient seismic noise. *Geophys. Res. Lett.*, **32**, L03310.

Schuster, G. T. 2009. *Seismic interferometry.* Cambridge University Press.

Si, X., Yuan, Y., Si, T., & Gao, S. 2019. Attenuation of random noise using denoising convolutional neural networks. *Interpretation*, **7**(3), SE269–SE280.

Siahkoohi, A., Louboutin, M., Kumar, R., & Herrmann, F. J. 2018. Deep-convolutional neural networks in prestack seismic: Two exploratory examples. *SEG Technical Program Expanded Abstracts*, 2196–2200.

Siahkoohi, A., Verschuur, D J., & Herrmann, F. J. 2019. Surface-related multiple elimination with deep learning. *SEG Technical Program Expanded Abstracts*, 4629–4634.

Singh, S., & Snieder, R. 2017. Strategies for imaging with Marchenko-retrieved Green's functions. *Geophysics*, **82**(4), Q23–Q37.

Snieder, R., & Larose, E. 2013. Extracting Earth's elastic wave response from noise measurements. *Annual Review of Earth and Planetary Sciences*, **41**(1), 183–206.

Snieder, R., & van Wijk, K. 2015. *A guided tour of mathematical methods for the physical sciences.* 3 edn. Cambridge Univ. Press.

Song, C., Alkhalifah, T., & Waheed, U. B. 2021. Solving the frequency-domain acoustic VTI wave equation using physics-informed neural networks. *Geophysical Journal International*, **225**(2), 846–859.

Soubaras, R. 2010. Deghosting by joint deconvolution of a migration and a mirror migration. *SEG Technical Program Expanded Abstracts*, 3406–3410.

Soubaras, R., & Whiting, P. 2011. Variable depth streamer - The new broadband acquisition system. *SEG Technical Program Expanded Abstracts*, 4349–4353.

Staring, M., Dukalski, M., Belonosov, M., Baardman, R. H., Yoo, J., Hegge, R. F., van Borselen, R., & Wapenaar, K. 2021. Robust estimation of primaries by sparse inversion and Marchenko equation-based workflow for multiple suppression in the case of a shallow water layer and a complex overburden: A 2D case study in the Arabian Gulf. *Geophysics*, **86**(2), Q15–Q25.

Stoughton, D., Stefani, J., & Michell, S. 2001. 2D elastic model for wavefield investigations of subsalt objectives, deep water Gulf of Mexico. *SEG Technical Program Expanded Abstracts*, 1269–1272.

Sun, J., Slang, S., Elboth, T., Greiner, T. L., McDonald, S., & Gelius, L. 2020. A convolutional neural network approach to deblending seismic data. *Geophysics*, **85**(4), WA13–WA26.

Sun, J., Hou, S., Vinje, V., Poole, G., & Gelius, L. 2022. Deep learning-based shot-domain seismic deblending. *Geophysics*, **87**(3), V215–V226.

Tao, L., Jiang, J., Ren, H., Ye, Y., & Bang, B. Wu. 2021. The separation of seismic surface-related multiples based on CAE-SAGAN. *First International Meeting for Applied Geoscience & Energy Expanded Abstracts*, 2914–2918.

Thorbecke, J., Slob, E., Brackenhoff, J., van der Neut, J., & Wapenaar, K. 2017. Implementation of the Marchenko method. *Geophysics*, **82**(6), WB29–WB45.

Thorbecke, J., Zhang, L., Wapenaar, K., & Slob, E. 2021. Implementation of the Marchenko multiple elimination algorithm. *Geophysics*, **86**(2), F9–F23.

van der Neut, J., Wapenaar, K., Thorbecke, J., Slob, E., & Vasconcelos, I. 2015. An illustration of adaptive Marchenko imaging. *The Leading Edge*, **34**(7), 818–822.

Verschuur, D. J., & Berkhout, A. J. 2011. Seismic migration of blended shot records with surface-related multiple scattering. *Geophysics*, **76**(1), A7–A13.

Verschuur, D. J., Berkhout, A. J., & Wapenaar, C. P. A. 1992. Adaptive surface-related multiple elimination. *Geophysics*, **57**(9), 1166–1177.

Verschuur, D. J., Berkhout, A. J., & Kelmis, P. G. 1995. Estimation of multiple scattering by iterative inversion, Part II: Examples on marine and land data. *SEG Technical Program Expanded Abstracts*, 1470–1473.

Versteeg, R. 1994. The Marmousi experience: Velocity model determination on a synthetic complex data set. *The Leading Edge*, **13**(9), 927–936.

Virieux, J., & Operto, S. 2009. An overview of full-waveform inversion in exploration geophysics. *Geophysics*, **74**(6), WCC1–WCC26.

Vrolijk, J., & Blacquière, G. 2021. Source deghosting of coarsely sampled common-receiver data using a convolutional neural network. *Geophysics*, **86**(3), V185–V196.

Waheed, U., Haghighat, E., & Alkhalifah, T. 2020. Anisotropic eikonal solution using physics-informed neural networks. *SEG Technical Program Expanded Abstracts*, 1566–1570.

Wang, B., Li, J., & Han, D. 2022. Iterative deblending using MultiResUNet with multilevel blending noise for training and transfer learning. *Geophysics*, **87**(3), V205–V214.

Wapenaar, K. 2014. Single-sided Marchenko focusing of compressional and shear waves. *Phys. Rev. E*, **90**, 063202.

Wapenaar, K., Fokkema, J., & Snieder, R. 2005. Retrieving the Green's function in an open system by cross correlation: A comparison of approaches. *The Journal of the Acoustical Society of America*, **118**(5), 2783–2786.

Wapenaar, K., Broggini, F., & Snieder, R. 2012. Creating a virtual source inside a medium from reflection data: Heuristic derivation and stationary-phase analysis. *Geophysical Journal International*, **190**(2), 1020–1024.

Wapenaar, K., Broggini, F., Slob, E., & Snieder, R. 2013. Three-dimensional single-sided Marchenko inverse scattering, data-driven focusing, Green's function retrieval, and their mutual relations. *Phys. Rev. Lett.*, **110**, 084301.

Wapenaar, K., Thorbecke, J., van der Neut, J., Broggini, F., Slob, E., & Snieder, R. 2014. Marchenko imaging. *Geophysics*, **79**(3), WA39–WA57.

Wapenaar, K., Snieder, R., de Ridder, S., & Slob, E. 2021. Green's function representations for Marchenko imaging without up/down decomposition. *Geophysical Journal International*, **227**(1), 184–203.

Weaver, R. L., & Lobkis, O. I. 2001. Ultrasonics without a source: Thermal fluctuation correlations at MHz frequencies. *Phys. Rev. Lett.*, **87**, 134301.

Weglein, A. B., Gasparotto, F. A., Carvalho, P. M., & Stolt, R. H. 1997. An inverse-scattering series method for attenuating multiples in seismic reflection data. *Geophysics*, **62**(6), 1975–1989.

Wu, X., Liang, L., Shi, Y., & Fomel, S. 2019. FaultSeg3D: Using synthetic data sets to train an end-to-end convolutional neural network for 3D seismic fault segmentation. *Geophysics*, **84**(3), IM35–IM45.

Wu, Y., & McMechan, G. A. 2019. Parametric convolutional neural network-domain full-waveform inversion. *Geophysics*, **84**(6), R881–R896.

Xu, Y., Li, J., & Chen, X. 2019. Physics informed neural networks for velocity inversion. *SEG Technical Program Expanded Abstracts*, 2584–2588.

Yilmaz, O. 2001. *Seismic data analysis*. Society of Exploration Geophysicists.

Zhang, D., de Leeuw, M., & Verschuur, E. 2021. Deep learning-based seismic surface-related multiple adaptive subtraction with synthetic primary labels. *First International Meeting for Applied Geoscience & Energy Expanded Abstracts*, 2844–2848.

Zhang, G., Lin, C., & Chen, Y. 2020. Convolutional neural networks for microseismic waveform classification and arrival picking. *Geophysics*, **85**(4), WA227–WA240.

Zu, S., Cao, J., Qu, S., & Chen, Y. 2020. Iterative deblending for simultaneous source data using the deep neural network. *Geophysics*, **85**(2), V131–V141.

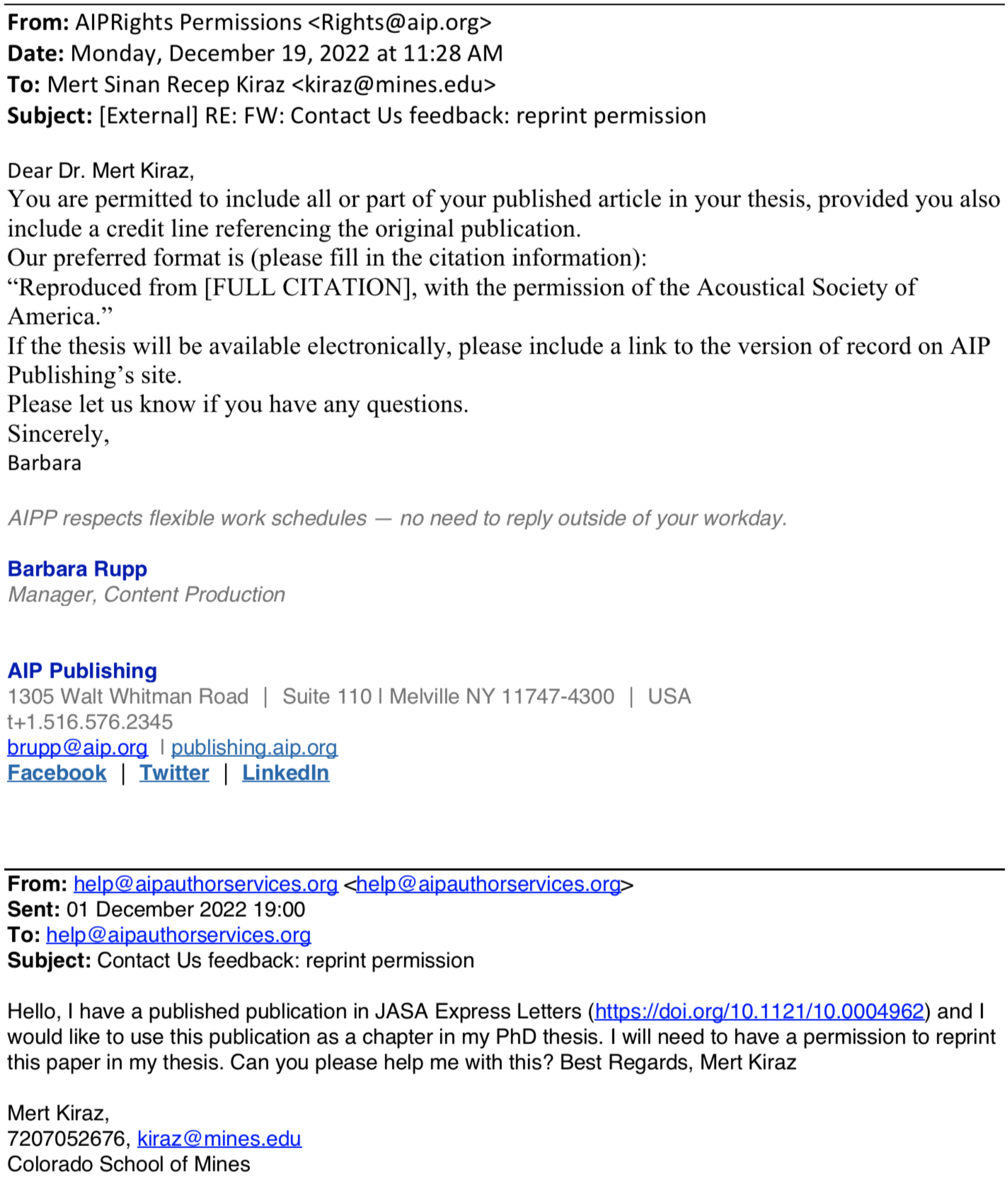
APPENDIX
PERMISSIONS FOR CHAPTERS

A.1 Permission From Journals

This section includes the copyright permissions from AIP Publishing representative in Figure A.1 for use of the published paper incorporated as a chapter in this thesis.

A.2 Permission From Co-Authors

This section includes copies of the emails exchanged with my co-authors, Dr. Kees Wapenaar in Figure A.2 and Dr. Jon Sheiman in Figure A.3, who granted permission to include the papers in the dissertation.



From: AIPRights Permissions <Rights@aip.org>
Date: Monday, December 19, 2022 at 11:28 AM
To: Mert Sinan Recep Kiraz <kiraz@mines.edu>
Subject: [External] RE: FW: Contact Us feedback: reprint permission

Dear Dr. Mert Kiraz,
You are permitted to include all or part of your published article in your thesis, provided you also include a credit line referencing the original publication.
Our preferred format is (please fill in the citation information):
“Reproduced from [FULL CITATION], with the permission of the Acoustical Society of America.”
If the thesis will be available electronically, please include a link to the version of record on AIP Publishing’s site.
Please let us know if you have any questions.
Sincerely,
Barbara

AIPP respects flexible work schedules — no need to reply outside of your workday.

Barbara Rupp
Manager, Content Production

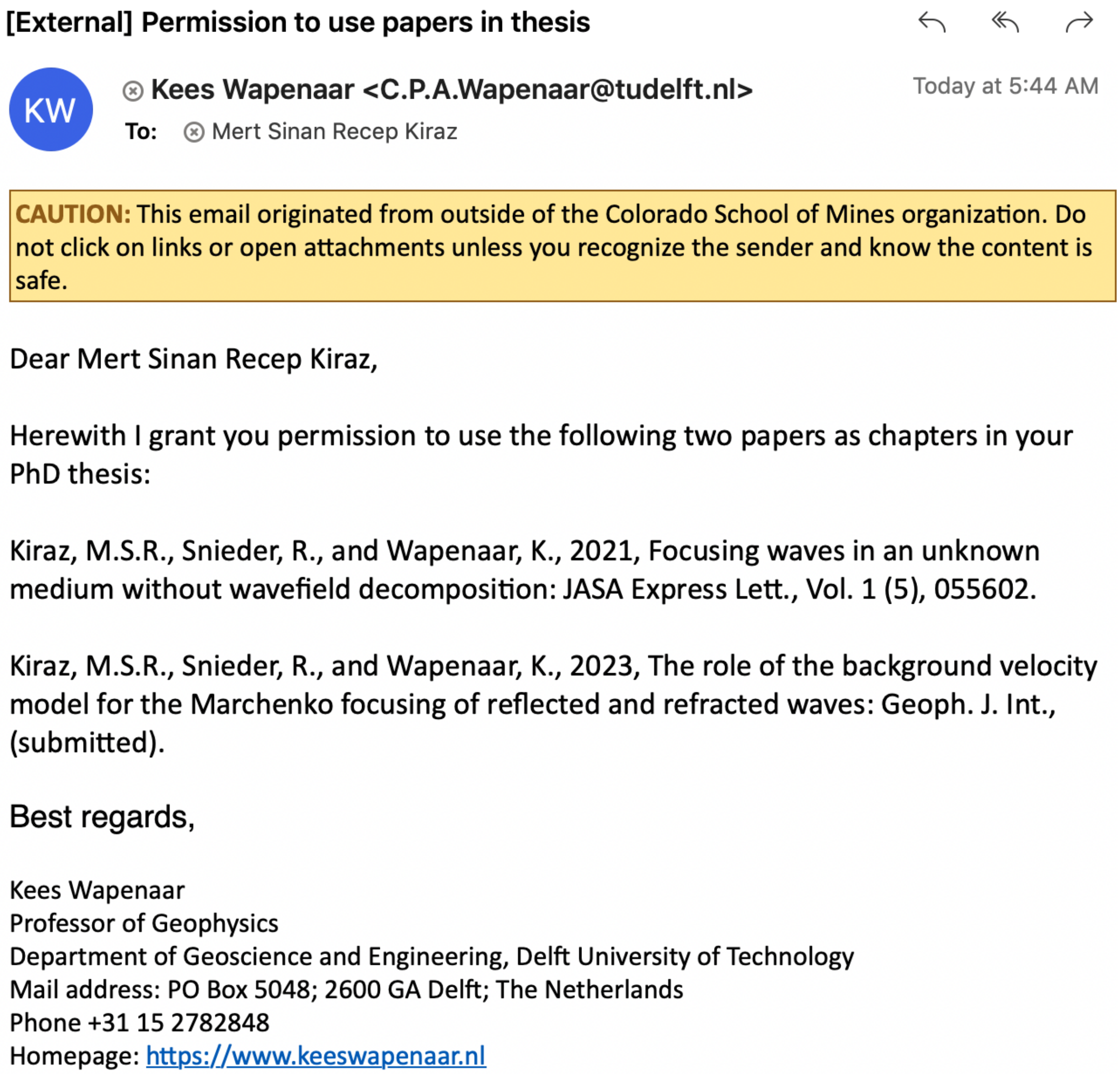
AIP Publishing
1305 Walt Whitman Road | Suite 110 I Melville NY 11747-4300 | USA
t+1.516.576.2345
brupp@aip.org I publishing.aip.org
Facebook | **Twitter** | **LinkedIn**

From: help@aipauthorservices.org <help@aipauthorservices.org>
Sent: 01 December 2022 19:00
To: help@aipauthorservices.org
Subject: Contact Us feedback: reprint permission

Hello, I have a published publication in JASA Express Letters (https://doi.org/10.1121/10.0004962) and I would like to use this publication as a chapter in my PhD thesis. I will need to have a permission to reprint this paper in my thesis. Can you please help me with this? Best Regards, Mert Kiraz

Mert Kiraz,
7207052676, kiraz@mines.edu
Colorado School of Mines

Figure A.1 Permission from AIP Publishing, publisher for JASA Express Letters.



[External] Permission to use papers in thesis

Kees Wapenaar <C.P.A.Wapenaar@tudelft.nl> Today at 5:44 AM

To: Mert Sinan Recep Kiraz

CAUTION: This email originated from outside of the Colorado School of Mines organization. Do not click on links or open attachments unless you recognize the sender and know the content is safe.

Dear Mert Sinan Recep Kiraz,

Herewith I grant you permission to use the following two papers as chapters in your PhD thesis:

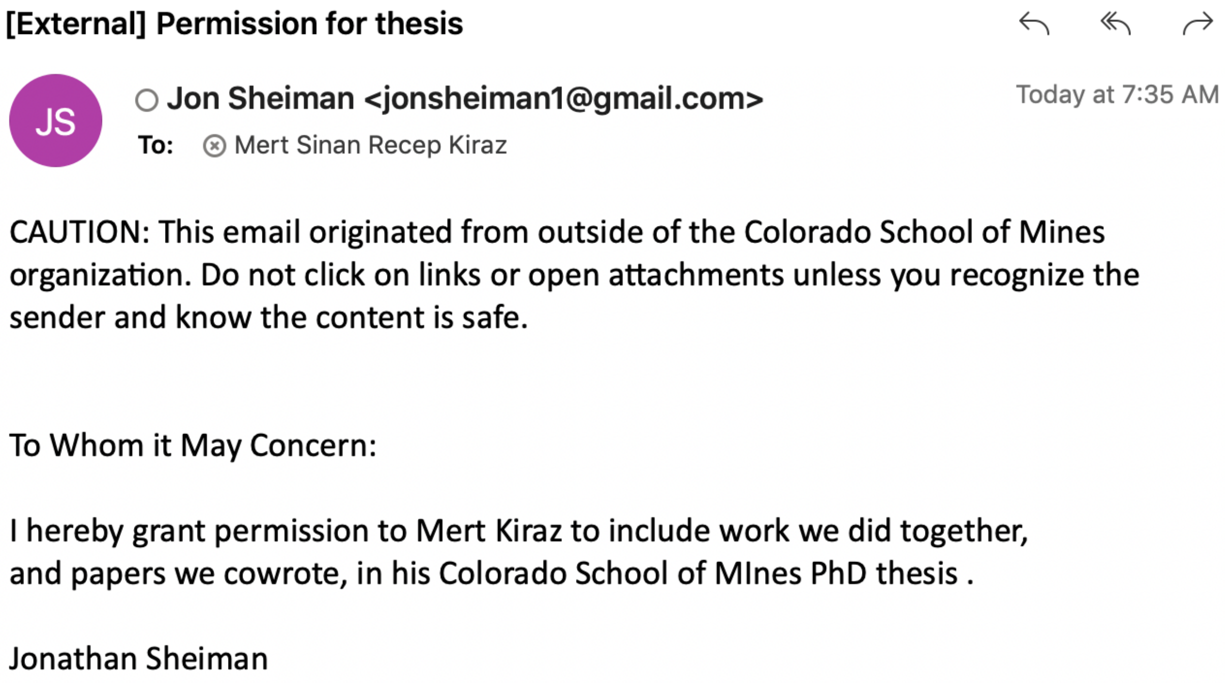
Kiraz, M.S.R., Snieder, R., and Wapenaar, K., 2021, Focusing waves in an unknown medium without wavefield decomposition: JASA Express Lett., Vol. 1 (5), 055602.

Kiraz, M.S.R., Snieder, R., and Wapenaar, K., 2023, The role of the background velocity model for the Marchenko focusing of reflected and refracted waves: Geoph. J. Int., (submitted).

Best regards,

Kees Wapenaar
Professor of Geophysics
Department of Geoscience and Engineering, Delft University of Technology
Mail address: PO Box 5048; 2600 GA Delft; The Netherlands
Phone +31 15 2782848
Homepage: https://www.keeswapenaar.nl

Figure A.2 Permission from Dr. Kees Wapenaar.



[External] Permission for thesis

○ Jon Sheiman <jonsheiman1@gmail.com>

Today at 7:35 AM

To: Mert Sinan Recep Kiraz

CAUTION: This email originated from outside of the Colorado School of Mines organization. Do not click on links or open attachments unless you recognize the sender and know the content is safe.

To Whom it May Concern:

I hereby grant permission to Mert Kiraz to include work we did together, and papers we cowrote, in his Colorado School of MInes PhD thesis .

Jonathan Sheiman

Figure A.3 Permission from Dr. Jon Sheiman.