LEARNING FROM SEISMIC DATA TO CHARACTERIZE SUBSURFACE VOLUMES

A Dissertation
Presented to
The Academic Faculty

By

Motaz Alfarraj

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Electrical and Computer Engineering

Georgia Institute of Technology

December 2019

LEARNING FROM SEISMIC DATA TO CHARACTERIZE SUBSURFACE VOLUMES

Approved by:

Dr. Ghassan AlRegib, Advisor
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. David Anderson
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. James H. McClellan
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Ying Zhang
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Zhigang Peng
School of Earth and Atmospheric Sciences
Georgia Institute of Technology

Date Approved: November 7, 2019

"Education is not something you can finish."

Isaac Asimov

I dedicate this dissertation to my wonderful family. To my loving parents Hailah and Abdulrahman, and to my wonderful wife Bayan.

Thank you for your unconditional love and support.

ACKNOWLEDGEMENTS

I owe my deepest gratitude to Prof. Ghassan Alregib, my Ph.D. advisor, my mentor, and my friend for his unparalleled guidance and support before and throughout my Ph.D. journey. With the supervision of Prof. AlRegib, I had the opportunity to explore and work on various research topics, and develop the needed skills to excel in research. Every discussion I had with Prof. AlRegib had a lesson in it, and for that, I am indebted to him.

I am also grateful to my Ph.D. thesis committee members: Prof. James McClellan, Prof. Zhigang Peng, Prof. David Anderson, and Prof. Ying Zhang for their invaluable comments and discussion to improve my Ph.D. work.

I am especially grateful to King Fahd University of Petroleum and Minerals (KFUPM) for generously providing me a scholarship to obtain my Ph.D. I am also thankful to the Center for Energy and Geo Processing (CeGP) at Georgia Tech and KFUPM for generously supporting my research.

I would also like to extend my thanks to all my friends and colleagues in the Omni Lab for Intelligent Visual Engineering and Science (OLIVES), and the Center for Energy and Geo Processing (CeGP). I was incredibly fortunate to have had the opportunity to work with such an amazing group of excellent researchers and loving friends. I am especially thankful to Dr. Can Temel, Charlie Lehman, Mohit Prabhushankar, Chih-Yao Ma, Min-Hung (Steve) Chen, Gukyeong Kwon, Jinsol Lee, Ahmad Mustafa, Joseph Aribiodo, Moamen Soliman, Chen Zhou, Yash-yee Logan, Yutong Sun, Benoit Marteau. I would also like to extend my deep gratitude to the former members of OLIVES and CeGP: Dr. Yazeed Alaudah, Dr. Mohammad Aabed, Dr. Zhiling long, Dr. Ashraf Alattar, Dr. Tariq Alshawi, Dr. Zhen Wang, Dr. Yuting Hu, and Dr. Mumhammad Shafiq. You helped me establish my research through your guidance and teaching.

I would also like to extend my thanks to the staff in the school of Electrical and Computer Engineering at Georgia Tech who made sure I had the best possible work environment to make progress in my research. I am especially thankful to Raquel Plaskett, Dr. Daniela Staiculescu, Christopher Thomas, and Pat Dixon.

Special thanks to my very dear friends: Abdulhakim Alhowaish, Maad Alowaifeer, and Andrew Kerr for always being there for me. I would also extend my appreciation and thanks to all my friends at Georgia Tech who made me feel home away from home. Special thanks to Mohannad Alkhuraijah, Dr. Yazeed Alaudah, Mohammad Alhassoun, Dr. Mohammad Nabhan, Dr. Said Alabri, and Dr. Abdullah Alshehri.

I owe my deepest gratitude to my family. I can not thank you enough for your unconditional love and support. To my father Abdulrahman, my mother Hailah, my siblings Zeyad, Lamees, Reenad, Motasim, and Moayyad, thank you for everything, I am forever in your debt. Finally, I am deeply indebted to my beloved wife Bayan for her love, support, companionship and for her sacrifices that made my Ph.D. journey possible.

TABLE OF CONTENTS

Acknowledgments v

List of Tables xi

List of Figures xiii

Chapter 1: Introduction 1

1.1 Motivation 1

1.2 Oil and Gas Exploration and Production Pipeline 3

1.3 Subsurface Volume Characterization 8

1.4 Outline 15

Chapter 2: Multiresolution Texture Analysis for Structure Characterization 17

2.1 Overview 17

2.2 Background 18

2.2.1 Content-based Image Retrieval 18

2.2.2 Image Similarity 20

2.2.3 Multiresolution Image Decomposition 24

2.3 Multiresolution Texture Similarity Measure 28

2.3.1 Methodology 29

2.3.2 Performance Evaluation 31
2.3.3 Results 35
2.4 Similarity-based Retrieval of Subsurface Structures 39
2.4.1 Experimental Procedure 39
2.4.2 Results and Discussion 40
2.5 Structure Characterization of Subsurface Volumes 46
2.5.1 Experimental Procedure 46
2.5.2 Results and Discussion 47
2.6 Summary 49

Chapter 3: Sequence Modeling for Lithology Characterization 51

3.1 Overview 51
3.2 Background 52
3.2.1 Lithology Characterization 52
3.2.2 Sequence Modeling 55
3.3 Proposed Method 60
3.4 Case Study: Density Estimation 61
3.4.1 Data Preprocessing 62
3.4.2 Training the Network 62
3.4.3 Results and Discussion 63
3.5 Case Study: Acoustic Impedance Estimation 65
3.5.1 Results and Discussion 67
3.6 Summary 71

Chapter 4: Semi-supervised Lithology Characterization with a Predefined Forward Model 72

4.1 Overview 72

4.2 Background 73

4.2.1 Seismic Modeling 73

4.2.2 Linear Seismic Inversion 78

4.3 Proposed Framework 82

4.3.1 The Inversion Network 85

4.3.2 The Forward Model 89

4.4 Case Study: Acoustic Impedance Estimation 90

4.4.1 Training the Network 91

4.4.2 Results 92

4.5 Case Study: Elastic Impedance Estimation 99

4.5.1 Results 99

4.6 Summary 101

Chapter 5: Semi-supervised Lithology Characterization with a Learned Forward Model 109

5.1 Overview 109

5.2 Background 110

5.3 Proposed Framework 111

5.3.1 Learning the Forward Model 114

5.4 Case Study: Acoustic Impedance Estimation 115

5.5 Case Study: Elastic Impedance Estimation 126

5.6 Summary 127

Chapter 6: Conclusion 137

6.1 Contributions 138

6.2 Future Research Directions 140

Chapter 7: Thesis Products 142

7.1 Magazines Articles 142

7.2 Journal Articles 142

7.3 Conference Papers 143

7.4 Datasets 144

7.5 Software 144

Appendix A: Evaluation Metrics 146

A.1 Image Retrieval 146

A.2 Clustering 148

A.3 Segmentation 150

A.4 Goodness-of-fit 151

References 164

Vita 165

LIST OF TABLES

2.1 Summary of the texture datasets used in the retrieval experiments. . 33

2.2 Performance evaluation of texture similarity measures on CUReT [67], PerTex [68], and Forrest [69] datasets. 38

2.3 Quantitative evaluation of the proposed similarity measure on LANDMASS-2 dataset. 41

2.4 Evaluation of the quality of the clusters obtained by k-means clustering. 45

2.5 Quantitative evaluation of different multiresolution decomposition methods for subsurface structure characterization. 48

3.1 Quantitative Evaluation of the estimated density traces using the proposed network. 63

3.2 Quantitative evaluation of the estimated acoustic impedance from noise-free synthetic seismic using recurrent and feed-forward networks. . . . 70

3.3 Quantitative evaluation of the estimated acoustic impedance from wave-equation-migrated seismic using recurrent and feed-forward networks. 70

4.1 Quantitative evaluation of the estimated acoustic impedance from synthetic seismic data with a 15dB noise. 93

4.2 Quantitative evaluation of the estimated elastic impedance from multi-angle synthetic seismic data with a 15dB noise. 107

5.1 Quantitative evaluation of the estimated acoustic impedance from synthetic seismic data with a 15dB noise. 117

5.2 Quantitative evaluation of the estimated acoustic impedance from migrated seismic data. 122

5.3 Quantitative evaluation of the estimated elastic impedance from synthetic multi-angle seismic data with a 15dB noise. 133

LIST OF FIGURES

1.1 P-impedance logs from wells F03-2 and F03-4 overlaid on a seismic section. Data is from the Netherlands F3 block [5]. 3

1.2 Oil and gas exploration and production pipeline. 4

1.3 An illustration of an offshore seismic survey (adapted from [6] with permission) 5

1.4 Well-logs of well F03-4 from the Netherlands F3 block [5]. 6

1.5 A migrated seismic volume with density logs from the Netherlands F3 block [5]. 7

1.6 Examples of the three types of subsurface characterization. 8

1.7 A seismic section and its corresponding attributes. Data shown for inline 301 from Netherlands offshore F3 block. 11

1.8 Synthetic seismogram generated from well-log data and its corresponding recorded seismic trace at the well location. 12

1.9 A visual illustration of the developed framework in this dissertation. . 14

2.1 The objective of the introduced structure characterization framework. 18

2.2 A general content-based image retrieval workflow. 19

2.3 An illustration of the similarity between seismic images and texture images. 20

2.4 Two images with similar texture and their absolute difference. 21

2.5 A generic block diagram of a texture similarity measure. 22

2.6 Part of a salt dome from inline 299 of the Netherlands offshore F3 block [5] 25

2.7 4-scale Gaussian pyramid workflow. 26

2.8 An illustration of the discrete wavelet transform. 27

2.9 The image in Figure 2.6 filtered with Gabor filters at three scales and four orientations. The filters are shown at the bottom left corners . . 28

2.10 An illustration of the curvelet transform of a seismic image. Orientations and scales are represented by numbers and colors, respectively. . 29

2.11 The proposed multiresolution texture similarity measure. 30

2.12 Sample images from CURet [67], PerTex [68], and Forrest [69] datasets. 34

2.13 Receiver Operating Characteristic curves (ROC) for different similarity measures on three texture datasets. 37

2.14 Sample images from LANDMASS-2 [72] depicting the four subsurface structures in the dataset. 39

2.15 Receiver Operating Characteristic curves (ROC) for different similarity measures on LANDMASS-2 [72] dataset. 42

2.16 Precision at n curves for different subsurface structure classes. 42

2.17 Three-dimensional representation of LANDMASS-2 [72] dataset using MDS based on similarity values. 44

2.18 Structure labeling workflow using a few exemplar images for each of the structures of interest. 46

2.19 Results of labeling inline 381 from the Netherlands F3 block using different multiresolution decomposition methods. 50

3.1 Examples of states transition diagrams for different orders of HMM (n). The numbers on the arrows represent the transition probability. . 57

3.2 An illustration of feed-forward and recurrent networks. 58

3.3 Gated Recurrent Unit (GRU) unfolded through time. 59

3.4 The architecture of the inversion network in the proposed framework. 60

3.5 A top view of the seismic survey area showing the locations of the wells used in the density estimation experiment. 61

3.6 Cross-validation results for density estimation from migrated seismic data. Training traces are shown in blue, validations traces are shown in red, and actual measurements are shown in black. 64

3.7 A Synthetic seismic section, a wave-equation-migrated seismic section from Marmousi 2 [23], and the true acoustic impedance section (in depth). 66

3.8 Estimated acoustic impedance from noise-free synthetic seismic data using a convolutional network and a recurrent network. 68

3.9 Estimated acoustic impedance from wave-equation-migrated seismic data using a convolutional network and a recurrent network. 69

4.1 Seismograms as a convolution of the reflectivity series with a wavelet. 75

4.2 An illustration of the reflected and refracted components of a plane wave at an interface. 76

4.3 The proposed semi-supervised lithology characterization framework with a predefined forward model. 84

4.4 The architecture of the inversion network. GRU: Gated Recurrent Unit, Conv: convolutional layer + group normalization + non-linearity, TConv: transposed convolutional layer + group normalization + non-linearity, Linear: fully connected layer. 86

4.5 An illustration of dilated convolution for multiscale feature extraction (kernel size = 5, dilation factors = $1, 2$, and 3). 87

4.6 The forward models used to synthesize seismograms from impedance. 90

4.7 A Synthetic seismic section (in time) from Marmousi 2 [23] and its corresponding acoustic impedance section. 91

4.8 Estimated acoustic impedance from a synthetic seismic section with 15dB noise. 96

4.9 Scatter plots of the true acoustic impedance versus the estimated acoustic impedance. The shaded region include all points that are within one standard deviation of the true acoustic impedance. The black line is the best linear fit. 97

4.10 Estimated acoustic impedance from a migrated seismic section. 98

4.11 Synthetic multi-angle seismic sections (left), and their corresponding true elastic impedance sections. 100

4.12 Estimated elastic impedance ($\theta = 0°$) from synthetic multi-angle seismic data. 103

4.13 Estimated elastic impedance ($\theta = 10°$) from synthetic multi-angle seismic data. 104

4.14 Estimated elastic impedance ($\theta = 20°$) from synthetic multi-angle seismic data. 105

4.15 Estimated elastic impedance ($\theta = 30°$) from synthetic multi-angle seismic data. 106

4.16 Scatter plots of the true elastic impedance versus the estimated elastic impedance. The shaded region include all points that are within one standard deviation of the true acoustic impedance. The black line is the best linear fit. 108

5.1 An illustration of cycle inconsistency loss for inverse problems. 111

5.2 The proposed semi-supervised lithology characterization framework with a learned forward model. 112

5.3 The architecture of the forward network in the semi-supervised lithology characterization framework. 115

5.4 Estimated acoustic impedance from a synthetic seismic section with a 15dB noise. 118

5.5 The distribution of the **PCC** and $\mathbf{r}^2$ for the estimated acoustic impedance from synthetic seismic data. 119

5.6 Scatter plots of the true acoustic impedance versus the estimated acoustic impedance. The shaded region include all points that are within one standard deviation of the true acoustic impedance. The black line is the best linear fit. 120

5.7 Selected traces from the estimated acoustic impedance section. 121

5.8 Estimated acoustic impedance from a migrated seismic section. 123

5.9 The distribution of the **PCC** and $\mathbf{r}^2$ for the estimated acoustic impedance from migrated seismic data. 124

5.10 Selected traces from the acoustic impedance estimated from section from wave-equation-migrated seismic data. 125

5.11 Estimated elastic impedance ($\theta = 0°$) from synthetic seismic data with a 15dB noise. 129

5.12 Estimated elastic impedance ($\theta = 10°$) from synthetic seismic data with a 15dB noise. 130

5.13 Estimated elastic impedance ($\theta = 20°$) from synthetic seismic data with a 15dB noise. 131

5.14 Estimated elastic impedance ($\theta = 30°$) from synthetic seismic data with a 15dB noise. 132

5.15 The distribution of **PCC** and $\mathbf{r}^2$ values for the estimated elastic impedance from synthetic multi-angle seismic data. 134

5.16 Scatter plots of the true elastic impedance versus the estimated elastic impedance. The shaded region include all points that are within one standard deviation of the true acoustic impedance. The black line is the best linear fit. 135

5.17 Selected traces from the estimated elastic impedance sections for all incident angles. 136

SUMMARY

The exponential growth of collected data from seismic surveys makes it impossible for interpreters to manually inspect, analyze and annotate all collected data. Deep learning has proved to be a potential mechanism to overcome big data problems in various computer vision tasks such as image classification and semantic segmentation. However, the applications of deep learning are limited in the field of subsurface volume characterization due to the limited availability of consistently-annotated seismic datasets. Obtaining annotations of seismic data is a labor-intensive process that requires field knowledge. Moreover, seismic interpreters rely on the few direct high-resolution measurements of the subsurface from well-logs and core data to confirm their interpretations. Different interpreters might arrive at different valid interpretations of the subsurface, all of which are in agreement with well-logs and core data. Therefore, to successfully utilize deep learning for subsurface characterization, one must address and circumvent the lack or shortage of consistent annotated data. In this dissertation, we introduce a learning-based physics-guided subsurface volume characterization framework that can learn from limited inconsistently-annotated data. The introduced framework integrates seismic data and the limited well-log data to characterize the subsurface at a higher-than-seismic resolution. The introduced framework takes into account the physics that governs seismic data to overcome noise and artifacts that are often present in the data. Integrating a physical model in deep-learning frameworks improves their generalization ability beyond the training data. Furthermore, the physical model enables deep networks to learn from unlabeled data, in addition to a few annotated examples, in a semi-supervised learning scheme. Applications of the introduced framework are not limited to subsurface volume characterization, it can be extended to other domains in which data represent a physical phenomenon and annotated data is limited.

CHAPTER 1
INTRODUCTION

1.1 Motivation

Technological Advances enable us to see beyond what is present before our eyes. In various fields like astronomy, medical imaging, material sciences, and seismic imaging, sensors are often used to measure physical properties, which are then turned into visual data. For instance, in Magnetic Resonance Imaging (MRI), magnetic fields are translated into images that enable visualizing body organs in a non-invasive procedure. In oil exploration and production (E&P), the reflections of seismic waves from the different layers of the subsurface are translated into a detailed 3-dimensional images of the subsurface. These images are then used to infer geological phenomena of the subsurface. We refer to the process of inferring physical properties and phenomena from indirect measurements as *volume characterization*.

Recent advances in data acquisition technology have resulted in the collection of massive amounts of data. Big data, paired with recent advances in computing technology, make it possible to visually represent physical phenomena. In the field of oil and gas exploration and production, massive amounts of data from the subsurface are collected every day in seismic surveys that span tens or hundreds of square kilometers. Moreover, the amounts of data grow exponentially with advances in seismic acquisition technology. For instance, CGG, a geophysical firm, reported that the number of sensors per square kilometer in their seismic surveys has grown from 400,000 in 2005 to 36,000,000 in 2009 [1]. This growth in the number of sensors increased the average data collected in an 8-hour shift from 100 gigabytes to more than 2 terabytes [1].

The collected seismic data is then processed and migrated into images of the sub-

surface. Images are then used to interpret the different structures, and estimate rock properties of the subsurface in a step we refer to as *subsurface volume characterization*. Characterizing the subsurface helps field engineers plan wells to target areas where oil reservoirs are highly likely. subsurface volume characterization is a labor-intensive and time-consuming process that can take a team of geoscientists up to several months to complete. With the exponential growth of collected data from different seismic surveys, it is impossible for geoscientists to manually inspect, analyze, and interpret all collected data. Moreover, seismic data are indirect low-resolution measurements of the subsurface that are often corrupted with noise and imaging artifacts. To overcome this limitation, geoscientists improve their existing characterization of seismic volumes by integrating other types of data such as well-logs and core data which are high-resolution and direct measurements of the subsurface. Well-logs are 1-dimensional measurements of the rock properties such as porosity and density at well locations. While seismic data are limited in resolution by their dominant wavelet (vertical resolution $\sim$ 10m-60m) [2], well-logs offer much higher resolution measurements of the subsurface with a vertical resolution up to a few inches depending on the logging equipment [3]. However, well-logs and core data are very sparse in the survey area due to the high cost of drilling wells. U.S. Energy Information Administration (UIE) reported an average cost of around $6,500,000 per well in Texas and New Mexico in 2015[4]. Figure 1.1 shows p-impedance logs from wells F03-2 and F03-4 overlaid on seismic data to highlight the higher resolution of well-log data compared to seismic.

Recent advances in deep learning have allowed the analysis of large amounts of data, that would usually require thousands of man-hours, in a matter of minutes, hours, or days. Deep learning enables artificial neural networks to learn various tasks and achieve better-than-human results. They have been a driving force in many of the advances in science and engineering fields. However, deep neural networks mostly

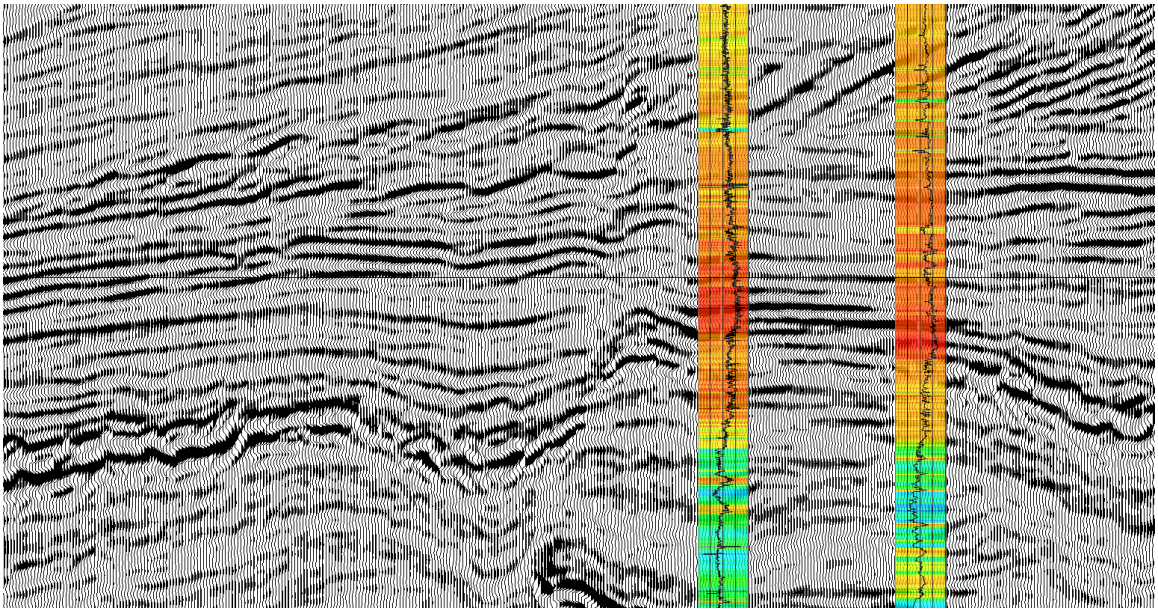


Figure 1.1: P-impedance logs from wells F03-2 and F03-4 overlaid on a seismic section. Data is from the Netherlands F3 block [5].

learn from examples, labeled data, to master certain tasks. This presents a major limitation of deep learning in fields where labeled data is not vastly available and is not easy to obtain such as the oil and gas E&P. Moreover, although seismic data is a representation of a physical phenomenon (wave reflection from the subsurface), the physics that governs seismic data is rarely integrated into deep-learning frameworks. Instead, deep neural networks are expected to learn the physical characteristics of the data in a data-driven manner. If the training data is noisy or is of a limited resolution, as in the case of seismic data, deep neural networks might fail to learn an adequate representation of the data.

1.2 Oil and Gas Exploration and Production Pipeline

In this section, we will introduce the main stages in the oil and gas exploration and production pipeline. We divide this pipeline into seven main stages that are shown in Figure 1.2.

The first stage in this pipeline is the *geological study* of a region of interest. In

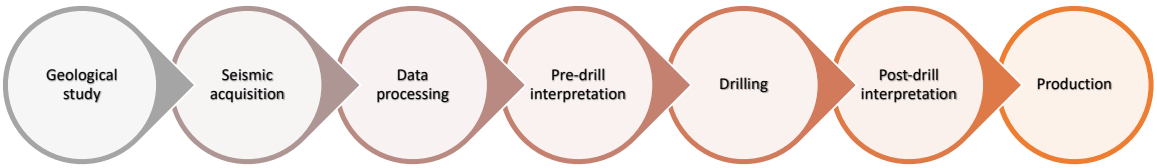


Figure 1.2: Oil and gas exploration and production pipeline.

this step, geologists review the full history of the region by going over maps and photographs, or by field trips to examine the local geology of the region. Information gathered from the geological study helps field experts determine the possibility of a petroleum system in the area.

The second step is *seismic acquisition* through large seismic surveys. In seismic surveys, a large 2-dimensional mesh of sensors is used to record the reflections of seismic waves generated by a seismic source (vibration source) from the subsurface. In onshore surveys, the sensors (geophones) are laid on land. An illustration of an onshore seismic survey is shown in Figure 1.3. In offshore surveys, hydrophones are towed by a seismic vessel on the surface of the water. Then, an energy source (seismic source) is used to generate seismic waves that travel through the subsurface. The reflections of those waves from the different layers of the subsurface are collected at the surface. Data collected by the sensors is what is known as *raw seismic* data.

The raw seismic data goes through the *data processing* stage of the pipeline [2]. It consists of several steps including data preprocessing, deconvolution, normal move-out correction, common-midpoint (CMP) sorting and stacking, velocity analysis, and migration. Data processing improves the quality of the seismic data, enhances its vertical resolution, align the events in the data with their true location in the subsurface. In this step, the data is converted to a field-specific data format such as SEG-Y [7]. The results of this step are migrated seismic volumes whose z-axis can be time or depth depending on the type of migration.

The next stage is the *pre-drill interpretation* which is the first part of the *subsurface volume characterization*. In pre-drill interpretation, the major events and

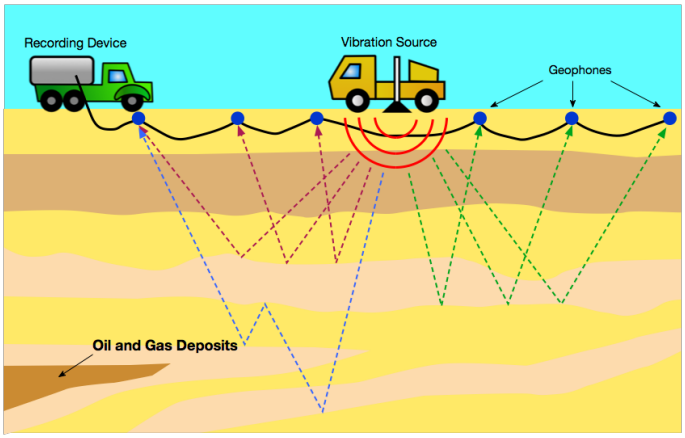


Figure 1.3: An illustration of an offshore seismic survey (adapted from [6] with permission)

structures of the subsurface volume are identified by interpreters. This step is critical to identify potential oil traps such as salt bodies and faults. It is also important to identify drilling hazards such as unconsolidated formations and over-pressured zones. The structurally interpreted volume is then used to determine the optimal drilling locations for the next stage.

The next stage of the pipeline is *drilling* in which wells are drilled to assess the true geology of the subsurface. While drilling, rock property measurements such as electrical, acoustic, electromagnetic measurements are recorded through sensors that are lowered into the borehole or well. Figure 1.4 shows an example of a few logs, including density, sonic, Gamma rays, porosity, p-impedance, and P-wave velocity, from well F03-4 of the Netherlands F3 block [5]. Rock samples are also collected from the borehole through the coring process to obtain core data. Drilling wells is a costly process with the cost varying depending on the type of well, its depth and the local geology. However, they reveal true properties of the subsurface at a higher resolution compared to seismic data which is a low-resolution reflection data. The

number of well-logs is limited in seismic survey whereas seismic data is abundantly available throughout the survey. Figure 1.5 shows a time-migrated seismic volume along with three wells from the Netherlands offshore F3 block [5].

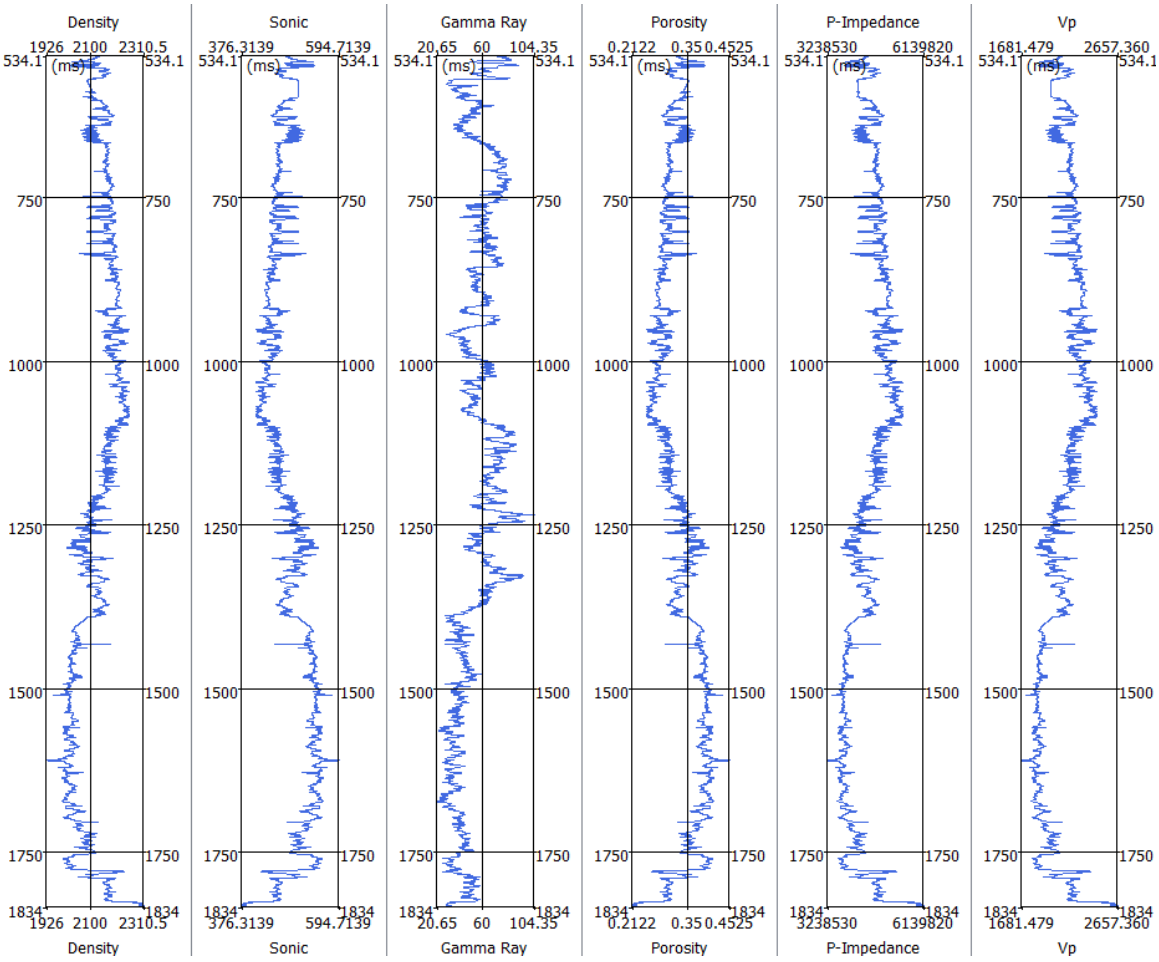


Figure 1.4: Well-logs of well F03-4 from the Netherlands F3 block [5].

Well-log data and seismic data are then used for a more detailed interpretation of the subsurface in a step we refer to as *post-drill interpretation*. At this stage, knowledge about the properties of the subsurface from well-logs and core data is extended beyond well locations guided by the seismic data. We refer to this type of characterization as lithology characterization. Also, well-logs and seismic data are integrated for stratigraphy characterization of the subsurface to identify the different depositional and sedimentary environments in the area. Integration of well-log data

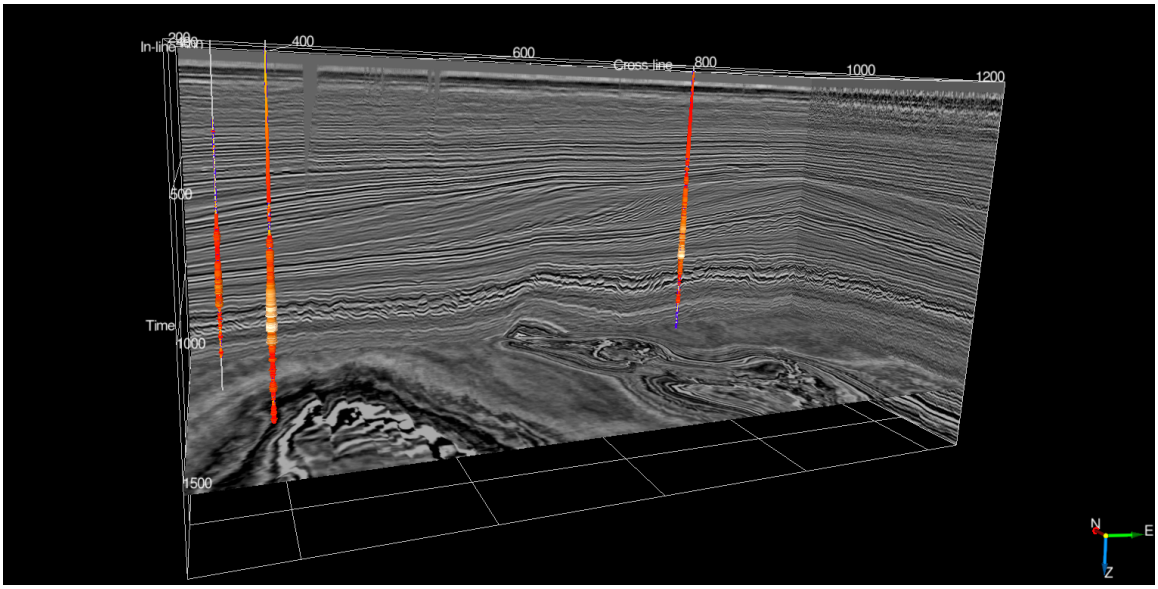


Figure 1.5: A migrated seismic volume with density logs from the Netherlands F3 block [5].

and seismic data is a key element in characterizing the reservoir. Finally, the last stage of of the pipeline is oil production through which oil fields are developed and oil is extracted and sent to the downstream pipeline.

Seismic surveys can span tens or hundreds of square kilometers and large amounts of data are collected in daily basis. Therefore, advances in signal processing and computation power have been pivotal to the success of oil and gas exploration and production. Moreover, with the exponential increase of collected data from seismic surveys with high-resolution acquisition technology, recent advances in image processing, machine learning, and deep learning have a great potential of accelerating the various stages in the pipeline. One objective of this dissertation is to *leverage these recent advances to accelerate subsurface volume characterization using seismic data in addition to the limited well-log data.*

1.3 Subsurface Volume Characterization

Subsurface volume characterization is a process through which the true geology of the subsurface is inferred from seismic reflection data and the sparsely-available well-log data. It can be categorized according to its objective into three categories: structure, stratigraphy, and lithology characterization [2]. In structure characterization (pre-drill), the objective is to use the recorded seismic data to identify and highlight geological phenomena in the volume that have common structural characteristics (e.g., salt bodies, channels, and faults). Such structures represent oil traps to be targeted in drilling, or drilling hazards to be avoided. Note that only seismic data is used in pre-drill interpretation. In stratigraphy characterization (post-drill), the objective is to highlight the different depositional and sedimentary environments in the volume. Finally, the objective of lithology characterization (post-drill) is to infer the physical properties of the subsurface such as permeability and porosity from the observed seismic data. Post-drill interpretation utilizes seismic data, well-logs and core data to characterize the subsurface at a resolution higher than the seismic resolution. Examples of the three types of characterization are depicted in Figure 1.6.

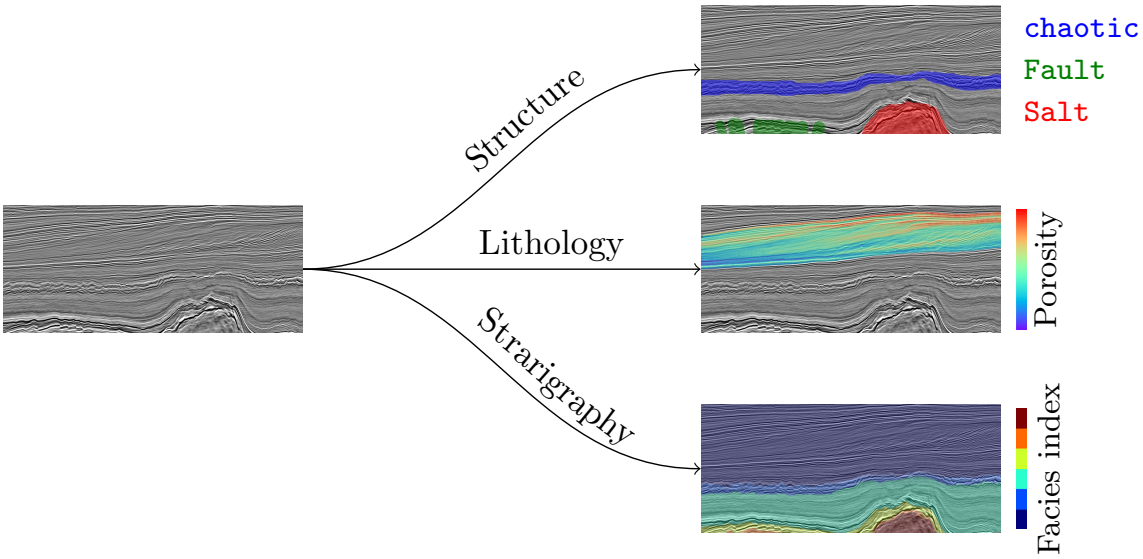


Figure 1.6: Examples of the three types of subsurface characterization.

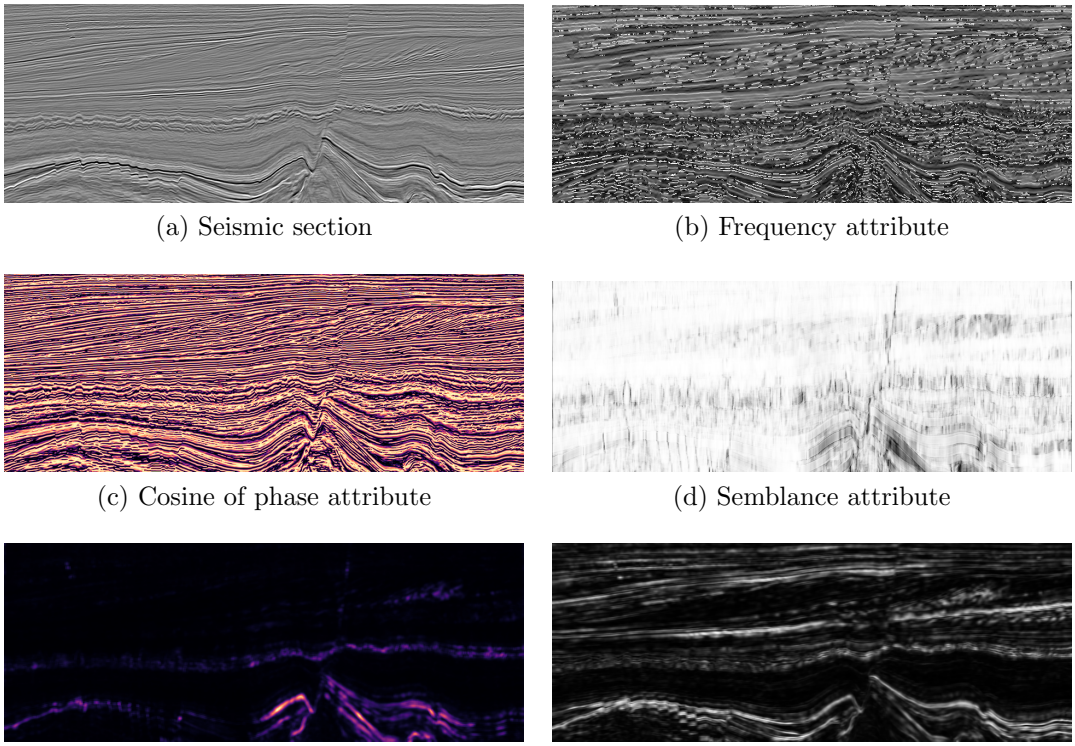
As discussed earlier, subsurface volume characterization is a labor-intensive process that can take a team of experienced geoscientists several months. It involves several steps including making structure and facies maps, petrophysical analysis of well-logs and core data, observing patterns in the data and linking them to geologic phenomena, and eventually, highlighting regions of potential reservoirs. The outcomes of this process are multiple valid interpretations of the subsurface, all of which are in agreement with the data. Thus, interpreters often improve their existing interpretations when new findings or data are available including well-logs, core data, technical reports from field engineers. For instance, in pre-drill interpretation where only low-resolution seismic data is available, interpretation uncertainty is high. However, when wells are drilled, data from those wells are used to compare the true geology with the existing interpretation to improve the interpretation, and thus, reduce uncertainty.

In structure characterization (pre-drill), interpreters often rely on seismic attributes to analyze seismic data. Seismic attributes are mathematical quantities that are computed from the seismic data such as discontinuity, curvature, dip angle, and semblance, and several others [8]. These attributes are also used to assess the similarity of two different regions within the seismic volume. If two regions of the subsurface volume have similar attribute values, they are likely to contain similar structures. Some attributes analyze waveform variations of the seismic traces and evaluate the attribute based on these variations. Examples of such attributes are frequency, the cosine of phase, and amplitude. Alternatively, the analysis could be carried out using image-oriented attributes of the seismic sections or on a local window of the section. Examples of image-oriented attributes are semblance and texture attributes from the gray level co-occurrence matrix (GLCM) [9].

Figure 1.7 shows a few waveform and texture attributes of a seismic section. The figure shows that image-oriented attributes, such as texture, can highlight the different subsurface structures since they operate on local 2-dimensional or 3-dimensional

windows of the seismic data. Thus, texture attributes have been widely used in structure interpretation of seismic data. In particular GLCM attributes have been used extensively in the seismic domain to characterize seismic images from a structural point of view [10, 11, 12, 13]. Seismic structures with similar characteristics have similar responses across the various attributes. Therefore, interpreters analyze seismic attributes to identify and distinguish seismic structures of interest which requires inspecting several attributes at once. However, with the exponential growth of data, interpreters relied on clustering and dimensionality reduction techniques to control the amount of data to be analyzed, and to identify the most effective attributes to distinguish subsurface structures [14]. However, a limitation of using classical machine learning techniques such as clustering methods is that they require a manual choice of the input attributes to be computed or used (feature engineering). On the other hand, deep learning methods can offer a more convenient way to characterize the subsurface without the need to manually select attributes [15, 16, 17]. Deep neural networks can compute their own features (attributes) that are optimized for the task for which they are trained. However, deep learning models are known to require tremendous amounts of annotated data to be trained properly which limits the applications of deep learning in the seismic domain.

In post-drill characterization, additional data obtained from well-logs and core data offer high-resolution measurements of the subsurface. Although well-log data is sparely-available, they play a key role in the development of a seismic model (forward model) that can adequately represent the seismic data [3]. Seismic modeling is a tool that is used to simulate the seismic data from well-log data. The simulated data (seismograms) is then compared with the recorded data. If the compared data do not match, the error is used to update the parameters of the model such as its wavelet. Figure 1.8 shows a seismogram generated from well-log data (density and P-wave velocity) compared to the recorded seismic trace. Seismic models are based on the



(a) Seismic section (b) Frequency attribute

(c) Cosine of phase attribute (d) Semblance attribute

(e) GLCM contrast attribute (f) GLCM variance attribute

Figure 1.7: A seismic section and its corresponding attributes. Data shown for inline 301 from Netherlands offshore F3 block.

physics governing wave-propagation to represent the seismic data. Once a seismic model is obtained, it can be used in lithology characterization to generalize well-log data beyond well locations guided by the seismic data. This process is referred to as seismic inversion [18]. In seismic inversion, an initial guess of the lithology of the subsurface is obtained from the few well-logs. Then, the seismic model is used to simulate seismic data from the initial guess. The error between the simulated seismic data and recorded seismic is then used to update the initial guess. The process is repeated until convergence.

More recently, there have been several attempts to integrate seismic data and well-logs in a learning-based lithology characterization framework [19, 20, 21, 22]. The problem is set up as a supervised learning problem with well-log data as labels for seismic data. Then, a mapping from seismic data to rock properties is learned in

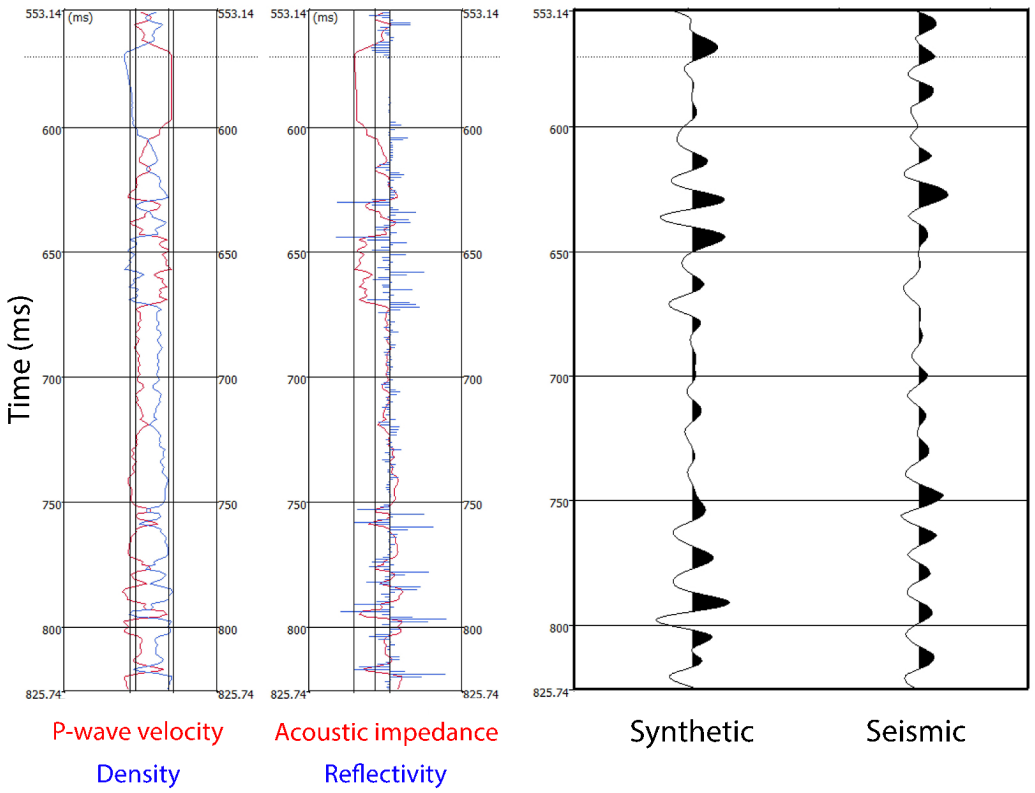


Figure 1.8: Synthetic seismogram generated from well-log data and its corresponding recorded seismic trace at the well location.

a regression routine. The learned mapping is then used to estimate rock properties throughout the seismic survey using the seismic data. However, the learned mapping often fails to generalize beyond the training data due to the limited number of available well-logs for training. The limited available data also makes it difficult to train a deep-learning model with hundreds of thousands or millions of parameters. Finally, such learning-based methods do not utilize a seismic model. Instead, they often rely on the training data to learn this complex mapping from seismic data to rock properties.

To overcome the limitations of using deep learning in the seismic domain, we introduce *a learning-based framework that enables the characterization of subsurface volumes with limited labeled data*. Specifically, given a few manually-labeled images of subsurface structures of interest, we utilize texture-based image similarity measures to

identify similar surface structures within a large seismic volume for pre-drill subsurface structure characterization using seismic data only. With a structural interpretation of a subsurface volume, wells are drilled which enable access to the true geology of the subsurface. Well-log data is then used in the introduced framework as limited high-resolution labels of the seismic data in *a semi-supervised sequence modeling framework* to characterize the subsurface with high resolution from a lithological point of view. The introduced framework offers an end-to-end subsurface characterization framework that integrates seismic data, well-log data and seismic modeling to learn from limited annotated data. An illustration of the different elements of the framework introduced in this dissertation is shown in Figure 1.9.

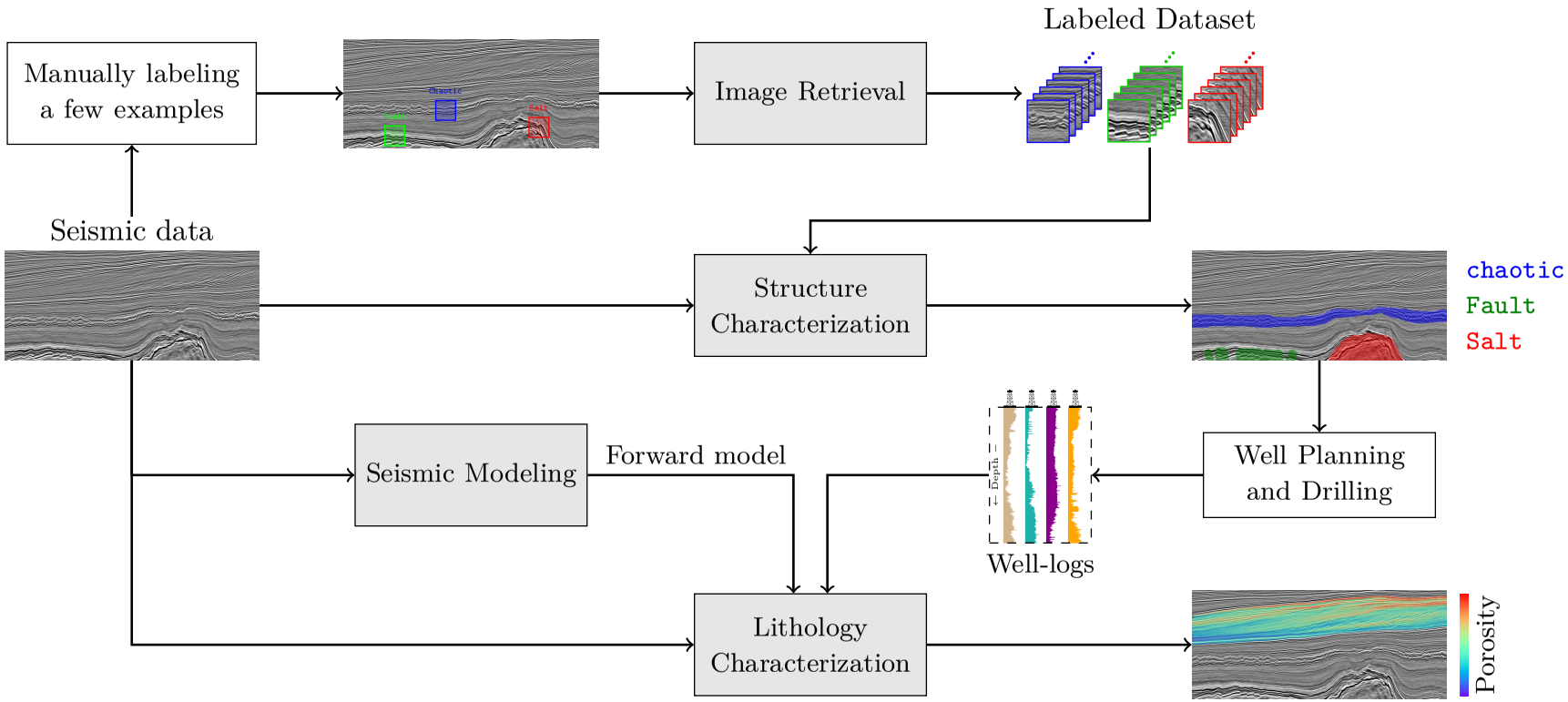


Figure 1.9: A visual illustration of the developed framework in this dissertation.

1.4 Outline

In Chapter 2, we introduce a novel multiresolution texture-based image similarity measure that can characterize seismic images from a structural point of view. The proposed similarity measure is then used to retrieve seismic images from the volume that are similar to a few manually labeled images. The outcome of image retrieval is a large dataset of labeled seismic images. We then present a learning-based framework that uses the labeled dataset to characterize a subsurface volume from a structural point of view. The presented framework is evaluated in a case study of structure interpretation of the Netherlands offshore F3 block [5].

In Chapter 3, we introduce *a sequence modeling framework for lithology characterization* using a shallow recurrent neural network that integrates well-logs data and seismic data in a learning framework. This framework learns a mapping from seismic data to rock property measurements from well-logs by using well-log data as labels of the seismic data. It can then be used to characterize the lithology of the entire subsurface volume from seismic data. We evaluate the proposed framework in two case studies of rock property estimation from real and synthetic seismic data. We also show the value of using a recurrent neural network for seismic modeling by comparing the proposed framework with an identical feed-forward one.

In Chapter 4, we introduce *a physics-guided semi-supervised deep sequence modeling framework for subsurface lithology characterization*. The framework uses knowledge from a seismic forward model based on wave-propagation physics to learn a better mapping from seismic to rock properties. The predefined forward model enables the framework to learn not only from seismic data for which well-logs are available, but it can learn from all seismic traces in the survey with explicitly requiring corresponding rock property measurements. We evaluate the proposed framework in two case studies of acoustic and elastic impedance estimation from noisy synthetic and migrated

seismic data from Marmousi 2 dataset [23]. We also compare the results using the proposed methods with classical seismic inversion methods that are commonly used in the literature. In Chapter 5, we introduce an extension of the semi-supervised framework that requires no modeling of the seismic data. Instead, it can learn the seismic forward model solely from the training data. The proposed framework is then used to estimate acoustic and elastic impedance from noisy synthetic and migrated seismic data from Marmousi 2 dataset. We also compare the results using the proposed method with classical inversion methods and the method proposed in Chapter 4. Finally, in Chapter 6, we present a summary of the proposed frameworks and main contributions, and future research directions to conclude this dissertation.

CHAPTER 2
MULTIRESOLUTION TEXTURE ANALYSIS FOR STRUCTURE CHARACTERIZATION

2.1 Overview

With the exponential growth of collected seismic data, manual annotations of seismic volumes becomes an increasingly demanding task. Moreover, uncertainty in seismic imaging can result in different but valid manual interpretations of the subsurface. This is especially true in pre-drill interpretation where seismic data is the only available data from the subsurface. For instance, in structure interpretation, interpreters rely on visual analysis of the seismic data to identify and highlight different subsurface structures. Moreover, seismic attributes, such as texture, are used to identify regions within the seismic volume that have a similar response which are then said to contain a similar subsurface structure. Thus, quantifying the similarity of seismic images can enable the acceleration of structure interpretation.

In this chapter, we introduce a framework to characterize seismic volumes using multiresolution texture analysis. We model seismic data as monochromatic images and use multiresolution texture analysis to characterize subsurface structures. Specifically, we develop a texture-based image similarity measure that quantifies the similarity between two images with respect to their textural content. The developed similarity measure is then used to search large unlabeled seismic volumes for images that are similar to a manually picked image of a given subsurface structure. Images that have high similarity to the reference image are then given the same label as the reference image. The process is repeated for different subsurface structures to obtain a large dataset of labeled seismic images. Finally, the introduced similarity measure

is integrated within a segmentation framework that uses the large labeled dataset of seismic images to characterize the subsurface from structural point of view. The objective of this chapter is illustrated in Figure 2.1

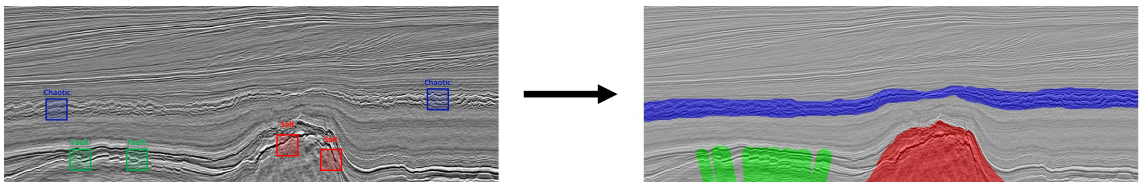


Figure 2.1: The objective of the introduced structure characterization framework.

Section 2.2 presents an overview of image retrieval, texture similarity, and the relevant literature. Then, in Section 2.2.3, we present an overview of image decomposition methods that are commonly used to characterize texture images. In Section 2.3, we present the details of the developed multiresolution texture similarity measure. In addition, we evaluate the developed similarity measure in a retrieval experiment for natural and synthetic texture images. In Section 2.4, we evaluate the similarity measure for subsurface structure characterization in retrieval and clustering experiments. Finally, in Section 2.5, we present a framework to characterize the subsurface using seismic data and only a few manually labeled images of the subsurface structures of interest.

2.2 Background

2.2.1 Content-based Image Retrieval

The exponential growth of visual and pictorial content undoubtedly drives an increasing need for image similarity measures that can be utilized for various computer vision applications such as image retrieval, characterization, and recognition. The similarity of two images is often measured with respect to some attributes such as object presence, color, shape, and texture [24]. If these attributes are available with the image, quantifying the similarity between two images is a matter of comparing

their respective attributes. However, obtaining image attributes often requires human input which is time-consuming and labor-intensive. Furthermore, the task becomes even more demanding and expensive in areas where image attributes can only be provided by subject-matter experts, e.g., medical images and seismic images. There have been many attempts in the literature to develop systems that can compute such attributes from a visual perspective, and use these computed attributes to compare images. This field of research is known as Content-Based Image Retrieval (CBIR) [24, 25, 26, 27]. The objective of CBIR is to quantify image similarity with respect to some attributes without the need for human-generated labels or descriptive data of images. Figure 2.2 shows a general CBIR framework to search a large dataset for images, and retrieve the most similar ones to a given query image.

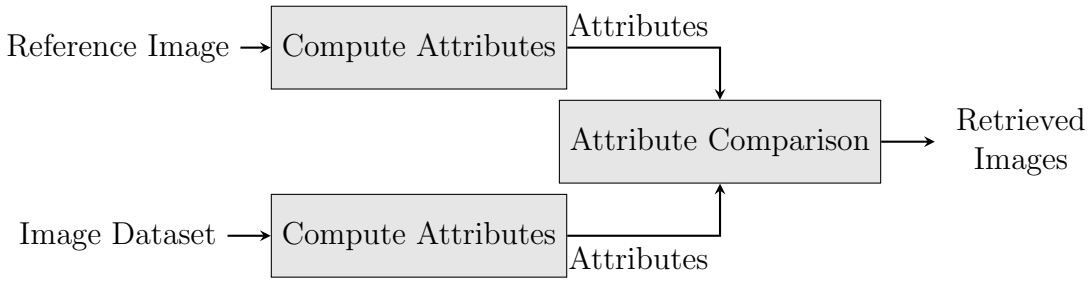
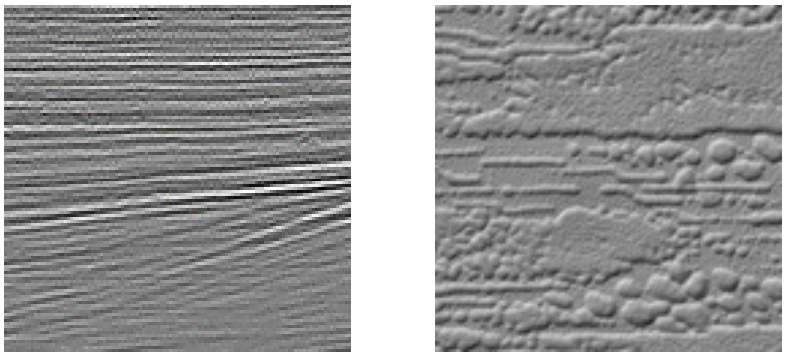


Figure 2.2: A general content-based image retrieval workflow.

To characterize images, attributes are chosen depending on the application. For instance, object presence is a pivotal attribute for image captioning applications where the goal is to automatically generate a description of an image [28, 29, 30]. Moreover, texture attributes have played a key role in the characterization of medical images, especially for identify different tissues [31, 27, 32, 33]. Texture has also been used extensively to characterize seismic images [34, 35] due their highly textural content as shown in Figure 2.3. Texture attributes are often utilized to discriminate various subsurface structures that are present within seismic images [36], i.e., subsurface structure characterization.



(a) Seismic image (b) Synthetic texture image

Figure 2.3: An illustration of the similarity between seismic images and texture images.

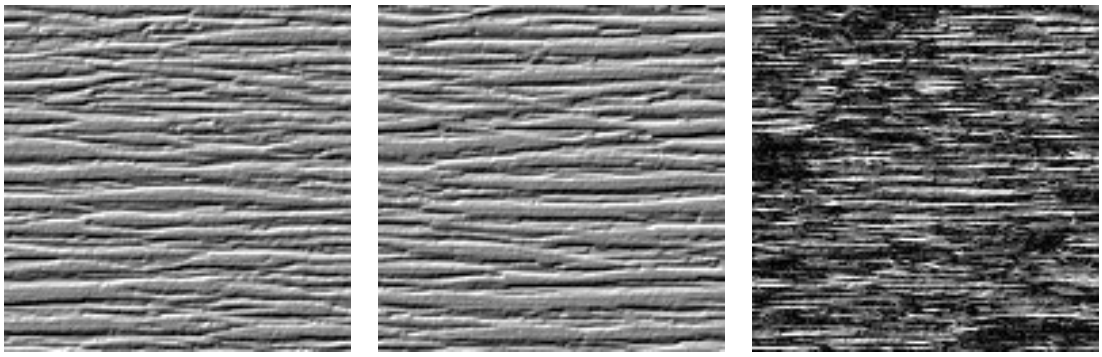
2.2.2 Image Similarity

Image similarity measures are functions that quantify the degree of similarity between two images $\mathbf{X}_1, \mathbf{X}_2 \in \mathbb{R}^{m \times n}$, i.e.,

$$\mathrm{SIM}(\mathbf{X}_1, \mathbf{X}_2) : \mathbb{R}^{m \times n} \times \mathbb{R}^{m \times n} \rightarrow \mathbb{R}. \tag{2.1}$$

Image similarity measures are developed for various applications such as image retrieval, image quality assessment, and image synthesis. The similarity of two images is often measured with respect to an application-dependent attribute such as shape, color or texture. For example, one can choose the attribute of interest to be the texture such that two images are said to be similar if their textural contents are similar. Texture-based image similarity (or distance) measures differ from their generic counterpart measures such as Peak Signal-to-Noise Ratio (PSNR) and Mean Square Error (MSE) in that they capture the content of an image rather than assuming a pixel-to-pixel correspondence. Figure 2.4 shows an example of two images with the same texture and their absolute difference. This example shows that pixel-based comparisons are not fit for measuring texture similarity.

Texture similarity measures can be categorized, according to the domain on which



(a) Image 1 (b) Image 2 (c) Absolute difference

Figure 2.4: Two images with similar texture and their absolute difference.

the feature extraction is carried out, into two main categories: spatial methods or frequency-based methods. For instance, the gray level co-occurrence matrix (GLCM) [9] is a popular spatial characterization method of texture images using local statistics of quantized images. Structure SIMilarity (SSIM)[37] measures improve upon the generic measures, e.g., MSE, by capturing structure using low-level local statistics such as mean and variance in the spatial domain. Local Binary Patterns (LBP) [38] is another spatial texture characterization method that constructs binary pattern maps based on the value of a pixel relative to its neighbors. Then, statistics are computed from the histograms of these patterns for different radii. Other variants of LBP have been proposed to improve upon the original LBP such as Dominant Local Binary Pattern (DLBP)[39], Local Derivative Pattern (LDB) [40], and Completed Local Derivative Pattern (CLDB) [41].

Alternatively, texture analysis can be carried out in the frequency domain using multiresolution analysis methods. For example, The Complex Wavelet Structure SIMilarity (CW-SSIM)[42] improves on SSIM by comparing low-level statistics of images in the frequency domain using the complex wavelet transform instead of the spatial domain. Structural Texture SIMilarity metric (STSIM)[43] uses sub-band statistics and correlations in the frequency domain using the steerable pyramid. Zhang *et al.* [44] proposed the use of low-level statistics of the curvelet coefficients to quantify

the similarity of texture images. Selvan and Ramakrishnan [45] proposed a texture characterization model based on the probability density function of the singular values of the wavelet transform. Arivazhagan *et al.* [46] proposed a similarity measure based on statistical and co-occurrence features of the curvelet coefficients for texture characterization.

In general, the practice of using image decomposition (either in the frequency or the spatial domain) followed by feature extraction and comparison is very common in texture image characterization [47]. A generic texture similarity workflow is depicted in Figure 2.5.

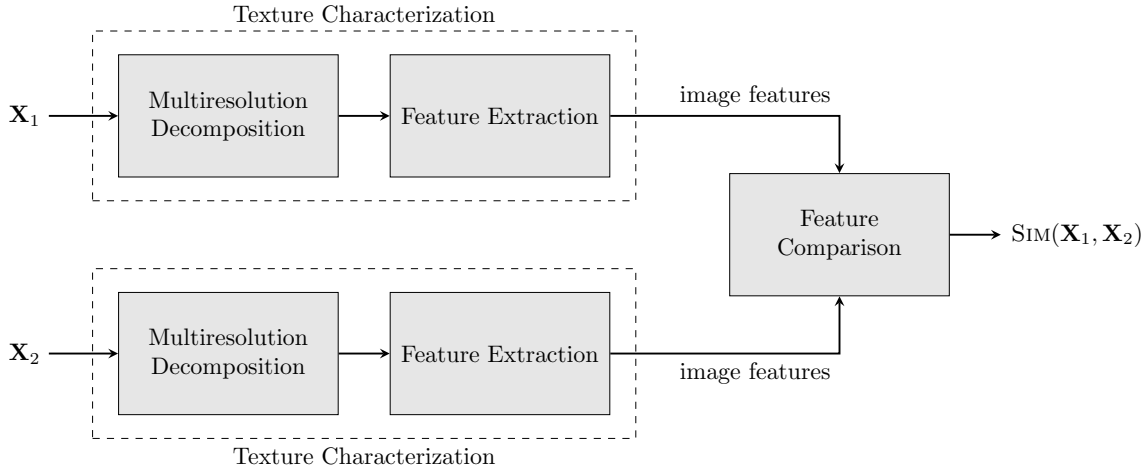


Figure 2.5: A generic block diagram of a texture similarity measure.

Texture characterization has been adopted widely to characterize seismic data by viewing seismic sections as images. Texture has proved to be an effective attribute to characterize seismic images due to the textural nature of migrated seismic data. For instance, Pitas and Kotropoulos [34] proposed a texture analysis method based on Hilbert transform features for seismic image segmentation. Röster and Spann [35] characterized subsurface structures from seismic reflection data using Gabor filters. Hegazy *et al.* [48] and Shafiq *et al.* [36] proposed methods for salt body detection using the Gradient of Texture (GoT). Di *et al.* [49] utilized a set of texture attributes to detect faults in seismic volumes. In addition, texture-based similarity measures

have been proposed to assess the similarity of seismic images based on the subsurface structures they contain. For example, Al-Marzouqi and AlRegib [50] proposed an adaptive version of the curvelet transform to characterize seismic images. Long *et al.* [51] proposed a similarity measure of seismic images using STSIM in addition to a seismic attribute known as discontinuity maps. Alaudah [52] proposed a seismic similarity measure that uses the histograms of the curvelet coefficients as features to characterize subsurface structures. Long *et al.* [53] and Mattos *et al.* [54] presented comparative studies of the use of texture attributes to retrieve seismic images that contain similar subsurface structures. Their findings indicate that texture attributes are suitable descriptors to characterize subsurface structures. This is especially true when subsurface structures are not separated by clear edges. For instance, Shafiq *et al.* [55] compared edge-based and texture-based methods for salt body detection. The results of the study showed that texture features can detect salt bodies even in the absence of strong reflections (edges). In general, a review of the literature shows that texture is a critical attribute to characterize seismic data for a wide variety of applications [11].

We identify four characteristics that we believe are important for a texture-based seismic similarity measure. The first characteristic is the insensitivity to image contrast since seismic data are often acquired with different equipment which directly affects the contrast of the seismic sections when viewed as images. The second characteristic is the use of multiresolution analysis which enables the similarity measure to characterize various subsurface features regardless of their physical dimensions. The third characteristic is feature sparsity, which improves robustness to noise. Sparse features allow the similarity measure to capture the most dominant features in seismic images and suppress noise which is common in seismic images. Finally, the developed measure must be cross-domain adaptable in order to work across different seismic surveys seamlessly. We believe that a similarity measure that addresses these four

characteristics is a good candidate for seismic images characterization.

2.2.3 Multiresolution Image Decomposition

In this section, we describe four multiresolution decomposition techniques that are commonly used for texture image analysis. Namely, the Gaussian pyramid, the discrete wavelet transform, Gabor filters, and the curvelet transform. The Gaussian pyramid is a technique that is used to decompose an image into different scales each of which comprises features of similar size. Pyramid decomposition, in general, laid the groundwork for multiresolution analysis techniques. The discrete wavelet transform improved on pyramid techniques by decomposing each scale into horizontal, vertical and diagonal components. Gabor filters introduce a directional decomposition by which the different scales are decomposed into more orientations than those obtained from the discrete wavelet transform. It is worth noting that Gabor filters do not form a transform, i.e., the image response to the filters does not fully represent the original image. The curvelet transform is an extension of the wavelet transforms to overcome its limited directionality. It decomposes an image into different frequency bands at different scales and orientations. Throughout this section, we will use the seismic image shown in Figure 2.6 to illustrate image decomposition using the aforementioned multiresolution analysis techniques.

The Gaussian Pyramid

The Gaussian pyramid is a classical multiscale analysis technique, which is the predecessor of multiresolution analysis techniques. It has been used for various applications in image and video processing such as image coding, image and video compression, and salient object detection [56, 57, 58]. In this work, the Gaussian pyramid serves as an efficient multiscale analysis tool to exploit features of different sizes in a seismic image. For a 2D seismic image, $\mathbf{X}_0$, the k-scale Gaussian pyramid is constructed as

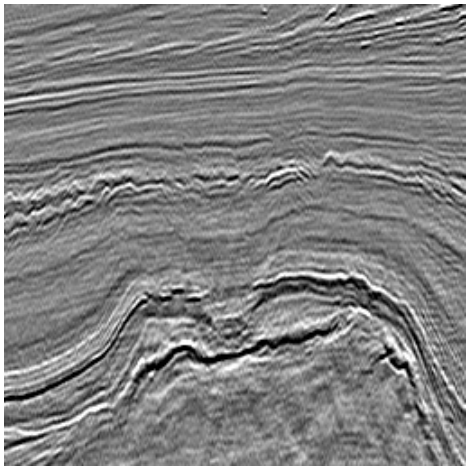


Figure 2.6: Part of a salt dome from inline 299 of the Netherlands offshore F3 block [5]

follows: First, $\mathbf{X}_0$ is set as scale 0 of the pyramid which represents the full resolution scale. Then, scale 1 of the pyramid, $\mathbf{X}_1$, is computed by smoothing $\mathbf{X}_0$ with a Gaussian filter, followed by downsampling it by a factor of 2. The remaining scales are generated in a similar fashion. An illustration of a 4-scale Gaussian pyramid of a seismic section is shown in Figure 2.7. Note that the dimensions of the image are reduced by a factor of 2 each time a one-level decomposition is performed (i.e., proceeding upward along the pyramid by one level). The Gaussian blurring filter serves as a low-pass filter and is followed by a downsampling step to avoid redundancy.

Discrete Wavelet Transform

The Discrete Wavelet Transform (DWT) is an orthonormal transform that represents an image using a dyadic dilation and translation of a certain basis function (or a mother wavelet). Different wavelet bases have been proposed and studied extensively such as Haar, Daubechies, symlet, Mexican hat, and coiflet wavelets, among many others [59]. A classical choice of such basis function is $\mathbf{h}_L = \frac{1}{2}[1,\ 1]^\top$ and $\mathbf{h}_H = \frac{1}{2}[-1,\ 1]^\top$, which is known as the Haar wavelet. The first-level discrete wavelet coefficients of an image, $\mathbf{X}$, are obtained by filtering along the horizontal direction

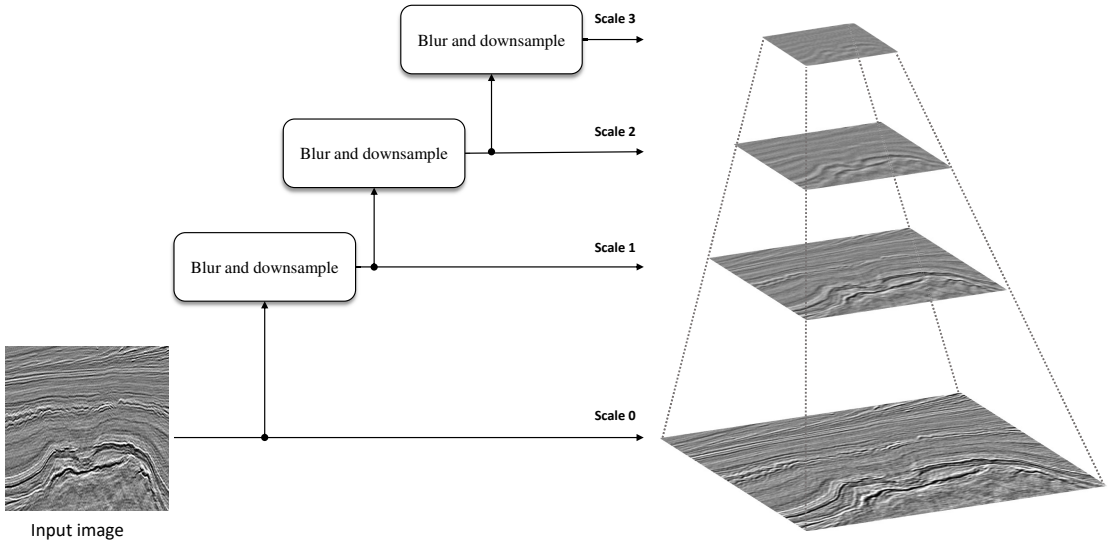
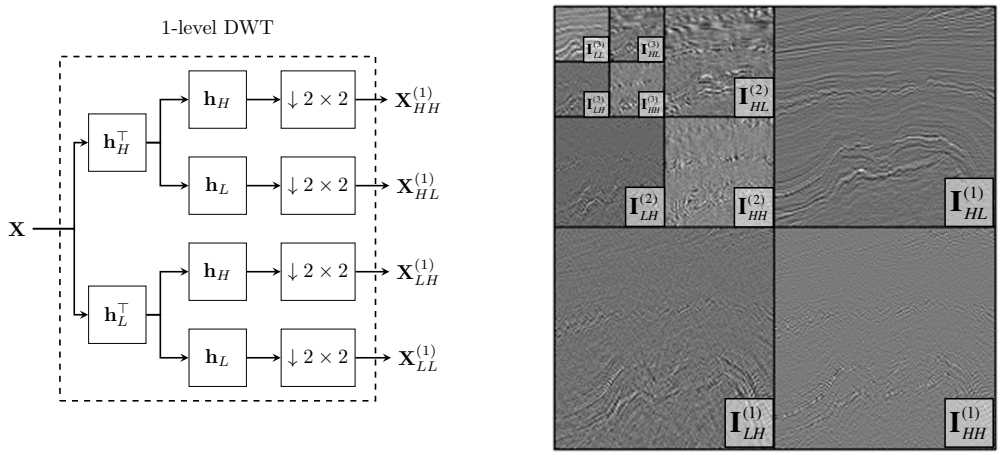


Figure 2.7: 4-scale Gaussian pyramid workflow.

with low pass $\mathbf{h}_L$ and high pass $\mathbf{h}_H$ filters to obtain $\mathbf{X}_L^{(1)}$ and $\mathbf{X}_H^{(1)}$ respectively. Then, $\mathbf{X}_L^{(1)}$ and $\mathbf{X}_H^{(1)}$ are filtered along the vertical direction with the same filter, and decimated by a factor of 2. The resulting images are three detail images: $\mathbf{X}_{HH}^{(1)}$, $\mathbf{X}_{HL}^{(1)}$, $\mathbf{X}_{LH}^{(1)}$, and one approximation image $\mathbf{X}_{LL}^{(1)}$. This process is illustrated in Figure 2.8(a). For more levels, the same process is repeated on the approximation image $\mathbf{X}_{LL}^{(1)}$. An example of a 2-level DWT of a seismic image is shown in Figure 2.8(b). The total number of subbands depends on the number of levels. For instance, a 4-level DWT will result in 13 subbands (3 detail subbands $\times$ 4 levels + 1 approximation subband).

Gabor Filters

Gabor filters are linear filters that are the product of a 2D plane wave with a Gaussian filter. Gabor filters are frequently used as models of the simple cell receptive fields in the human visual system [60]. They have been utilized to characterize natural and texture images especially for applications such as edge detection [61] and image segmentation [62]. Furthermore, Gabor filters have been used in seismic image



(a) 1-level 2D discrete wavelet transform workflow

(b) A 3-level discrete wavelet transform of the image in Figure 2.6.

Figure 2.8: An illustration of the discrete wavelet transform.

processing to extract useful characteristic features and define some seismic attributes [63]. Figure 2.9 shows the response of the image in Figure 2.6 to different Gabor filters at three scales and four orientations.

The Curvelet Transform

The curvelet transform is a multiscale directional decomposition. Despite their popularity, wavelets fail to compactly represent images with highly directional elements such as curves and edges. To the contrary, curvelet frames have been shown to represent images with geometrically regular edges (such as seismic images) more compactly than other traditional multiscale representations [64]. For an image with $N_1 \times N_2$ pixels, the fast discrete curvelet transform (FDCT) allows the computation of curvelet coefficients in $\mathcal{O}(N_1 N_2 \log N_1 N_2)$ operations making the curvelet transform not only fast to compute but also scalable to large images.

For the purposes of this work, we present a simplified overview of the FDCT. For a detailed description, refer to [64]. Given an image, the FDCT divides the Fourier support of the image into J scales and $K(j)$ orientations as is shown in 2.10(a) such

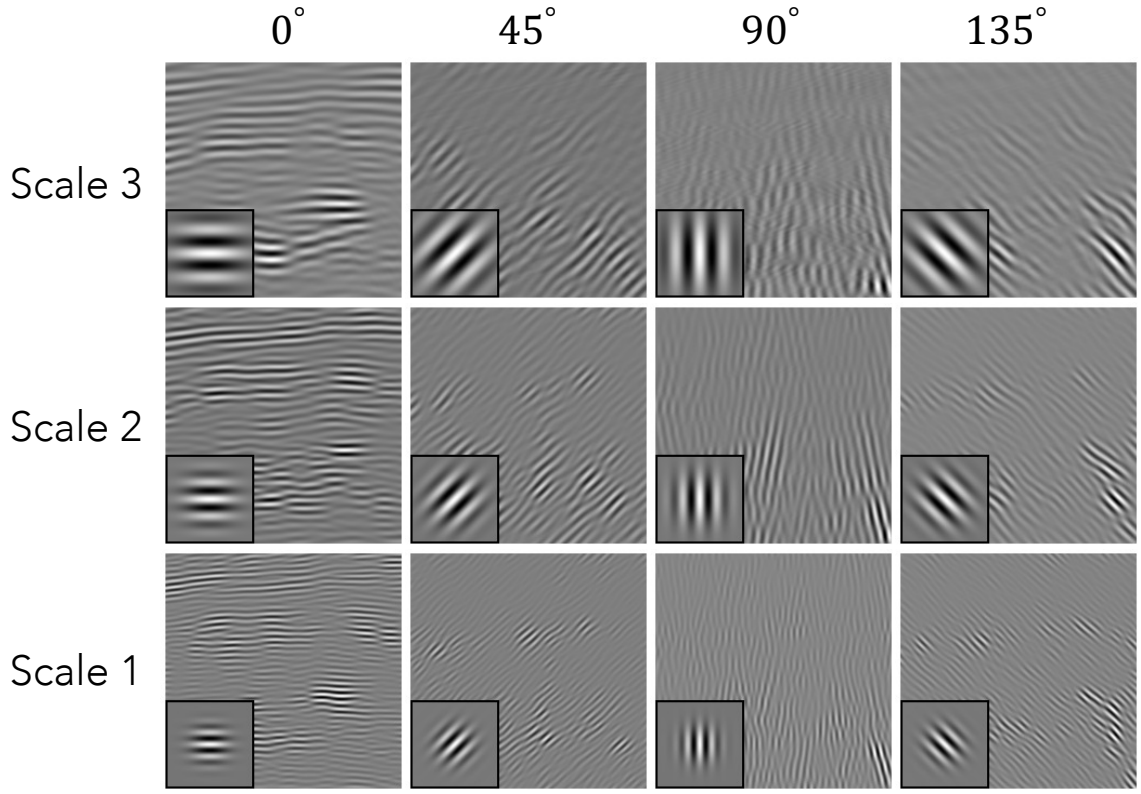
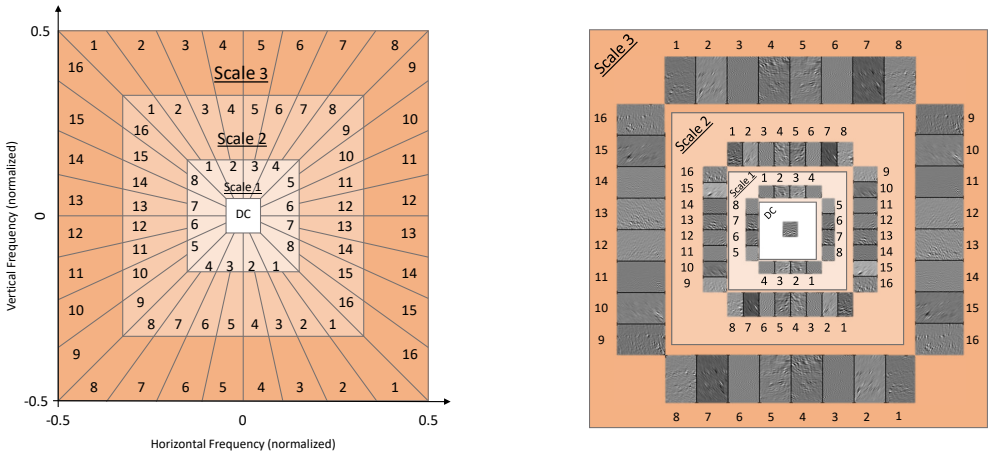


Figure 2.9: The image in Figure 2.6 filtered with Gabor filters at three scales and four orientations. The filters are shown at the bottom left corners

that $J = \lceil \log_2 \min(N_1, N_2) - 3 \rceil$, where $\lceil \cdot \rceil$ is the ceiling function. The number of orientations at scale j is given by $K(j) = 16 \times 2^{\lceil (j-1)/2 \rceil}$. The curvelet coefficients are then obtained by taking the Inverse Fast Fourier Transform (IFFT) of each subband after multiplying it by a smoothing function and wrapping it around the origin. The 3-scale curvelet coefficients of the image in Figure 2.6 are shown in Figure 2.10(b). Note that we use generic labels of the axes in Figure 2.10(a), namely, horizontal and vertical frequency instead of wavenumber and frequency in order to generalize the decomposition for inline/crossline as well as time slices.

2.3 Multiresolution Texture Similarity Measure

In this section, we present a multiresolution texture similarity measure based on the effective singular values of the curvelet transform. We evaluate the proposed similarity



(a) Frequency domain partitioning of a 3-scale curvelet transform.

(b) The real part of the corresponding curvelet coefficients.

Figure 2.10: An illustration of the curvelet transform of a seismic image. Orientations and scales are represented by numbers and colors, respectively.

measure in a retrieval experiment on three natural and synthetic texture datasets.

2.3.1 Methodology

Following the general structure of texture similarity measure, the proposed similarity measure consists of three main modules: multiresolution image decomposition, feature extraction, and feature comparison as shown in Figure 2.11. The details of each of the modules are presented in the next sections.

Multiresolution Image Decomposition

For an image, $\mathbf{X}$, multiresolution decomposition methods represent the input image using filters with different orientations and scales (sub-bands). Each sub-band $\mathbf{X}_{j,k}$ captures certain features of the input image at scale j and orientation $k(j)$ of the filter. In the proposed texture similarity measure, we utilize the discrete curvelet transform [65] to decompose images due to its effectiveness in capturing the directional content of images. The details of the curvelet transform were discussed in Section 2.2.3.

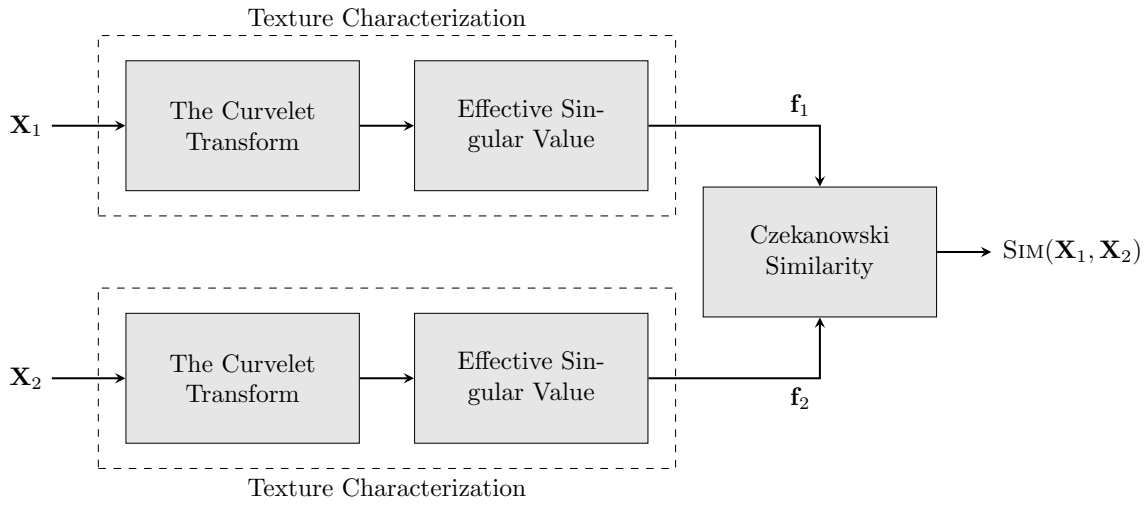


Figure 2.11: The proposed multiresolution texture similarity measure.

Feature Extraction and Comparison

As we have discussed in Chapter 2.2, it is common for texture characterization methods to extract features from statistics of each sub-band of the image, i.e., $\mathbf{X}_{j,k}$, such as mean and variance. Then, the feature vector, $\mathbf{f}$, that represents the image is obtained by concatenating all sub-band features. In the proposed similarity measure, we use the largest $r_{\text{effective}}$ singular values of each sub-band as features. The largest $r_{\text{effective}}$ singular values are referred to as the effective singular values, where $r_{\text{effective}}$ is an estimate of the effective dimensionality of the matrix proposed by [66]. Ideally, if the rank of a matrix is r, only the first r singular values are non-zero and the remaining ones are identically equal to zero. However, when we consider SVD on digital images that are subject to different types of noise, the number of non-zero singular values is greater than r. In most cases, none of the singular values are identically zero even for a rank-deficient image. To calculate the effective rank, a normalization function of the singular values is defined as:

$$p_{j,k}[i] = \frac{\sigma_{j,k}[i]}{\|\sigma_{j,k}\|_1}, \tag{2.2}$$

where $\sigma_{j,k}[i]$ is the i^{th} largest singular value of the sub-band $\mathbf{X}_{j,k}$ and $\|\sigma_{j,k}\|_1$ is the ℓ_1 norm of the vector of all singular values of $\mathbf{X}_{j,k}$. Then, the effective rank is calculated as a function of the entropy of the expression in (2.2), i.e.,

$$r_{\text{effective}}(j,k) = \exp\left(-\sum_{k=1}^{L} p_{j,k} \log p_{j,k}\right). \tag{2.3}$$

The effective rank of a matrix is a real number less than or equal to the smaller dimension of the matrix, with equality *if and only if* all singular values are equal. We use log to denote the natural logarithm and we follow the convention that $0 \log 0 = 0$.

For each sub-band, $r_{\text{effective}}$ is computed as in (2.3). A new vector of effective singular values is formed by keeping the first $\lfloor r_{\text{effective}} \rfloor$ singular values, where $\lfloor r_{\text{effective}} \rfloor$ is the integer part of $r_{\text{effective}}$. The remaining singular values are set to 0, i.e. for scale j and orientation k, we form the vector $\sigma_{j,k} = [\sigma_{j,k}[1], \ldots, \sigma_{j,k}[r_{\text{effective}}], 0, \ldots, 0]$. The feature vector of the image $\mathbf{X}$ is obtained by concatenating all $\sigma_{j,k}$ for all scales and orientations. The feature vector of an image is expected to be sparse since the non-effective singular values were set to zero.

Finally, the similarity of two images $\mathbf{X}_1$ and $\mathbf{X}_2$ is computed by comparing their corresponding feature vectors, $\mathbf{f}_{\mathbf{X}_1}$ and $\mathbf{f}_{\mathbf{X}_2}$. To compare the two feature vectors, we use an ℓ_1-norm based metric, namely, the Czekanowski similarity as follows:

$$\textsc{Sim}(\mathbf{X}_1, \mathbf{X}_2) = 1 - \frac{\|\mathbf{f}_{\mathbf{X}_1} - \mathbf{f}_{\mathbf{X}_2}\|_1}{\|\mathbf{f}_{\mathbf{X}_1} + \mathbf{f}_{\mathbf{X}_2}\|_1}. \tag{2.4}$$

Since the singular values real matrices are non-negative by definition, $\textsc{Sim}(\mathbf{f}_{\mathbf{X}_1}, \mathbf{f}_{\mathbf{X}_2}) \in [0, 1]$ with a value of 1 for identical images.

2.3.2 Performance Evaluation

In order to assess the performance of the proposed multiresolution texture similarity measure, we set up a retrieval experiment. With a dataset of N images that are

uniquely grouped into C distinct classes, image retrieval is carried out as follows. First, each image in the dataset is used as a query image. The similarity between the query images and all images in the dataset is computed using a similarity measure. Then, the top $\frac{N}{C} - 1$ images are retrieved. We then used information retrieval metrics in order to assess the quality of the retrieved images.

We also compare the proposed multiresolution texture similarity measure with various texture and generic similarity measures such as:

- Mean Squared Error (MSE).
- Mean Structural Similarity (M-SSIM) [37].
- Complex-Wavelet Structural SIMilaity (CW-SSIM) [42].
- Structural Texture SIMilarity (STSIM) [43].
- Local Binary Patterns (LBP) [38].
- Rotation-variant (RV) and rotation-invariant (RI) curvelet features [44].

Datasets

The retrieval experiments were carried out on three datasets of natural and synthetic texture images. For each dataset, image retrieval was carried out independently. The datasets used in this experiment are:

1. `CUReT` [67]: A dataset of 61 unique natural texture images. We extracted 3 non-overlapping patches of size 128×128 from all images with viewing condition number 55.
2. `PerTex` [68]: A dataset of 334 high resolution synthetic texture images. Each image in the database was downsampled by a factor of 4 then divided into 4 non-overlapping quadrants of size 128×128.

3. `Forrest` [69]: A dataset of natural textures. In our experiments, we randomly selected 100 images from Fabric, Stone and Wood albums. Then, we extracted 4 patches of size 128×128 from each image with no more than 50% overlap.

The datasets are summarized in Table 2.1. Sample images from these datasets are shown in Figure 2.12.

Dataset	Number of classes (C)	Number of images (N)
CUReT [67]	61	183
PerTex [68]	334	1336
Forrest [69]	100	400

Table 2.1: Summary of the texture datasets used in the retrieval experiments.

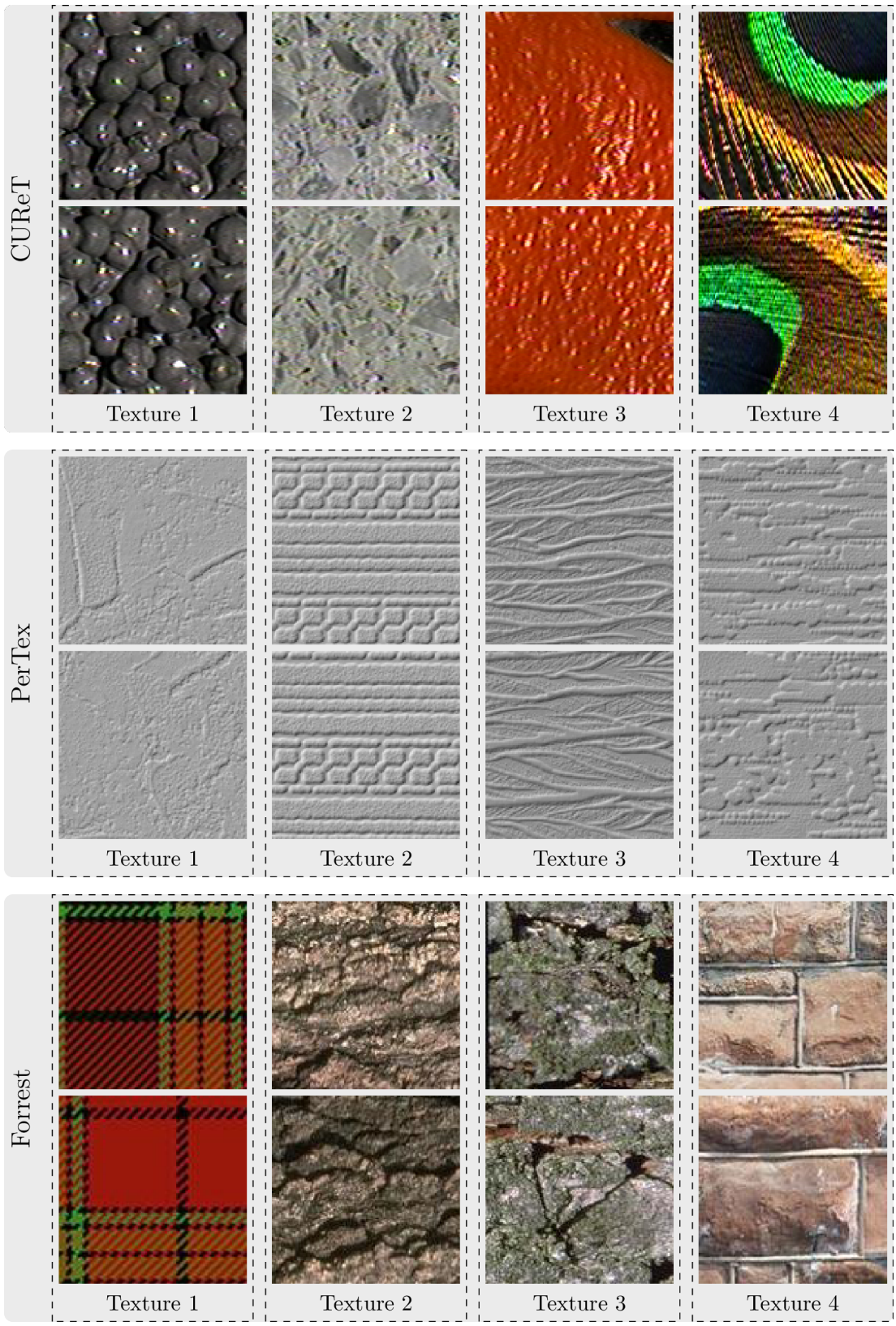


Figure 2.12: Sample images from CURet [67], PerTex [68], and Forrest [69] datasets.

2.3.3 Results

We evaluate the quality of the retrieved images using information retrieval metrics. Namely, we use the following metrics:

1. Precision at one (**P@1**) is the percentage of the first retrieved images that are of the same class as the query image.

2. Retrieval Accuracy (**RA**) is the percentage of images that have been correctly retrieved after retrieving all relevant images.

3. Mean Reciprocal Rank (**MRR**) is the average reciprocal rank of the first correctly retrieved images. **MRR** values are within $[0, 1]$ with a value of 1 indicating a perfect score.

4. Mean Average Precision (**MAP**) is a generalization of **MRR** that takes into account all retrieved images instead of only the first correctly retrieved one. Note that **MRR** differs from **RA** in that it takes into account the order in which the images are retrieved.

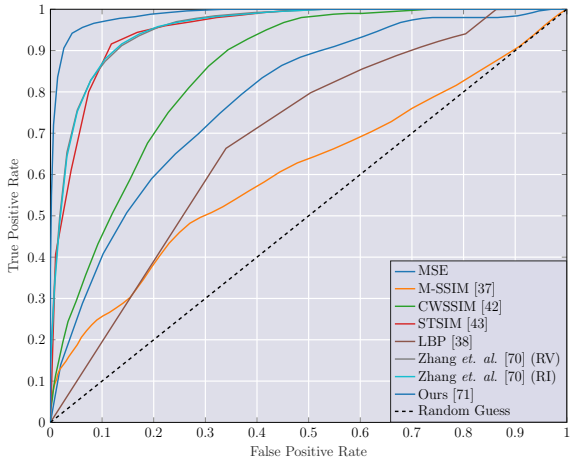
5. Receiver Operating Characteristic (**ROC**) is a plot of the True Positive Rate (TPR) versus the False Positive Rate (FPR) for different threshold values of detection. The Area Under the ROC Curve (AUC), which is a value between 0 and 1, is a measure of the discriminative power of the similarity measure with higher values indicating more discriminative power.

The details of these metrics are presented in Appendix A.

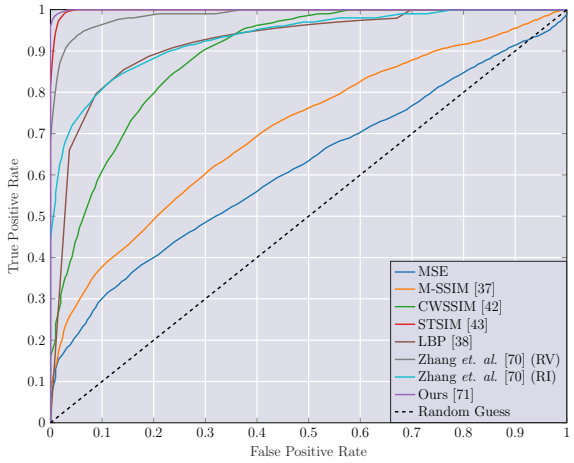
The results of the retrieval experiments for CUReT, PerTex, and Forrest datasets are summarized in Table 2.2. The results show that the proposed multiresolution texture similarity measure outperforms all other similarity measures across all three datasets. The performance on PerTex dataset is higher than it is on the other datasets

due to the fact that PerTex is a synthetic dataset of texture images. However, the proposed similarity measure has shown consistent performance across all three datasets despite the unique characteristics of each dataset. Furthermore, note that MSE, SSIM, and CW-SSIM perform significantly lower than the other measures because they are generic image similarity measures that were not designed for texture images.

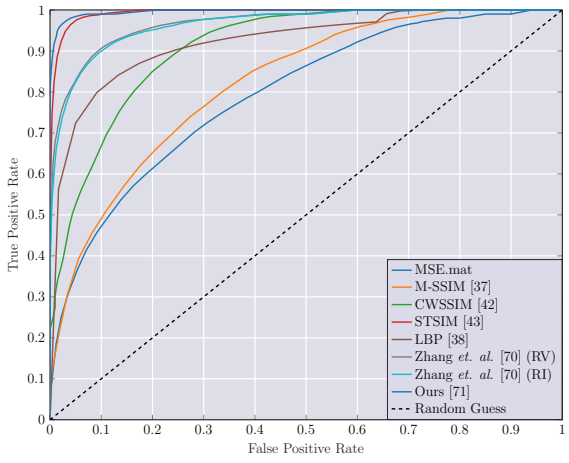
Moreover, Figure 2.13 shows the ROC curves for the different similarity measures along with the ROC curve for a random guess on the three texture datasets. The ideal ROC curve is the one with than passes along the point $(1, 0)$. Thus, an ROC curve that is pushed towards the upper left corner reflects a high discriminate power and, equivalently, has an AUC that is close to 1. The AUC curves are reported in Table 2.2.



(a) CUReT [67]



(b) PerTex [68]



(c) Forrest [69]

Figure 2.13: Receiver Operating Characteristic curves (ROC) for different similarity measures on three texture datasets.

Table 2.2: Performance evaluation of texture similarity measures on CUReT [67], PerTex [68], and Forrest [69] datasets.

CUReT [67]

Method	P@1	MRR	MAP	RA	AUC
MSE	0.1093	0.1867	0.1728	0.0902	0.7917
M-SSIM [37]	0.0546	0.0952	0.0935	0.0410	0.6080
CW-SSIM [42]	0.1366	0.2638	0.1967	0.1339	0.8533
STSIM [43]	0.8251	0.8893	0.8331	0.7596	0.9702
LBP [38]	0.8415	0.8843	0.8347	0.7732	0.7920
Zhang *et. al.* [70] (RV)	0.8142	0.8649	0.7669	0.7022	0.9624
Zhang *et. al.* [70] (RI)	0.8033	0.8562	0.7528	0.6803	0.9620
Ours [71]	**0.9617**	**0.9732**	**0.9304**	**0.8880**	**0.9918**

PerTex [68]

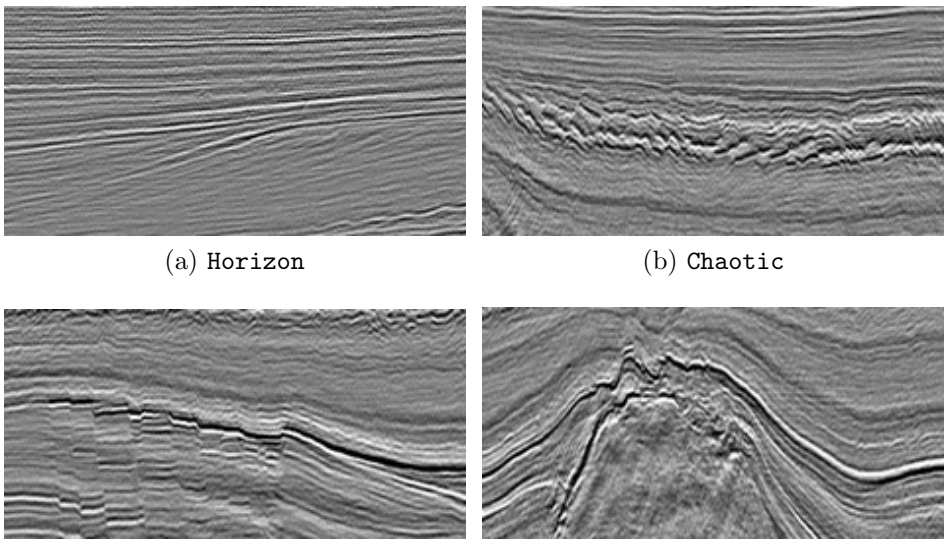
Method	P@1	MRR	MAP	RA	AUC
MSE	0.1115	0.1296	0.0641	0.0469	0.6168
M-SSIM [37]	0.1572	0.1855	0.1051	0.0828	0.7092
CW-SSIM [42]	0.1826	0.2527	0.1665	0.1402	0.8897
STSIM [43]	0.9094	0.9405	0.8887	0.8507	0.9987
LBP [38]	0.9117	0.9407	0.8628	0.8161	0.9842
Zhang *et. al.* [70] (RV)	0.9499	0.9659	0.9098	0.8695	0.9876
Zhang *et. al.* [70] (RI)	0.7680	0.8112	0.6336	0.5833	0.9317
Ours [71]	**0.9880**	**0.9917**	**0.9736**	**0.9601**	**1.0000**

Forrest [69]

Method	P@1	MRR	MAP	RA	AUC
MSE	0.1400	0.2010	0.1672	0.1117	0.7963
M-SSIM [37]	0.1625	0.2339	0.1920	0.1317	0.8264
CW-SSIM [42]	0.3550	0.4613	0.3273	0.2667	0.9114
STSIM [43]	0.9375	0.9597	0.9069	0.8692	0.9951
LBP [38]	0.9050	0.9328	0.8270	0.7767	0.9823
Zhang *et. al.* [70] (RV)	0.8375	0.8741	0.7370	0.6758	0.9687
Zhang *et. al.* [70] (RI)	0.7900	0.8387	0.6996	0.6333	0.9664
Ours [71]	**0.9725**	**0.9780**	**0.9313**	**0.8992**	**0.9966**

2.4 Similarity-based Retrieval of Subsurface Structures

In this section, we evaluate the introduced similarity measure in a image retrieval experiment of seismic images. In this experiment, we use `LANDMASS-2` [72] dataset which contains 4000 images of size 150×300 pixels extracted from the public Netherlands offshore F3 block dataset [5]. Each image in the dataset is manually labeled according to the dominant geological structure present in the image, i.e., `horizon`, `chaotic`, `faults`, or `salt`. Example images from the dataset are shown in Figure 2.14.



(a) `Horizon` (b) `Chaotic`

(c) `Fault` (d) `Salt`

Figure 2.14: Sample images from LANDMASS-2 [72] depicting the four subsurface structures in the dataset.

2.4.1 Experimental Procedure

Similar to the procedure detailed in Section 2.3, each image in the dataset is compared against all other images in the dataset using a similarity (or distance) measure. The images are then ordered according to their similarity to the reference image in descending order and the top 999 images are retrieved. Then, the retrieval performance

of the similarity measure is assessed by the number and order of correctly retrieved images.

In practice, we first form a 4000×4000 similarity matrix, M_{sim}, for the entire dataset where $M_{\text{sim}}[i, j]$ is the similarity between the i^{th} and j^{th} images, i.e., $M_{\text{sim}}[i, j] = S(\mathbf{X}_i, \mathbf{X}_j)$ for $1 \leq i, j \leq 4000$. Then, each row of M_{sim} is ordered in descending order. The top 999, excluding the reference image, are retrieved and are said to belong to the same class as that of the reference image.

2.4.2 Results and Discussion

Image retrieval is carried out using the proposed similarity measure in addition to other commonly used texture and seismic similarity (or distance) measures. These measures are:

- Mean Squared Error (MSE).
- Structural SIMilarity (SSIM) [37].
- Complex Wavelet Structural SIMilarity (CW-SSIM) [42].
- Structural Texture SIMilarity (STSIM) [43].
- Seismic SIMilarity (SeiSIM) [51].
- Curvelet-based seismic distance measure [52].

Table 2.3 shows a quantitative evaluation of these measures on LANDMASS-2. Note that we do not report first instance metrics such as **P@1** and **MRR** because the number of images that belong to a single class is large compared to the texture dataset. Furthermore, the high correlation and similarity of seismic images that are extracted from adjacent inline or crossline sections deem first instance metrics less informative than all-instances metrics such as **RA** and **MAP**.

Table 2.3: Quantitative evaluation of the proposed similarity measure on LANDMASS-2 dataset.

Method	MAP	RA	AUC
MSE	0.3944	0.3450	0.5145
M-SSIM [37]	0.4426	0.3222	0.5268
CW-SSIM [42]	.8062	0.7211	0.8584
STSIM [43]	0.9255	0.8668	0.9655
SeiSIM [51]	0.8860	0.8192	0.9450
Alaudah and AlRegib [52]	0.9494	0.8962	0.9776
Ours [71]	**0.9537**	**0.9105**	**0.9830**

The results in Table 2.3 indicate that the proposed measure can distinguish sub-surface structures in seismic images better than all other measures, including seismic similarity measures such as [51] and [52]. We also show ROC curves for all similarity measures used in the experiment in Figure 2.15.

To further analyze the retrieval results, we show in Figure 2.16 the precision after retrieving n images (P@n curve) for each of the four classes in the dataset using the proposed similarity measure. The cyan curve represents the `horizon` class, and it shows that the precision is 1 for all values of n. This is mainly due to the simplicity of the `horizon` structure. On the other hand, the curves of the other classes drop at different rates depending on the complexity of their structures. For example, `fault` images contain faults that are of different scales and dipping angles, which makes them harder to distinguish using texture only. Hence, the `fault` precision curve drops at a faster rate than those of other classes. The black curve is the average P@n for all four classes.

In addition, in order to further assess the discriminative power of the proposed similarity measure, we perform a clustering experiment using the similarity matrix M. In this experiment, the similarity matrix is used solely to learn a 3-dimensional

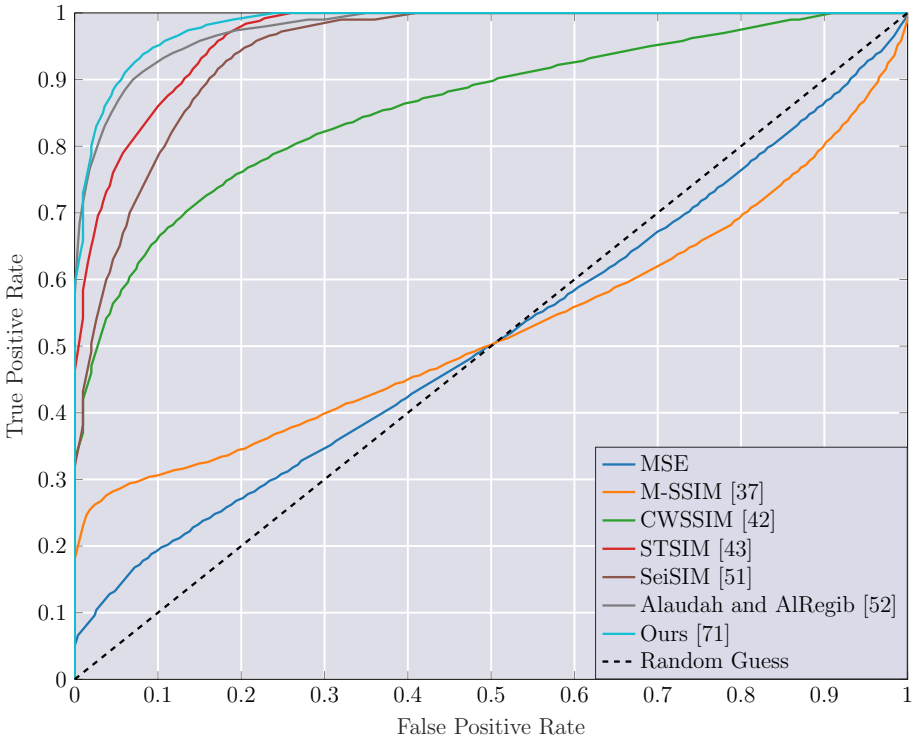


Figure 2.15: Receiver Operating Characteristic curves (ROC) for different similarity measures on LANDMASS-2 [72] dataset.

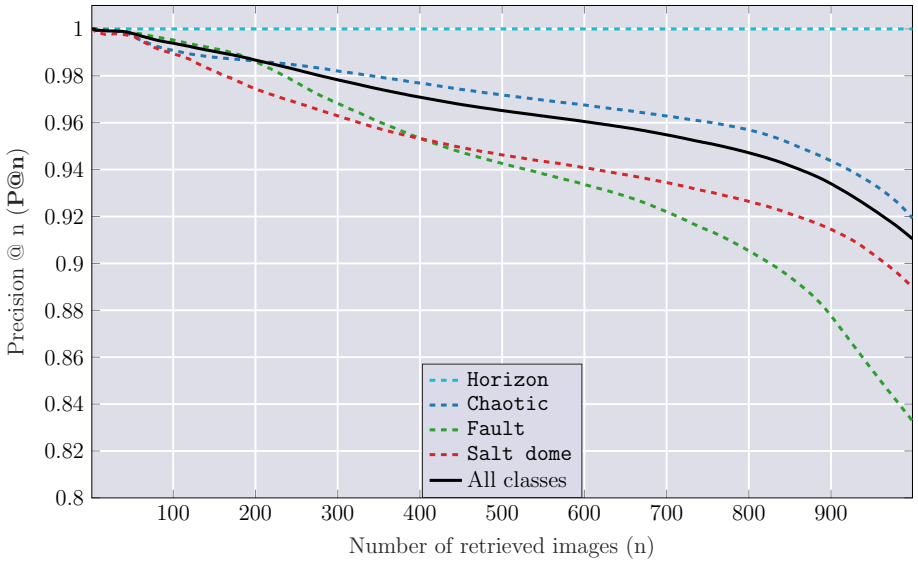


Figure 2.16: Precision at n curves for different subsurface structure classes.

representation for each of the images in the dataset. The learned 3-dimensional representation is then used to cluster the images into four unique clusters; a cluster for each of the classes. The quality of the clusters is then evaluated by comparing with the

ground-truth labels. Furthermore, visualization of the low dimensional representation space reveals more details about which classes were better captured by the similarity measure.

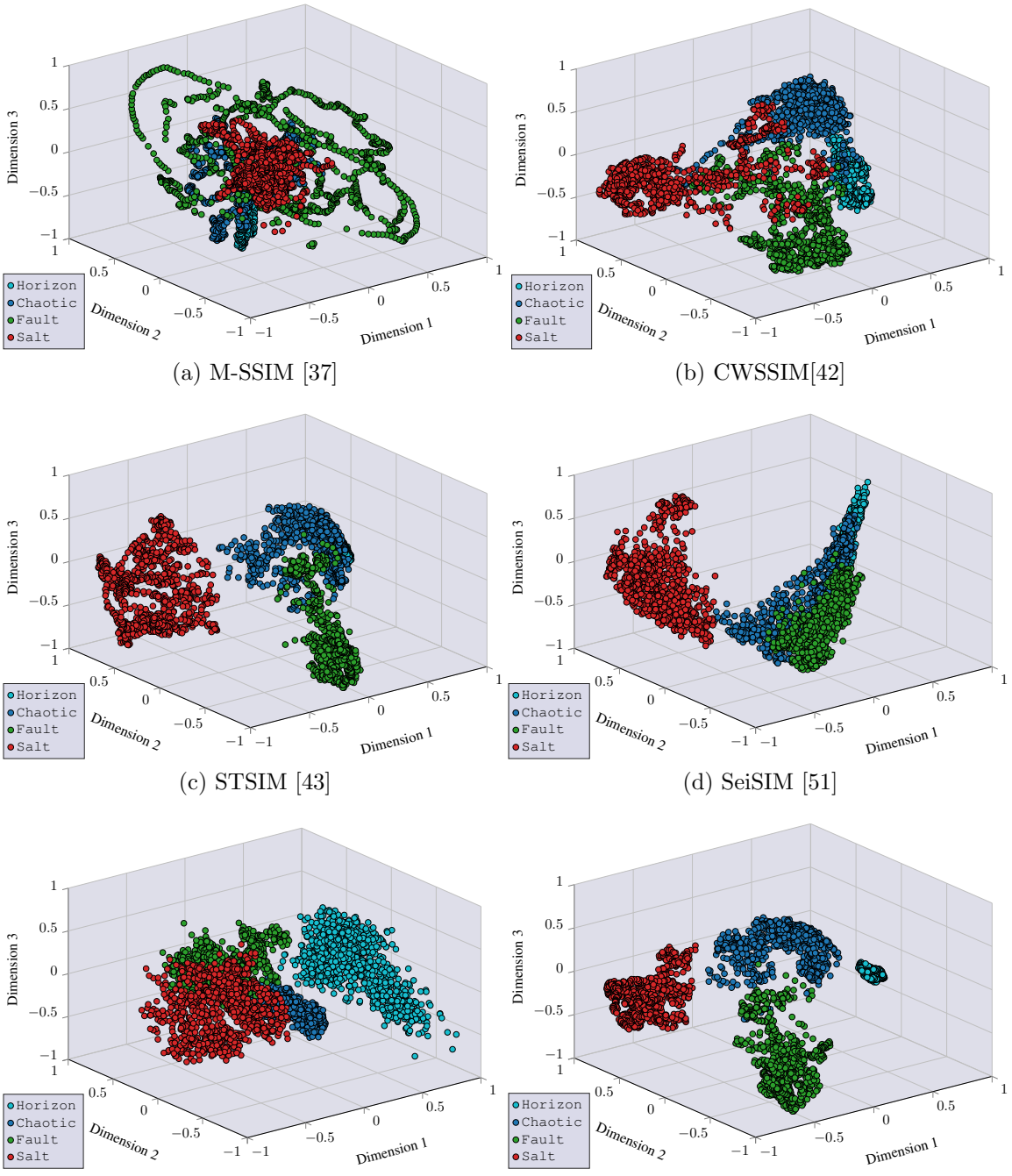
First, a 3-dimensional representation is learned for each image using Multi-dimensional Scaling (MDS) [73]. MDS is a method used to geometrically represent the images in the dataset using a single n-dimensional vector for each image. Images with the highest similarity score assigned by the similarity measure will be placed closer to each other in the n-dimensional space. MDS learns a low-dimensional representation such that the Euclidean distance between two points representing two images is inversely proportional to the similarity between the images,

$$\|\mathbf{x}_i - \mathbf{x}_j\|_2 \propto \frac{1}{\textsc{Sim}(X_i, X_j)}, \tag{2.5}$$

where $\mathbf{x}_i$ is the low-dimensional representation of the i^{th} image, $\mathbf{X}_i$.

Figure 2.17 shows the 3-dimensional representations of LANDMASS-2 dataset obtained using the different similarity measures. It is evident from the figure that M-SSIM does not group images of similar subsurface structures close to each other since it is a generic image similarity measure. On the other hand, texture similarity measures placed images that have similar structures within distinct clusters. However, the clusters for some classes might overlap which indicates that the similarity measure does not represent these classes very well. From a visual point of view, the low-dimensional representation obtained using our proposed similarity measure is the most distinctive out of all similarity measures used in this experiment.

In order to quantitatively evaluate the quality of the low-dimensional representation, we use k-means clustering algorithm to automatically cluster the images in the low-dimensional representation into four clusters. Then, we use Rand index, and the adjusted Rand index to evaluate the quality of these clusters with respect to the ground-truth labels of the images. Rand index is the percentage of pairs of images



(a) M-SSIM [37]

(b) CWSSIM[42]

(c) STSIM [43]

(d) SeiSIM [51]

(e) Alaudah and AlRegib [52]

(f) Ours [71]

Figure 2.17: Three-dimensional representation of LANDMASS-2 [72] dataset using MDS based on similarity values.

that have been correctly identified by the clustering algorithm to belong to the same cluster and the pairs that have been correctly identified to belong to different clusters. One issue with Rand index is that its expected value for two random partitions is

not constant. This issue is corrected with the adjusted Rand index. Both indices are between $[0, 1]$ with a value of 1 if and only if the clustering results match the ground-truth labels. More details about Rand index and Adjusted Rand index are discussed in Appendix A. The clustering results are summarized in Table 2.4.

Table 2.4: Evaluation of the quality of the clusters obtained by k-means clustering.

Method	Rand Index	Adjusted Rand Index
MSE	0.4254	0.0484
M-SSIM [37]	0.6114	0.0693
CW-SSIM [42]	0.9059	0.7510
STSIM [43]	0.9645	0.8779
SeiSIM [51]	0.8743	0.6665
Alaudah and AlRegib [52]	0.9067	0.7514
Ours [71]	**0.9771**	**0.9390**

2.5 Structure Characterization of Subsurface Volumes

In this section, we utilize the introduced multiresolution texture similarity measure to develop a large labeled dataset of seismic images. Given a few manually selected exemplars for each of the subsurface structures of interest, the introduced multiresolution texture similarity measure is used to populate a large labeled dataset of seismic images from an unlabeled seismic volume. The labeled dataset is then used to train a structure labeling workflow introduced in [74]. The objective of structure labeling is to identify and highlight the different subsurface structures present in the seismic volume. The structure labeling workflow is depicted in Figure 2.18.

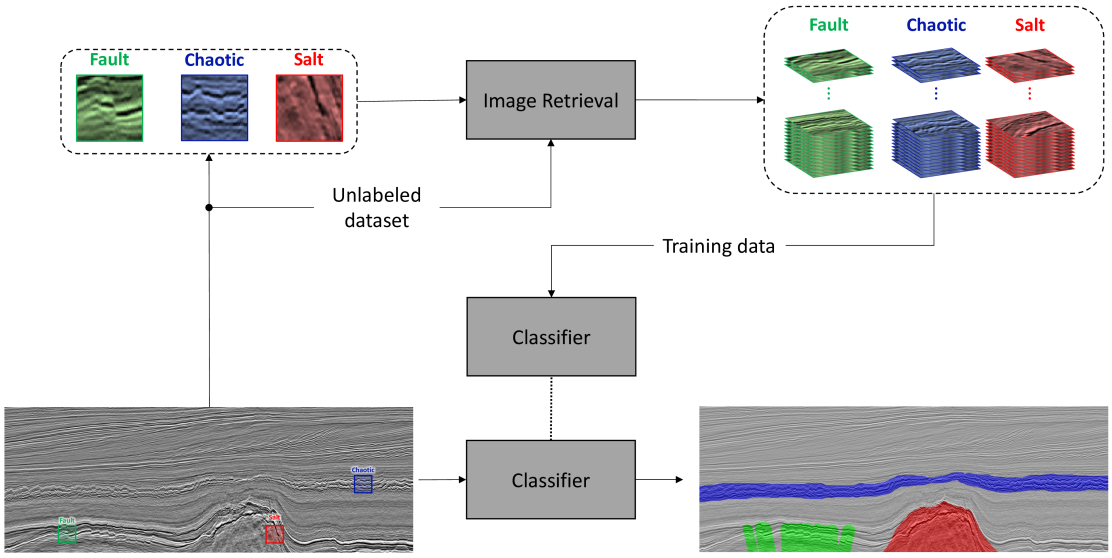


Figure 2.18: Structure labeling workflow using a few exemplar images for each of the structures of interest.

2.5.1 Experimental Procedure

We choose four subsurface structures of interest to be identified in the seismic volume: `chaotic`, `faults`, `salt`, and `other`. The `other` class serves the purpose of showing negative examples of structures that do not belong to any of the first three classes.

Then, using a single or a few exemplars for each subsurface structure, we retrieve 500 images for each class that have the highest similarity to the exemplar images as we have shown in Section 2.4. The outcome is a populated dataset of labeled seismic images, which is then used to train a classifier to label a large seismic volume. The trained classifier is then used to label the subsurface structures in a large seismic volume.

In this experiment, the structure labeling workflow using the introduced multiresolution similarity measure is applied on inline 381 of the Netherlands offshore F3 block [5]. Furthermore, we use different multiresolution image decomposition methods in the similarity measure to show the effectiveness of the curvelet features in capturing subsurface structures. Namely, instead of using the curvelet transform, we use the Gaussian pyramid [56], the discrete wavelet transform, and Gabor filters [63]. We also amplitude features of seismic images without multiresolution image decomposition as a baseline in the experiment.

2.5.2 Results and Discussion

The labeling results and a manual labeling of inline 381 are shown in Figure 2.19. A visual inspection of the results shows that the labeled section using our method is superior in terms of consistency. Moreover, the quality of the labeling is evaluated using objective measures that are commonly used in the semantic segmentation literature. These measures compare the labels obtained using the texture attributes with the labels obtained manually. These measures are Pixel Accuracy (PA), Intersection over Union (IoU) for each class, Mean Intersection over Union (MIoU) over all classes, and Frequency-Weighted Mean Intersection over Union (FW-MIoU). The details of these measures are presented in Appendix A. The quantitative results are reported in Table 2.5.

The results indicate that using the curvelet transform and effective singular values

Table 2.5: Quantitative evaluation of different multiresolution decomposition methods for subsurface structure characterization.

Method	PA	IoU				MIoU	FWIoU
		Other	Chaotic	Fault	Salt		
Amplitude	0.5571	0.4925	0.2295	0.1728	0.4264	0.3303	0.4492
Gaussian Pyramid	0.5639	0.4778	0.2425	0.2662	0.4463	0.3582	0.4442
Wavelet Transform	0.5691	0.4909	0.2627	0.2127	0.4195	0.3465	0.4524
Gabor filters	0.7594	0.7223	0.5335	0.2452	0.4265	0.4819	0.6643
Ours [75]	**0.8173**	**0.7807**	**0.5397**	**0.3373**	**0.5309**	**0.5472**	**0.7222**

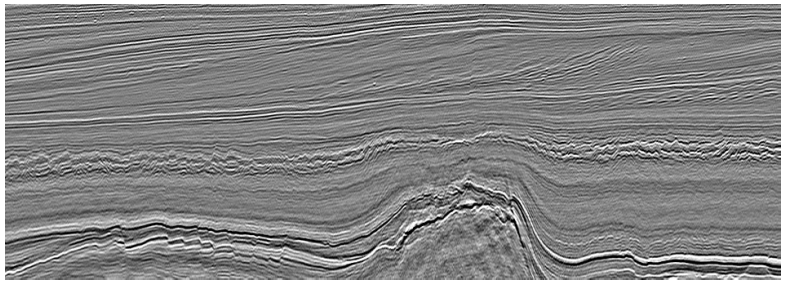
(as in the proposed similarity measure) is an effective method to characterize subsurface structures. Also, better labeling performance indicates that the dataset on which the classifier was trained describes the subsurface structures adequately. The results also suggest that directional features play a more important role than scale features for seismic image characterization. For example, the curvelet transform and Gabor filters performed the best as they are the only two decomposition methods with directional features. Moreover, the curvelet transform is superior to all other attributes on all metrics; mainly because of its effectiveness in representing curve-like features which constitute a large portion of the seismic section.

The IoU metric is computed for each class to provide more insight about the effectiveness of the proposed measure in characterizing each class. For example, IoU for faults indicates that faulty regions were not effectively labeled. However, this is not a shortcoming of the characterization method but rather of the nature of the labels that were used in the training. For example, a large number of fault images in the training dataset have a strong reflector which is a much more dominant feature than the faults themselves, making the classifier confuse images with strong reflectors as faults. Since every image is labeled with one label, the classifier assumes that all features present in the images belong to the same class, which is not true as we have seen in the case of faults and strong reflectors. This drives the need for fine-

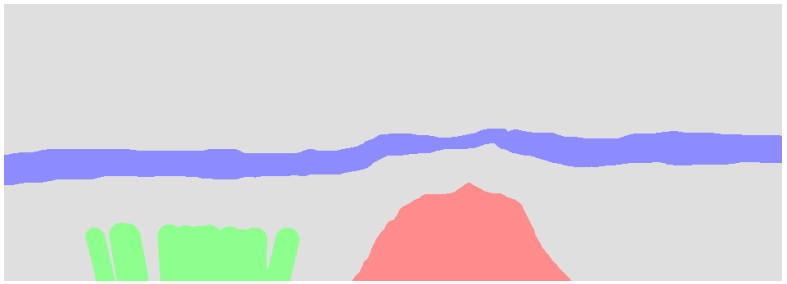
grained labels of the subsurface. An example of such labels are ones obtained from petrophysical studies of well-logs and core data. Such data is 1-dimensional but of a high vertical resolution. In the next chapter, we discuss the details of using such data as labels using sequence modeling.

2.6 Summary

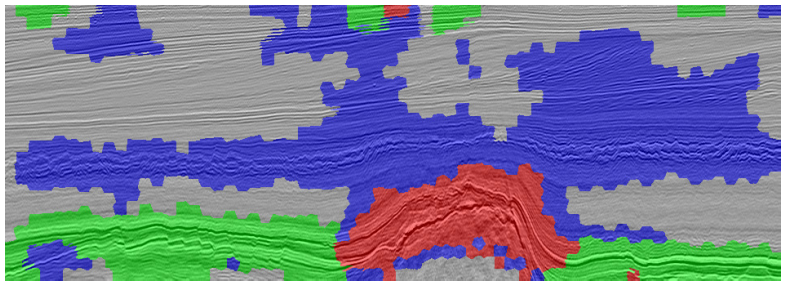
In this chapter, we introduced a multiresolution texture similarity measure based on the effective singular values of the curvelet transform. The introduced similarity was shown to outperform the state-of-the-art texture and seismic similarity measures. Furthermore, it was shown that the introduced similarity measure effectively represents seismic images in lower dimensional space such that different subsurface structures were grouped into well-separated clusters. The introduced measure was then used to form a large dataset of labeled seismic images from a few hand-labeled exemplars, each highlighting a subsurface structure of interest. The dataset is then used to train a classifier in a structure labeling framework. The trained classifier is then used to label different subsurface structures in a large unlabeled seismic volume. The developed multiresolution similarity measure, along with the structure labeling framework, were shown to be effective in segmenting the most dominant structures in the subsurface volumes such as salt bodies, faults, and chaotic structures. The identification of such structures enables field engineering to accurately place wells to be drilled in the survey area with minimal drilling hazards.



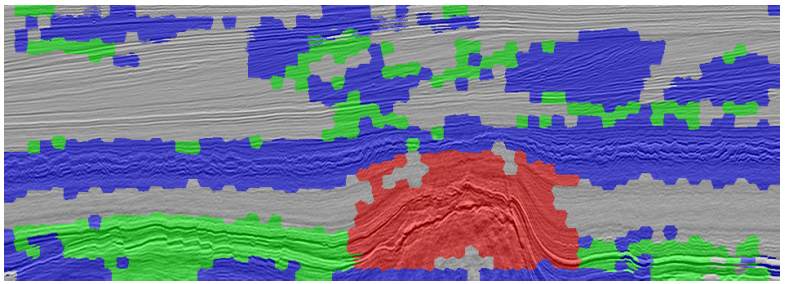
(a) Inline 381



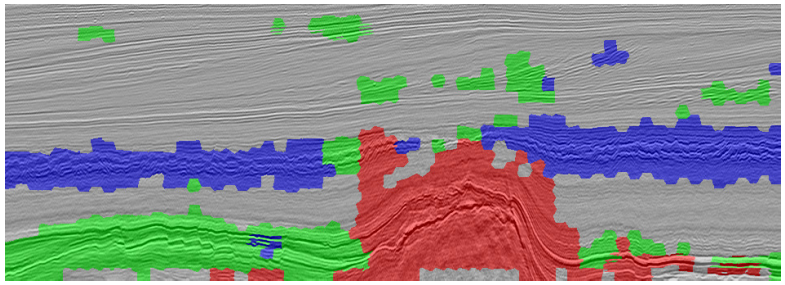
(b) Manual Labeling



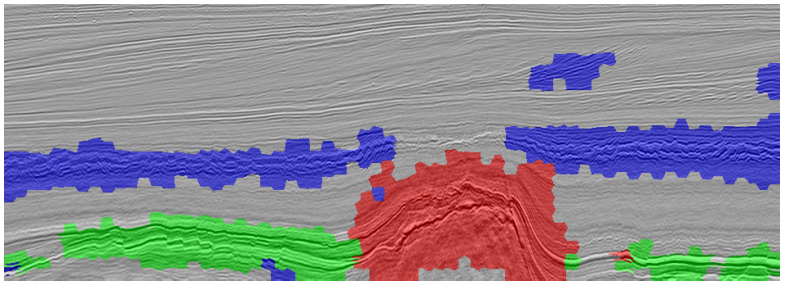
(c) Gaussian Pyramid



(d) Discrete Wavelet Transform



(e) Gabor Filters



(f) Curvelet Transform

Figure 2.19: Results of labeling inline 381 from the Netherlands F3 block using different multiresolution decomposition methods.

CHAPTER 3
SEQUENCE MODELING FOR LITHOLOGY CHARACTERIZATION

3.1 Overview

In Chapter 2, we introduced a multiresolution texture similarity measure that was used to create a large labeled dataset of seismic structures. The labeled dataset was then used to structurally label a seismic volume using seismic data only. The labeled volume can then be used for well-planning by drilling engineers to avoid drilling hazards. In this chapter, we present a framework that integrates data obtained from well-logs along with the seismic data to characterize the subsurface volume from a lithological point of view. In the lithology characterization framework, we use well-log data as labels for the seismic data around the wells. The proposed framework is trained on the pairs of seismic data and their labels from well-logs to learn a mapping function from seismic data to rock properties. Well-logs are 1-dimensional measurements of the lithology of the subsurface. To utilize this data in the characterization of subsurface volumes, we model seismic traces and well-log data as time series. We introduce a supervised sequence modeling workflow based on Gated Recurrent Units (GRUs) for lithology characterization of seismic data.

In Section 3.2.1, we present an overview of the literature of lithology characterization in addition to an overview of sequence modeling. In Section 3.3, we present the details of the proposed lithology characterization framework. Then, we evaluate the proposed framework in two case studies on real and synthetic datasets in Section 3.4 and 3.5, respectively. Finally, we present a summary of this chapter in Section 3.6.

3.2 Background

In this section, we define subsurface lithology characterization and present relevant literature. In addition, we present an overview of sequence modeling and their applications in the field of image and video processing.

3.2.1 Lithology Characterization

Lithology characterization is the process of estimating rock properties of the subsurface, such as density, porosity, and impedance, using seismic reflection data. It is worth noting that the target domain of lithology characterization is continuous (i.e., a range of possible rock property values) whereas the target domain of structure and stratigraphy characterization is discrete. Therefore, characterizing the subsurface lithologically provides a fine-grained characterization of the physical rock properties of the subsurface.

In general, the goal of lithology characterization is to infer true model parameters (rock properties, $\mathbf{m} \in X$) through an indirect set of measurements (seismic data, $\mathbf{d} \in Y$). Seismic data can be formulated mathematically as a function of subsurface properties of through a forward operator $\mathcal{F} : X \rightarrow Y$, i.e.,

$$d = \mathcal{F}(m) + n, \tag{3.1}$$

where $\mathbf{d}$ is the measured data (seismic), $\mathbf{m}$ is the true model (rock properties), and $n \in Y$ is a random variable that represents noise in the measurements.

To estimate the model parameters from the measured data, one needs to solve an inverse problem. The solution depends on the nature of the forward model and observed data. In the case of seismic inversion, and due to the non-linearity and heterogeneity of the subsurface, the inverse problem is highly ill-posed. In order to find a stable solution to an ill-posed problem, the problem is often regularized by

imposing constraints on the solution space and incorporating prior knowledge about the model.

The classical approach to solve the seismic inversion problem is to set up the problem as a Bayesian inference problem, and improve prior knowledge by optimizing for a cost function based on the data likelihood,

$$\hat{\mathbf{m}} = \underset{\mathbf{m}\in X}{\operatorname{argmin}} \left[\mathcal{H}\left(\mathcal{F}(\mathbf{m}), d\right) + \lambda \mathcal{C}(\mathbf{m})\right], \tag{3.2}$$

where $\hat{\mathbf{m}}$ is the estimated model parameters, $\mathcal{H} : Y \times Y \to \mathbb{R}$ is an affine transform of the data likelihood, $\mathcal{C} : X \to \mathbb{R}$ is a regularization function that incorporates prior knowledge, and $\lambda \in \mathbb{R}$ is a regularization parameter that controls the influence of the regularization function. Moreover, in this approach, a good initial guess of $\mathbf{m}$ is needed to ensure the convergence of the optimization routine to a valid solution. The literature of seismic inversion is rich in techniques to solve the inverse problem using Bayesian inference and various regularization techniques (e.g., [76, 77, 78, 79, 80, 81, 82]). More details about classical inversion are presented in Chapter 4.

Recently, there have been several successful applications of machine and deep learning methods in inverse problems in general [83], and to seismic inversion in specific [84, 85]. For example, seismic inversion has been attempted using supervised-learning algorithms such as Support Vector Regression (SVR) [86], artificial neural networks [87, 21, 22], committee models [88], Convolutional Neural Networks (CNNs) [20, 19], Bayesian learning [89], and transfer learning [90].

In general, machine learning algorithms can be used to invert seismic data by learning a non-linear inverse mapping parameterized by $\Theta \in Z$ ($\mathcal{F}^{\dagger}_{\Theta} : Y \to X$) using labels obtained from well-log data such that:

$$\mathcal{F}^{\dagger}_{\Theta}(\mathbf{d}) \approx \mathbf{m}. \tag{3.3}$$

Neural networks are then used to learn such inverse mapping (inversion network). The inversion network is learns by minimizing the following loss function over well-log data ($\mathbf{m}^*$), and their corresponding seismic data ($\mathbf{d}^*$):

$$\mathcal{L}(\Theta) := \mathcal{D}\left(\mathcal{F}_\Theta^\dagger(\mathbf{d}^*), \mathbf{m}^*\right), \tag{3.4}$$

where $\mathcal{D}$ is a distance measure to compare the estimated rock properties with the true rock property measurements from well-logs. The trained inversion network is then used to estimate rock properties from seismic data for the entire seismic survey, i.e., $\hat{\mathbf{m}} = \mathcal{F}_\Theta^\dagger(\mathbf{d})$.

There are many challenges that might prevent neural networks from finding a proper mapping that can be generalized beyond the training data. One of the challenges is the lack of data from a given survey area on which the network can be trained. For this reason, such neural networks must have a limited number of learnable parameters and a good regularization method in order to prevent over-fitting and to be able to generalize beyond the training data. In general, there are two common approaches to characterize seismic data in a learning framework. The first approach is to treat each data point of a seismic trace (in the z-direction) as an independent sample and try to estimate rock properties from the corresponding seismic data sample. This method fails to capture the global trends and temporal dependencies in seismic traces. An alternative approach is to treat each trace as a single training sample to incorporate global features by using the all points in the depth direction as features of a single sample. However, this approach severely limits the amount of data from which the algorithm can learn since each trace is treated as a single training sample. With a limited amount of training data, common machine learning algorithms might fail to generalize beyond the training data because of their high data requirements. Such a requirement might be difficult to meet in practical settings where the number of well-logs is limited. In order to remedy this, we propose modeling seismic traces

as sequential data via recurrent neural networks to capture the temporal dynamics of the traces and generalize better beyond the training data.

3.2.2 Sequence Modeling

A sequence is an ordered list of objects or data points. Thus, sequence modeling is used to represent data for which order is important. For example, sequence modeling is commonly used in developing language models by modeling a sentence as a set of ordered words that convey a specific meaning. Sequence modeling has been utilized widely to represent various types of data such as speech and audio signals [91], videos [92], and seismic traces [93]. Sequence modeling has been often attempted using Hidden Markov Models (HMMs) by modeling the probabilities of observing one output given the current observation using a set of discrete states. Recent advances in machine learning extended the concept of HMMs using Recurrent Neural Networks (RNN).

Hidden Markov Models

One of the applications of sequence modeling is sequence prediction or completion where the task is to predict the next data point of a sequence given the previously observed data points. The task is classically modeled using statistical sequence models such as HMMs [94]. For instance, for a sequence $\{x^{(1)}, x^{(2)}, \dots\}$, and a finite set of discrete states $\mathcal{S}$, where $x^{(i)} \in \mathcal{S}$, an n^{th}-order HMM assumes that the next point of the sequence depends only on the past n points of the sequence, i.e.,

$$P(x^{(t)}|x^{(t-1)}, x^{(t-2)}, \dots, x^{(1)}) = P(x^{(t)}|x^{(t-1)}, x^{(t-2)}, \dots, x^{(t-n)}), \text{ for } n < t-1, \quad (3.5)$$

where $P(x^{(t)}|x^{(t-1)}, x^{(t-2)}, \dots, x^{(t-n)})$ denotes the probability of observing $x^{(t)}$ given that $x^{(t-1)}, x^{(t-2)}, \dots,$ and $x^{(t-n)}$ have been observed. Then, the next point of the se-

quence is predicted as the point that has the maximum probability of being observed. Sequence prediction has been used for various applications such as text generation [95], traffic observation [96], and text translation [97]. Furthermore, HMMs have been utilized in the seismic domain for various applications. For example, Lomnitz [98] used HMMs to predict earthquakes from past observations by modeling data recorder by seismographs as time series. Godfrey *et al.* [99] utilized HMMs to model seismic impedance using a discrete set of states obtained from well-log data. Amin *et al.* [100] used HMMs to model the presence of salt bodies within subsurface volumes and used the Expectation Maximization (EM) algorithm to delineate the edges of salt bodies.

Hidden Markov models are known as fixed-window models where the window size refers to the order of the Markov model. Theoretically, HMMs are capable of modeling a sequence of an arbitrary length given a sufficiently large window. However, the number of states (the size of the set S) grows exponentially with the order of the HMM. Such limitation makes it difficult to model long-term interactions in sequences. Figure 3.1 shows three examples of state transition diagrams of $\mathcal{S}$ with different values of n.

Recurrent Neural Networks

Recurrent Neural Networks (RNNs) [101] are a class of artificial neural networks that are designed to capture the temporal dynamics of sequential data. Unlike feed-forward neural networks, RNNs have a hidden state variable that can be passed between sequence samples which allows them to capture long temporal dependencies in sequential data. Figure 3.2 shows a side-by-side comparison between a feed-forward unit and a recurrent unit. Furthermore, RNNs overcome the exponential growth of the number of states in HMMs by introducing a continuous state variable that can represent an arbitrary number of states.

RNNs are widely used for sequence-to-sequence mapping where the task is to

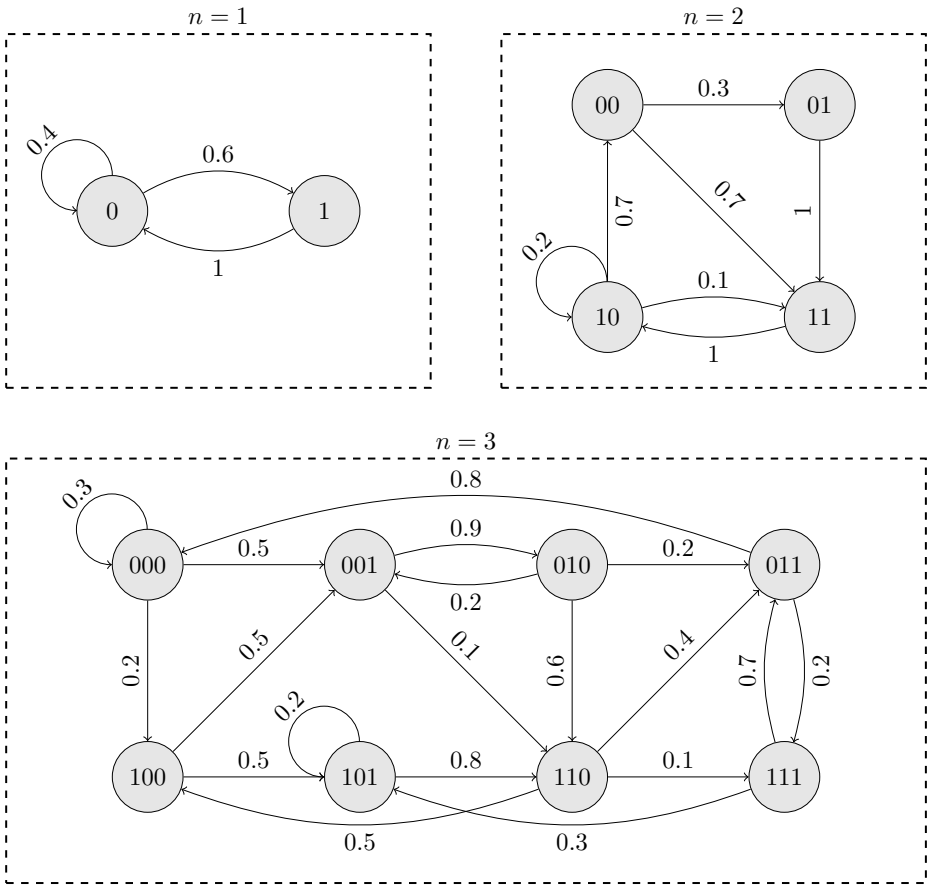


Figure 3.1: Examples of states transition diagrams for different orders of HMM (n). The numbers on the arrows represent the transition probability.

predict one sequence $\mathbf{x} \in X$, $\mathbf{x} = \left[x^{(1)}, x^{(2)}, \ldots, x^{(T_x)}\right]$ from another sequence $\mathbf{y} \in Y$, $\mathbf{y} = \left[y^{(1)}, y^{(2)}, \ldots, y^{(T_y)}\right]$. In sequence-to-sequence mapping, the objective is to find a mapping function $\mathcal{F} : X \rightarrow Y$ such that:

$$y^{(t)} = \mathcal{F}(x^{(t)}, y^{(1)}, y^{(2)}, \ldots, y^{(t-1)}). \tag{3.6}$$

Note that sequence-to-sequence mapping reduces to sequence prediction when $\mathbf{x} = \{x^{(1)}, x^{(2)}, \ldots, x^{(t-1)}\}$ and $\mathbf{y} = x^{(t)}$. RNNs have been utilized widely for various sequence modeling tasks such as language modeling and natural language processing (NLP) [102], speech and audio processing [103], and activity recognition in videos [104]. In addition, RNNs have been utilized in the seismic domain to model long-

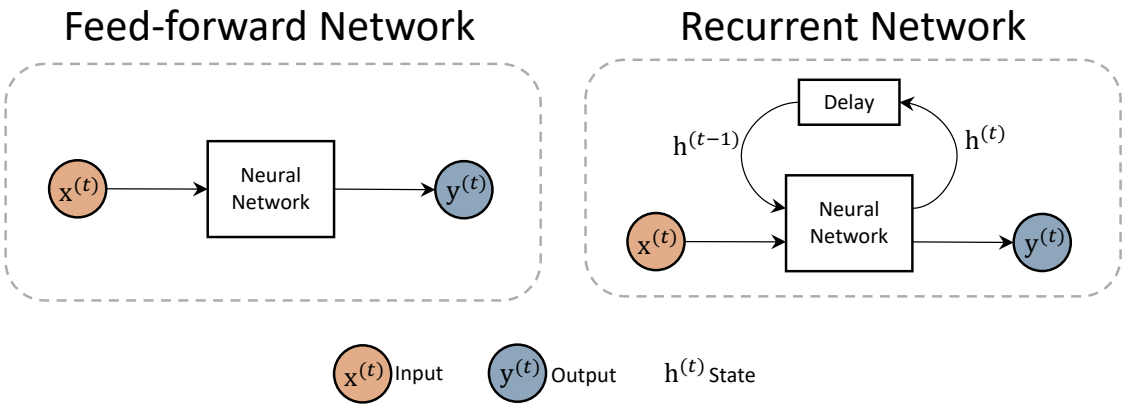


Figure 3.2: An illustration of feed-forward and recurrent networks.

term interactions in seismic signals. For example, Wiszniowski *et al.* [105] used real-time RNNs to detect small natural events in recorded seismic signals. Biswas *et al.* [106] used RNNs to estimate stacking velocity from pre-stack 2D data by training the network on a subset of the seismic traces using a given stacking velocity model. The network is then used to estimate the stacking velocity of the remaining data. Despite the success of RNNs in language, audio, and video domains, they have not been widely utilized in the seismic domain, especially in subsurface characterization.

When RNNs were proposed in the 1980s, they were difficult to train because they introduced temporal dependency which made gradients more difficult to compute. The problem was later solved using backpropagation through time (BPTT) algorithms [107], which turn the gradient into a long product of terms using the chain rule. Theoretically, RNNs are capable of learning long-term dependencies through their hidden state variables. However, even with BPTT, RNNs fail to learn long-term dependencies mainly because the gradients tend to vanish for long sequences as they are backpropagated through time. New RNN architectures have been proposed to overcome this issue using gated units. Examples of such architectures are Long Short-Term Memory (LSTM) [108] and Gated Recurrent Units (GRU) [109]. Such architectures have been shown to capture long-term dependency and perform well for various sequence modeling tasks.

GRUs improve on the simple RNN by introducing reset-gate and update-gate variables which are internal states that are used to evaluate the long-term dependency and keep information from previous time steps only if they are needed. A forward step through a GRU is given by the following equations:

$$\begin{aligned}
\mathbf{u}_i^{(t)} &= \text{sigmoid}\left(\mathbf{W}_{xu}\mathbf{x}_i^{(t)} + \mathbf{W}_{yu}\mathbf{y}_i^{(t-1)} + \mathbf{b}_u\right) \\
\mathbf{r}_i^{(t)} &= \text{sigmoid}\left(\mathbf{W}_{xr}\mathbf{x}_i^{(t)} + \mathbf{W}_{yr}\mathbf{y}_i^{(t-1)} + \mathbf{b}_r\right) \\
\hat{\mathbf{y}}_i^{(t)} &= \tanh\left(\mathbf{W}_{x\hat{y}}\mathbf{x}_i^{(t)} + \mathbf{b}_{\hat{y}_1} + r_i^{(t)} \circ \left(\mathbf{W}_{h\hat{y}}\mathbf{y}_i^{(t-1)} + \mathbf{b}_{\hat{y}_2}\right)\right) \\
\mathbf{y}_i^{(t)} &= (1 - \mathbf{u}^{(t)}) \circ \mathbf{y}_i^{(t-1)} + \mathbf{u}^{(t)} \circ \hat{\mathbf{y}}^{(t)},
\end{aligned} \tag{3.7}$$

where $\mathbf{x}_i^{(t)}$ and $\mathbf{y}_i^{(t)}$ are the i^{th} input sequence and its estimated output at time t, respectively, $\mathbf{u}_i^{(t)}$ and $\mathbf{r}_i^{(t)}$ are the update-gate and reset-gate vectors, respectively, $\hat{\mathbf{y}}_i^{(t)}$ is the candidate output for the current time step, $\mathbf{W}$'s and $\mathbf{b}$'s are the learnable parameters, and the operator $\circ$ is the element-wise product. Figure 3.3 shows a GRU network unfolded through time. Note that all GRUs in an unfolded network share the same $\mathbf{W}$ and $\mathbf{u}$ parameters.

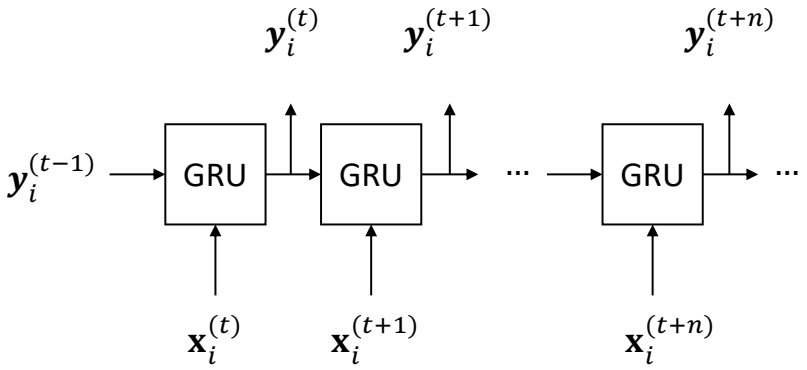


Figure 3.3: Gated Recurrent Unit (GRU) unfolded through time.

3.3 Proposed Method

Using neural networks, we can learn the inverse mapping ($\mathcal{F}_{\Theta}^{\dagger}$ in equation (3.3)) by setting up the inversion problem as a supervised sequence-to-sequence. The neural network (inversion network) is given a set of measurement-model pairs $(\mathbf{d}^*, \mathbf{m}^*)$ (e.g., seismic traces and their corresponding property traces from well-logs) to learn the inverse mapping by minimizing the loss function in (3.4).

In [110], we proposed a shallow sequence modeling network based on GRUs for property estimation from seismic data. The network consists of two convolutional layers to extract local features followed by 2-layer GRU to extract sequential features. Then, sequential features are mapped to rock property using a simple linear layer. The proposed network is trained on pairs of property traces and their corresponding seismic traces. Then, the trained network is used to estimate the property for all traces in the survey. The proposed network is shown in Figure 3.4.

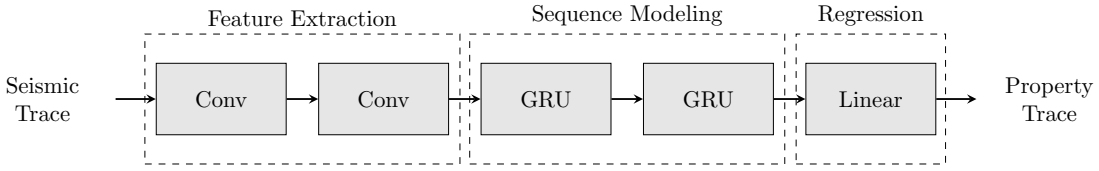


Figure 3.4: The architecture of the inversion network in the proposed framework.

The network in Figure 3.4 consists of three main modules. The first module is the *Feature extraction* module which is a series of two convolutional blocks (Conv) with small kernels size. A convolutional block consists of a convolutional layer followed by a batch normalization layer and a non-linearity. This module extracts local features from the input seismic traces. The next module is the *sequence modeling* which consists of two in-series GRUs. The *Sequence modeling* module computed global features from the input traces using its memory element. Finally, the computed features are fed to the *Regression* module which is a linear layer that maps these features to the output domain, i.e., the rock property domain.

The network is trained on pairs of seismic traces and their corresponding lithology traces from well-logs. We use the Mean Square Error (MSE) as a distance measure to compute the error between the estimated traces and the true traces. Hence, (3.4) reduces to:

$$\mathcal{L}(\Theta) = \frac{1}{N^*} \sum_i \|\mathcal{F}_\Theta^\dagger(\mathbf{d}_i^*) - \mathbf{m}_i^*\|_2^2, \tag{3.8}$$

where N^* is the number of training samples.

3.4 Case Study: Density Estimation

We evaluate the proposed supervised sequence modeling lithology characterization framework on data from Netherlands offshore F3 block [5]. The dataset contains seismic data, and 4 wells: `F02{1`, `F03{2`, `F03{4`, and `F06{1`. The locations of these wells within the seismic survey is shown in Figure 3.5.

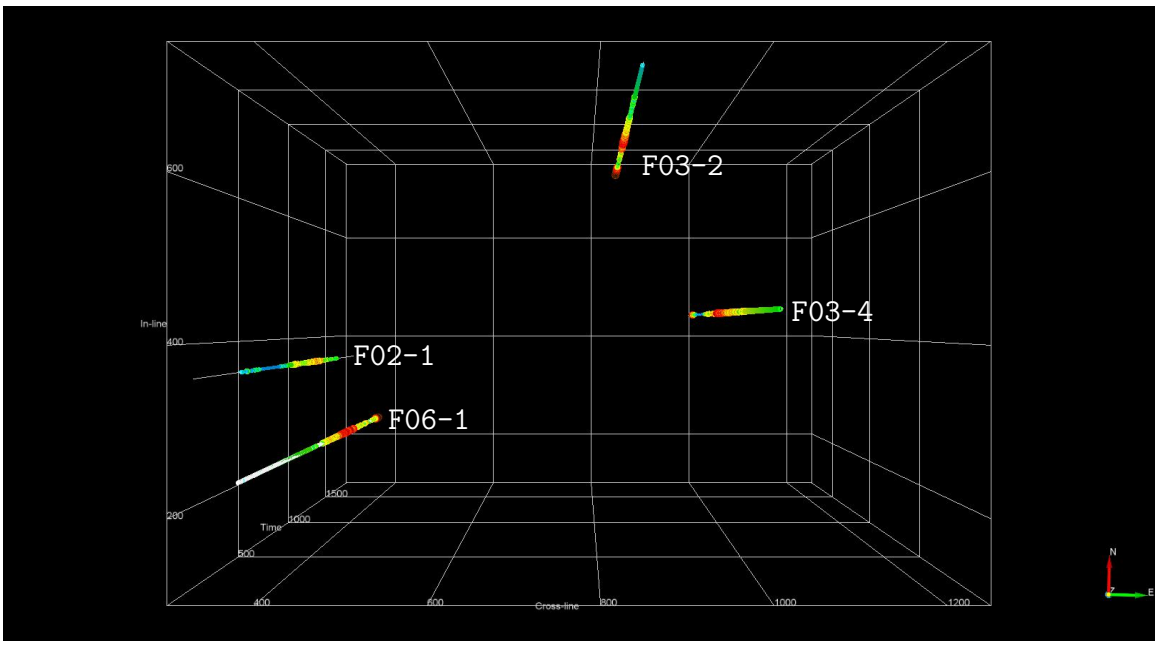


Figure 3.5: A top view of the seismic survey area showing the locations of the wells used in the density estimation experiment.

For each of the wells, we extracted a cube of seismic data of size [$7 \times 7 \times$ depth]

centered at the well. The proposed workflow is then trained using seismic cubes as inputs and the density traces from well-logs as labels. It is worth noting that a problem as difficult as property estimation might need a more complex and deeper learning model; however, the number of model parameters increases with complexity and thus much more data is required to train such models properly. The goal of this experiment was to show the power of recurrent neural networks for property estimation by utilizing their temporal dependencies.

3.4.1 Data Preprocessing

Since the well-log data is captured at a higher resolution than that of the seismic data, we first smooth then downsample the well-log data to match the resolution of the seismic data. We also normalize the seismic data by subtracting the mean and dividing by the standard deviation.

3.4.2 Training the Network

A seismic trace is fed to the network to obtain an estimated property trace. The error is then competed between the estimated property traces and the actual property traces from well-logs. The gradients of the error, with respect to the parameters of the network, are computed. Finally, the gradients are used to update the network parameters. This process is repeated until a stopping criterion is met.

Due to the small size of the dataset, training regularization is needed to ensure that the model does not over-fit to the training data. We used data augmentation by rotating the seismic cubes along the time axis to increase the number of training samples. Furthermore, we used early stopping where the training is stopped after a small number of epochs. More training epochs will definitely improve the performance on the training dataset, but the network will fail to generalize. In our experiments, the training is stopped when the training correlation reaches 0.97.

3.4.3 Results and Discussion

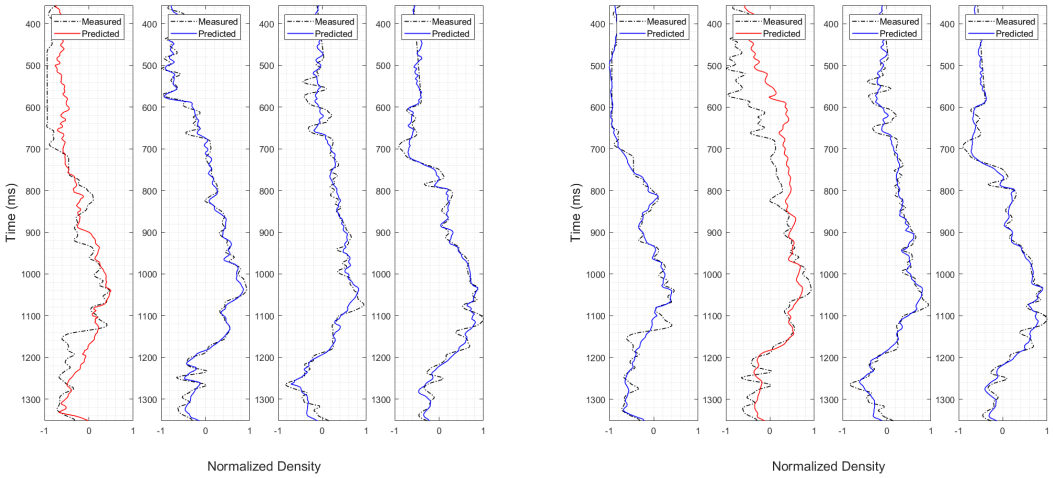
The performance of the models is assessed using 4-fold validation, where 3 of the wells are used for training and the remaining well is used for testing. The process is repeated 4 times, and the results are averaged for all experiments. The estimated density traces are shown in Figure 3.6. Estimated density traces for the validation well is shown red, estimated density traces for the training wells are shown in blue, and the true density traces from well-logs are shown in black.

To quantitatively evaluate the estimated traces, we compute two goodness-of-fit metrics that are commonly used in regression analysis. Namely, we compute the Pearson Correlation Coefficient (**PCC**), and the coefficient of determination ($\mathbf{r}^2$). **PCC** is a metric between 0 and 1 that measures the correlation between the estimated and true traces that takes into account the global shapes of the traces. The coefficient of determination is a more strict measure than **PCC** as it takes into account the local variations in the traces. It gives a value in $(-\infty, 1]$ with a higher value indicating a better fit. More details about **PCC** and $\mathbf{r}^2$ are presented in Appendix A. The results are summarized in Table 3.1.

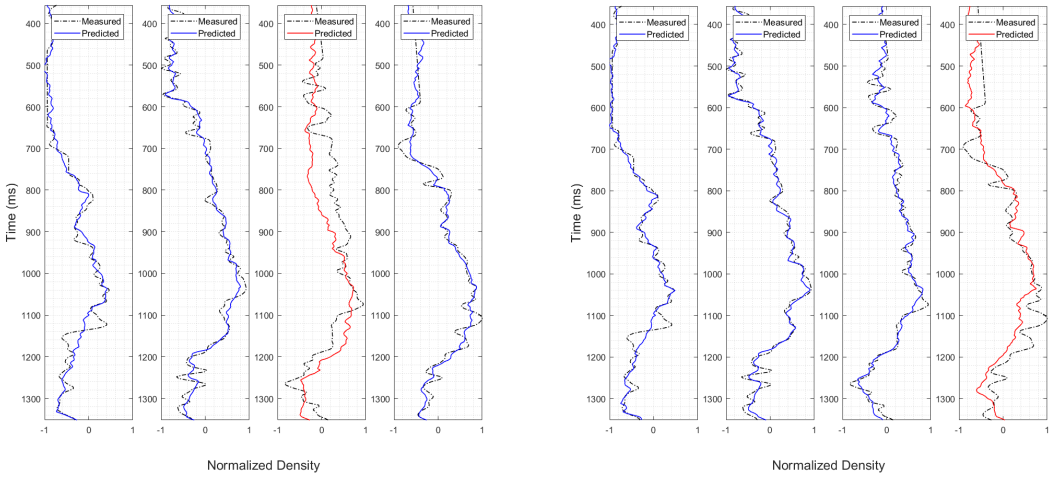
Training wells	Validation well	**PCC**		$\mathbf{r}^2$	
		Training	Validation	Training	Validation
F03–2, F03–4, F06–1	F02–1	0.97	0.88	0.94	0.63
F02–1, F03–4, F06–1	F03–2	0.97	0.86	0.94	0.55
F02–1, F03–2, F06–1	F03–4	0.97	0.72	0.94	0.32
F02–1, F03–2, F03–4	F06–1	0.97	0.90	0.94	0.75

Table 3.1: Quantitative Evaluation of the estimated density traces using the proposed network.

The results of this case study show that the network was able to learn a mapping from seismic data to density using only 3 well-log traces for training. With such a small dataset of real data, an average correlation of 0.90 indicates that the estimated density follows the low-frequency trends in the true density. Moreover, an average $\mathbf{r}^2$



(a) 1st fold (b) 2nd fold



(c) 3rd fold (d) 4th fold

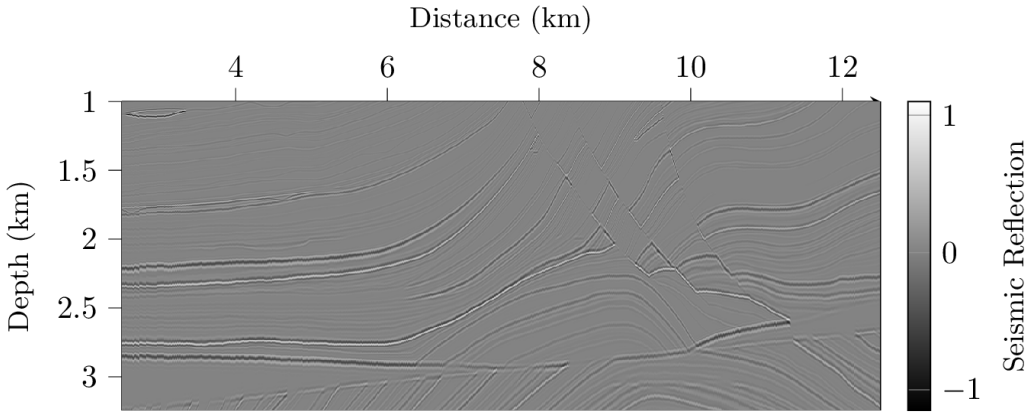
Figure 3.6: Cross-validation results for density estimation from migrated seismic data. Training traces are shown in blue, validations traces are shown in red, and actual measurements are shown in black.

also indicated that the small variations also match between the estimated and true traces. However, note that large variability in results between the different folds of the experiments. This is mainly due to the locations of the wells within the survey as shown in Figure 3.5. For instance, the wells `F02-1` and `F06-2` are close in distance which makes them have highly similar characteristics. Thus, when one of them is used for training, and the other is used for validation, the results are generally higher.

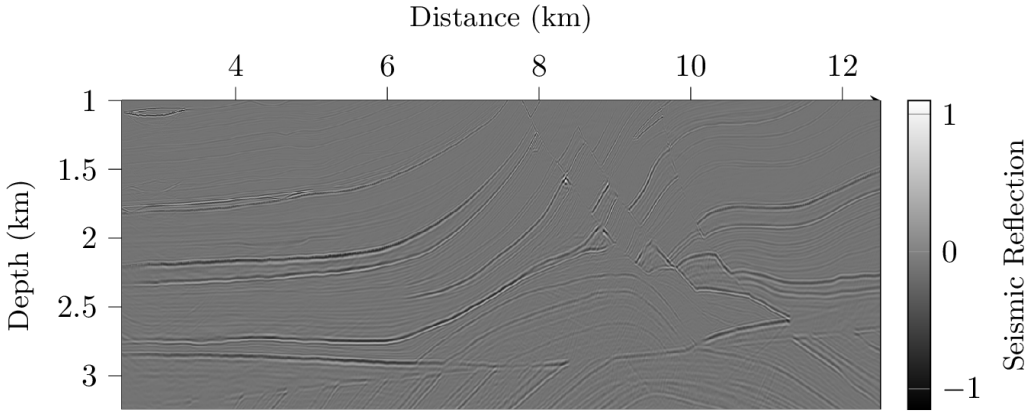
3.5 Case Study: Acoustic Impedance Estimation

In the previous case study, we used real data to evaluate the performance of the proposed workflow. Although real data is more challenging to characterize due to noise and migration artifacts, we do not have a means of evaluating the generalization power of the proposed workflow beyond areas near the wells. Therefore, in this case study, we evaluate the performance of the proposed workflow on a synthetic dataset for which the true lithology is available throughout the survey. In particular, we use Marmousi 2 synthetic model [23] which is an elastic upgrade of the original Marmousi model [111]. The original Marmousi model is based on a profile through the North Quenguela Trough in the Cuanza Basin, Angola. It has been used for numerous studies in geophysics for various applications including seismic inversion, seismic modeling, and seismic imaging. Marmousi 2 spans 17 km in width and 3.5 km in depth with a vertical resolution of 1.25 m. Using a synthetic model, we can quantitatively evaluate the quality of the estimated rock properties by comparing with the ground-truth synthetic model.

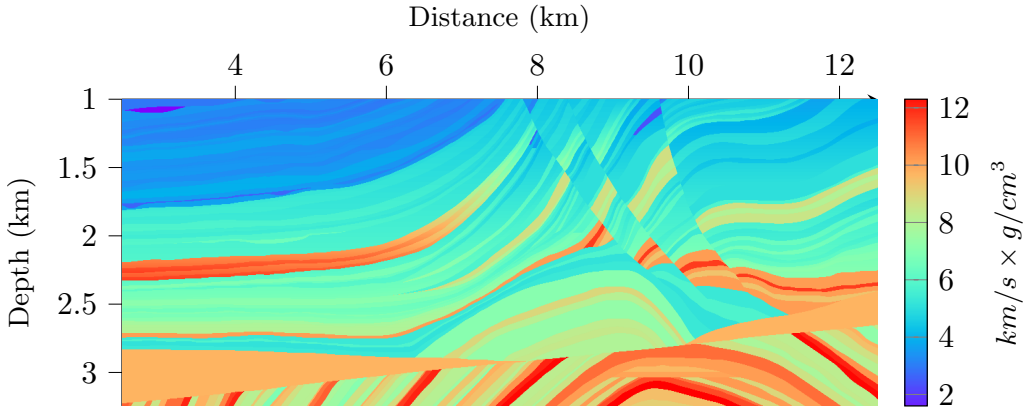
In this case study, we estimate acoustic impedance from noise-free zero-offset synthetic seismic data. We chose 10 uniformly-spaced acoustic impedance traces to serve as pseudo-wells. For each pseudo-well, a seismic trace is extracted, and the acoustic impedance value from the acoustic model is used as a label for the seismic trace. Data preprocessing and network training is carried as detailed in Section 3.4. In addition, we train the network, independently, on wave-equation-migrated seismic data to test the robustness of the proposed workflow to migration artifacts that are common in field seismic data. The synthetic seismic section, migrated seismic section, and the corresponding acoustic impedance section are shown in Figure 3.7.



(a) Noise-free synthetic zero-offset seismic



(b) Wave-equation migrated seismic



(c) Acoustic impedance

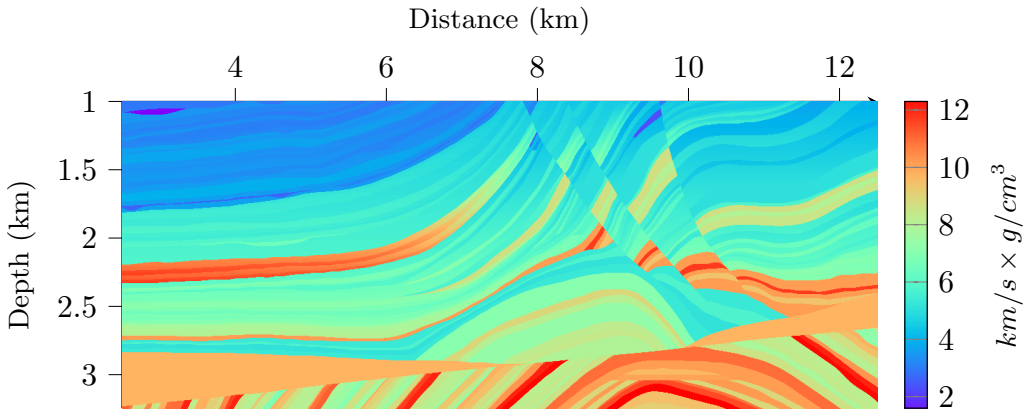
Figure 3.7: A Synthetic seismic section, a wave-equation-migrated seismic section from Marmousi 2 [23], and the true acoustic impedance section (in depth).

3.5.1 Results and Discussion

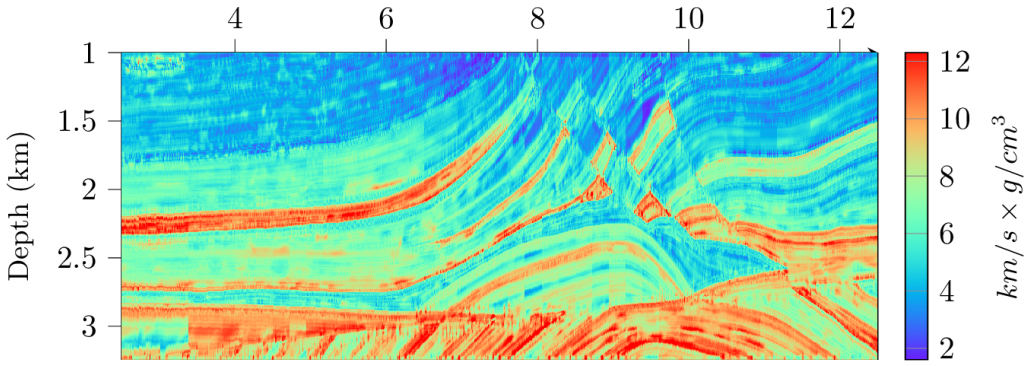
The proposed supervised sequence modeling workflow is trained on 10 pseudo-well-logs ($N^* = 10$) and their corresponding seismic data. Then, the trained model is used to estimate acoustic impedance for all traces in Marmousi 2 model. In addition, to show the power of recurrent neural networks compared to feed-forward ones, we train a convolutional network, similar to the one in Figure 3.4 where recurrent layers are replaced by convolutional layers with the same number of parameters. The estimated acoustic impedance profiles from noise-free synthetic seismic data using the two networks are shown in Figure 3.8.

The results in Figures 3.8 show the estimation of acoustic impedance using synthetic noise-free seismic data. With noise-free synthetic seismic, all reflectors are clearly visible which makes the estimation task relatively easier. However, in practice, seismic data is often noisy and has migration artifacts which make the reflectors less visible, especially in deep and complex geology. To evaluate the performance of the proposed network on migrated data, we repeat the same experiment using wave-equation-migrated seismic data instead of the synthetic seismic data to estimate acoustic impedance. The results are shown in Figure 3.9.

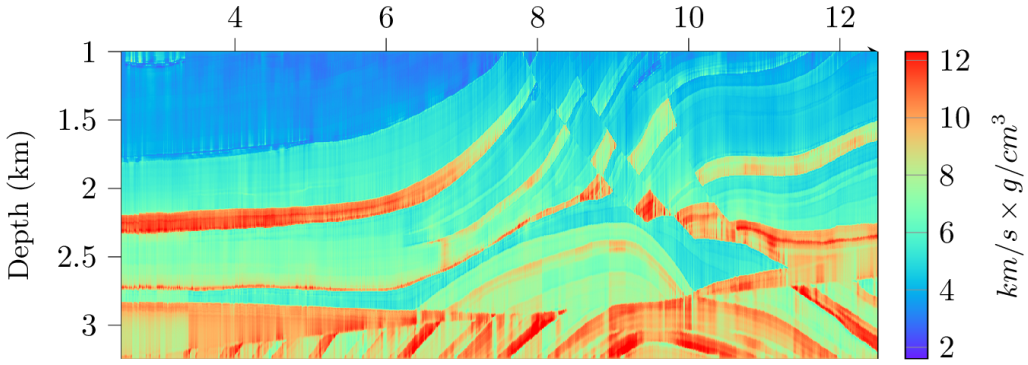
Visual inspection of the estimated acoustic impedance profile in Figure 3.8 shows the superiority of the recurrent neural network to the convolutional neural network in lithology characterization. This is clearly visible from the smoothness of the estimated acoustic impedance using the recurrent neural network compared to those estimated using the convolutional neural network. It is interesting that the recurrent network produces a laterally-smooth estimation although the estimation was performed on a trace-by-trace manner. On the other hand, the Convolutional network was able to estimate the acoustic impedance in general but the results are noisy although the input seismic was noise-free. The estimated acoustic impedance section using recurrent layers is much more consistent and less noisy compared to the estimated acoustic



(a) True acoustic impedance



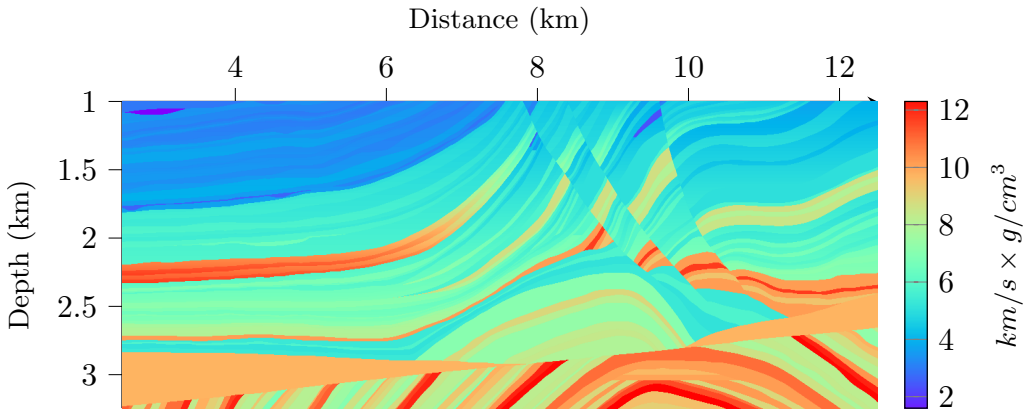
(b) Estimated acoustic impedance using using a convolutional network



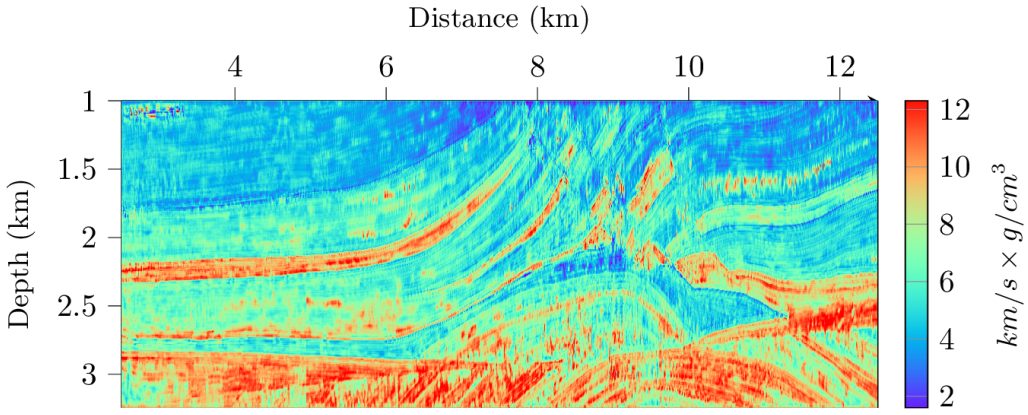
(c) Estimated acoustic impedance using a recurrent network

Figure 3.8: Estimated acoustic impedance from noise-free synthetic seismic data using a convolutional network and a recurrent network.

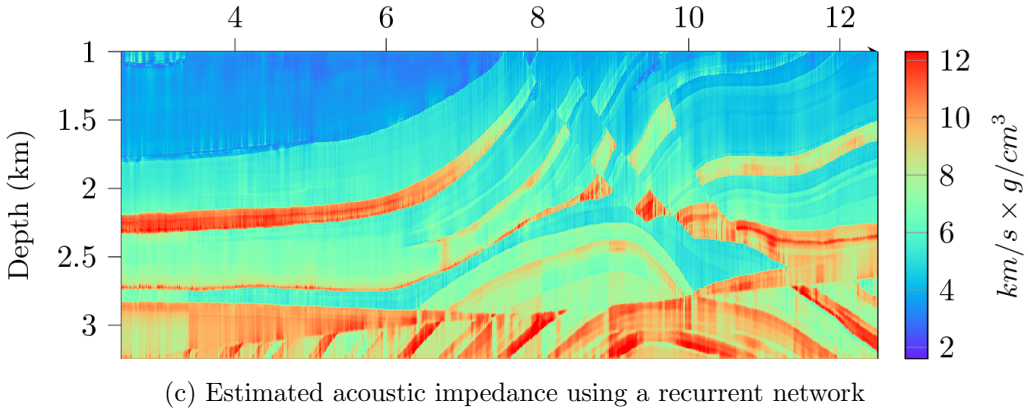
impedance section using the feed-forward network, especially in thick layers where state variables help in maintaining the estimated value. This indicates that global features captured by the recurrent layers are important for such a task. This feature of



(a) True acoustic impedance



(b) Estimated acoustic impedance using using a convolutional network



(c) Estimated acoustic impedance using a recurrent network

Figure 3.9: Estimated acoustic impedance from wave-equation-migrated seismic data using a convolutional network and a recurrent network.

recurrent neural networks is more visible in Figure 3.9 when wave-equation-migrated seismic are used to estimate acoustic impedance instead of noise-free synthetics.

To further analyze the results, the estimated acoustic impedance profiles are com-

pared quantitatively to the true acoustic impedance using Pearson Correlation Coefficient (**PCC**) and the coefficient of determination ($\mathbf{r}^2$). The quantitative results are summarized in Table 3.2 for noise-free synthetic seismic data and in Table 3.3 for wave-equation-migrated seismic data.

Table 3.2: Quantitative evaluation of the estimated acoustic impedance from noise-free synthetic seismic using recurrent and feed-forward networks.

	PCC		$\mathbf{r}^2$	
Network	Training	Validation	Training	Validation
Feed-forward	0.9829	0.9253	0.9650	0.7778
Recurrent	0.9940	**0.9728**	0.9876	**0.9249**

Table 3.3: Quantitative evaluation of the estimated acoustic impedance from wave-equation-migrated seismic using recurrent and feed-forward networks.

	PCC		$\mathbf{r}^2$	
Network	Training	Validation	Training	Validation
Feed-forward	0.9776	0.8600	0.9513	0.6345
Recurrent	0.9887	**0.9363**	0.9774	**0.8233**

The results in Table 3.2 show that using sequence modeling improved the estimation of acoustic impedance by 5% on **PCC** and 15% on $\mathbf{r}^2$ in the noise-free synthetic case. Moreover, acoustic improved by 6%, 19% on **PCC** and $\mathbf{r}^2$ for the wave-equation-migrated case. Furthermore, a smaller gap between the training and validation results using a recurrent network indicates a better generalization ability of the network. Although the performance of both networks has dropped significantly when estimating acoustic impedance from wave-equation-migrated seismic data, the recurrent network maintains the superior performance compared to the feed-forward network with even larger margins on both metrics. The lower performance is excepted because of the migration artifacts present in the wave-equation-migrated seismic which prevent the network from resolving all reflectors. Although using a deeper network can help improve the resolution of the estimated acoustic impedance, deeper networks cannot be

trained with a limited number of labeled data in a supervised-learning scheme.

3.6 Summary

In this chapter, we present a learning-based lithology characterization framework based on recurrent neural networks. Unlike classical inversion methods, the proposed framework does not require an initial model of the subsurface lithology. Instead, the initial model is inferred from the training data. The inversion network in the proposed framework is trained on well-log data and their corresponding seismic data in a supervised-learning paradigm. Then, the network is used to estimated rock properties throughout the survey using seismic data. The performance of the proposed framework was evaluated in two case studies for density and acoustic impedance estimation from real and synthetic seismic data. The results showed the superiority of recurrent neural networks over convolutional networks in modeling seismic and property traces even with the presence of noise and migration artifacts.

CHAPTER 4
SEMI-SUPERVISED LITHOLOGY CHARACTERIZATION WITH A PREDEFINED FORWARD MODEL

4.1 Overview

In Chapter 3, we presented a supervised lithology characterization framework based on a shallow recurrent neural network. The proposed framework was trained on well-log and seismic data that were of the same vertical resolution. However, well-log data is often sampled at a higher vertical resolution than that of the seismic. One way to overcome this issue is to downsample well-log data to have the same resolution as that of the seismic data. However, this approach limits the resolution of the estimated rock properties. An alternative approach is to interpolate the seismic data to have a similar resolution to that of the well-logs [85]. Interpolation artifacts added to the already noisy seismic data can prevent the network from learning a proper inverse mapping. Furthermore, the inversion network presented in Chapter 3 was shallow due to the limited availability of well-log data. A shallow network does not have a learning capacity to learn a complex mapping such as the mapping from seismic to rock properties. Thus, in this chapter, we present a physics-guided semi-supervised lithology characterization framework that enables training deep neural networks with limited data, with the aid of a physical forward model. Using the physical forward model enables the training of a deep network for lithology characterization with limited data by injecting the physics of wave propagation through the subsurface. The forward model enables the proposed framework to learn from seismic data without requiring a corresponding rock property measurement for each of the seismic traces in the survey. In addition, the deep network in the semi-supervised lithology charac-

terization framework can learn from the full resolution well-logs without the need to downsample them to match that of the seismic data.

In Section 4.2, we present an overview of seismic modeling, and relevant seismic inversion literature using classical and deep-learning methods. Then, in Section 4.3, we present the details of the proposed semi-supervised lithology characterization framework. Sections 4.4 presents a case study of acoustic impedance estimation from seismic data using the proposed framework. We evaluate the proposed framework for elastic impedance estimation from multi-offset prestack seismic data in Section 4.5. Finally, we present a summary of this chapter in Section 4.6.

4.2 Background

4.2.1 Seismic Modeling

When a plane wave (from the seismic source) strikes an interface between two media at a normal incidence, part of the wave's energy is transmitted and the remaining part is reflected and picked up by the geophones at the surface. The ratio between the reflected wave's amplitude and the incident wave's amplitude is known as the Reflection Coefficient (RC). The Transmission Coefficient (TC) is the ratio of the amplitude of the transmitted wave to that of the incident wave. The reflection and transmission coefficients for a normal incident are fully determined by the acoustic properties of the layers below and above the interface, as follows:

$$RC = \frac{V_{p_2}\rho_2 - V_{p_1}\rho_1}{V_{p_2}\rho_2 + V_{p_1}\rho_1} = \frac{AI_2 - AI_1}{AI_2 + AI_1}, \tag{4.1}$$

$$TC = 1 - RC, \tag{4.2}$$

where V_{p_1}, ρ_1, and AI_1 are the P-wave velocity, density, and acoustic impedance of the layer above the interface, and V_{p_2}, ρ_2, and AI_2 are the P-wave velocity, density, and

acoustic impedance of layer below the interface. Higher acoustic impedance contrast between the two media will result in a larger reflection coefficients in (4.2), which gives an intuitive understanding of the relationship between the acoustic properties of the media and the reflection coefficients.

Convolving the reflectively series $RC(t)$ with a wavelet produces a synthetic seismic trace (seismogram) as follows:

$$S(t) = w(t) * RC(t), \tag{4.3}$$

where $w(t)$ is the source wavelet, and $*$ is the linear convolution operator. This process is illustrated in Figure 4.1.

The acoustic impedance trace undergoes two main operations to obtain the seismogram. First, the reflectivity series is computed by taking the derivative of the acoustic impedance trace (high-pass filtering). Then, the seismogram is computed by convolving the reflectivity series with a wavelet which results in a low-pass signal. Hence, the seismograms can be seen as a band-limited version of the acoustic impedance trace which makes the task of recovering acoustic impedance from seismic data very challenging.

Moreover, when the plane wave strikes at an angle $< 90°$, and due to shearing, in addition to compression, part of the incident P-wave is also converted to S-waves. Therefore, an incident P-wave at an acute angle gives rise to reflected P- and S-waves, and transmitted P- and S-waves. In such scenario, (4.2) is no longer applicable. However, the angles at which these waves are reflected or refracted can be computed using Snell's law depending on the elastic properties of the layers, i.e., P-wave velocity V_p and S-wave velocity V_s,

$$\frac{\sin\theta_r}{V_{p_1}} = \frac{\sin\theta_t}{V_{p_2}} = \frac{\sin\phi_r}{V_{s_1}} = \frac{\sin\phi_t}{V_{s_2}}, \tag{4.4}$$

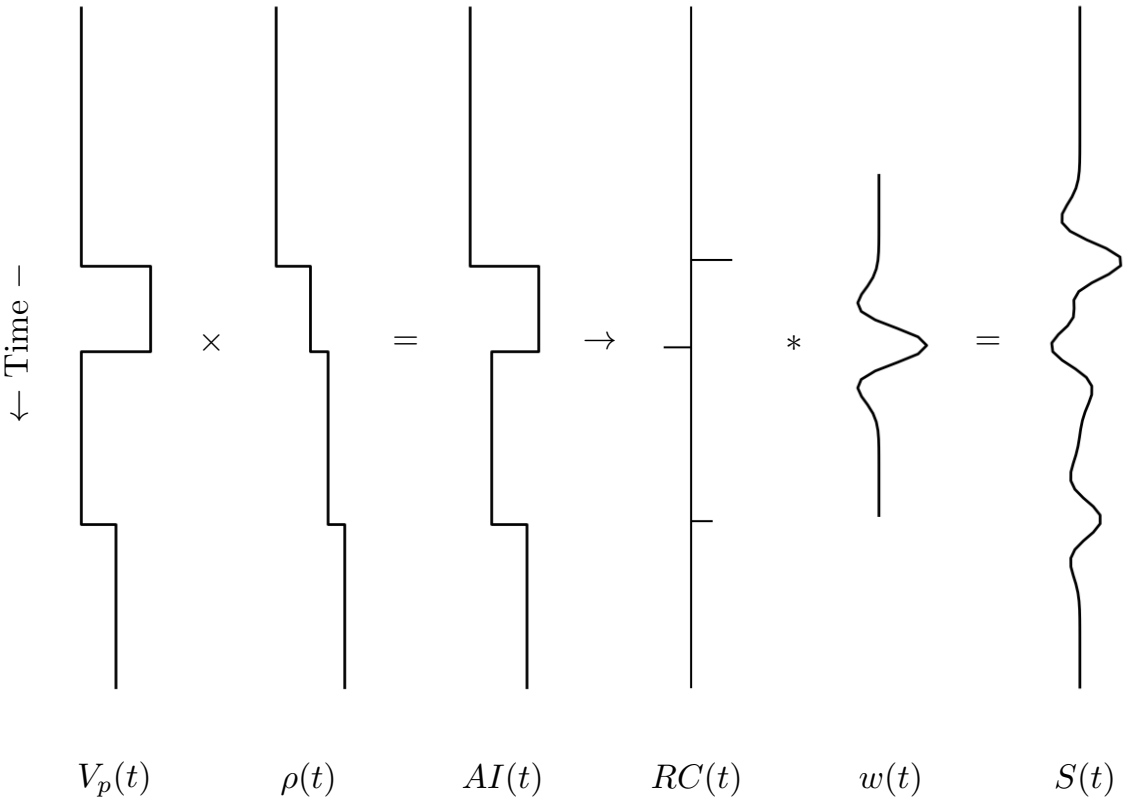


Figure 4.1: Seismograms as a convolution of the reflectivity series with a wavelet.

where θ_r is the reflected P-wave angle which is also equal to the incident angle (θ_i), θ_t is the transmitted P-wave angle, and ϕ_r and ϕ_t are the reflected and transmitted S-wave angles, respectively. The ray paths are illustrated in Figure 4.2.

Hence, knowing the elastic properties of the two media and the incident angle, one can find the reflection and refraction angles for P-wave and S-wave components. Zoeppritz equations [112] give the reflection coefficients for these reflected and refracted components as a function of the elastic properties and the incident angle.

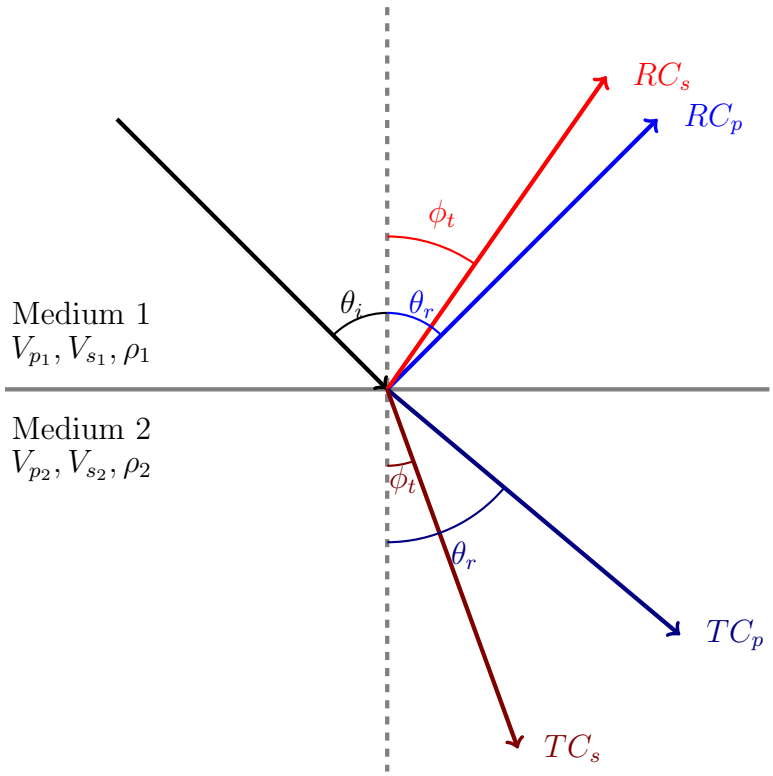


Figure 4.2: An illustration of the reflected and refracted components of a plane wave at an interface.

Zoeppritz equations are expressed in matrix form in [113] as follows:

$$\begin{bmatrix} \cos\theta_i & \frac{V_{p_1}}{V_{s_1}}\sin\phi_r & \frac{V_{p_1}}{V_{p_2}}\cos\theta_t & -\frac{V_{p_1}}{V_{s_2}}\sin\phi_r \\ -\sin\theta_i & \frac{V_{p_1}}{V_{s_1}}\cos\phi_r & \frac{V_{p_1}}{V_{p_2}}\sin\theta_t & \frac{V_{p_1}}{V_{s_2}}\cos\phi_r \\ -\cos 2\phi_r & -\sin 2\phi_r & \frac{\rho_2}{\rho_1}\cos 2\phi_t & -\frac{\rho_2}{\rho_1}\sin 2\phi_t \\ \sin 2\phi_r & -\frac{V_{p_1}^2}{V_{s_1}^2}\cos 2\phi_r & \frac{\rho_2}{\rho_1}\frac{V_{s_2}^2}{V_{s_1}^2}\frac{V_{p_1}}{V_{p_2}}\sin 2\theta_t & \frac{\rho_2}{\rho_1}\frac{V_{p_1}^2}{V_{s_1}^2}\cos 2\phi_t \end{bmatrix} \begin{bmatrix} RC_p \\ RC_s \\ TC_p \\ TC_s \end{bmatrix} = \begin{bmatrix} \cos\theta_i \\ \sin\theta_i \\ \cos 2\phi_r \\ \sin 2\theta_i \end{bmatrix}, \quad (4.5)$$

where RC_p and RC_s are the reflection coefficients for the P-wave and the S-wave components, respectively, and TC_p and TC_s are the transmission coefficients of the P-wave and the S-wave component, respectively.

The solution of Zoeppritz equations is quite complex due to the complexity of their formulation. In addition, they do not reveal an intuitive understanding of how the reflection coefficients are affected by the elastic properties as we have seen in (4.2) [18]. Therefore, there have been many efforts to approximate Zoeppritz equations

under different assumptions [114, 113, 115]. In particular, the Aki-Richards [113] approximation of Zoeppritz equation has been used quite extensively in the literature. It formulates amplitude variations with the offset or incident angle (AVO/AVA) [18] based on the changes of elastic properties and the incident angle. The reflection coefficient, for incident angles smaller than 35°, Aki-Richards approximation is given as a function of the incidence and transmission angle as follows:

$$RC(\theta) = \frac{1}{2}\left(1 - 4\frac{\overline{V}_s^2}{\overline{V}_p^2}\sin^2\theta\right)\frac{\Delta\rho}{\overline{\rho}} + \frac{1}{2\cos^2\theta}\frac{\Delta V_p}{\overline{V}_p} - \left(4\frac{\overline{V}_s^2}{\overline{V}_p^2}\sin^2\theta\right)\frac{\Delta V_s}{\overline{V}_s}, \quad (4.6)$$

where,

$$\overline{V}_p = \frac{1}{2}(V_{p_1} + V_{p_2}), \qquad \overline{V}_s = \frac{1}{2}(V_{s_1} + V_{s_2}), \qquad \overline{\rho} = \frac{1}{2}(\rho_1 + \rho_2),$$
$$\Delta V_p = V_{p_2} - V_{p_1}, \qquad \Delta V_s = V_{s_2} - V_{s_1}, \qquad \Delta\rho = \rho_2 - \rho_1,$$
$$\text{and } \theta = \frac{1}{2}(\theta_i + \theta_t).$$

Based on Aki-Richards' approximation, Connolly [116] formulated the Elastic Impedance (EI) which is an extension of the acoustic impedance for different incident angles. The elastic impedance was later normalized by Whitcombe [117] as follows:

$$EI(\theta) = V_{p_0}\rho_0\left(\frac{V_p}{V_{p_0}}\right)^a\left(\frac{V_s}{V_{s_0}}\right)^b\left(\frac{\rho}{\rho_0}\right)^c, \quad (4.7)$$

where,

$$a = 1 + \tan^2\theta \qquad c = 1 - 4K\sin^2\theta$$
$$b = -8K\sin^2\theta \qquad K = \frac{V_s^2}{V_p^2},$$

and V_p,V_s and ρ are P-wave velocity, S-wave velocity, and density, respectively, and V_{p_0},V_{s_0} and ρ_0 are their respective averages. Note that for $\theta = 0 \rightarrow a = c = 1,\ b = 0,$

the elastic impedance reduces to acoustic impedance, i.e. $EI(0) = AI = V_p\rho$. Then, the reflection coefficients can be computed from elastic impedance as follows:

$$RC(\theta) = \frac{EI_2(\theta) - EI_1(\theta)}{EI_2(\theta) + EI_1(\theta)}. \tag{4.8}$$

Finally, the multi-angle seismogram can be computed by convolving the multi-angle reflectivity series with a wavelet,

$$S(t, \theta) = w(t) * RC(t, \theta). \tag{4.9}$$

4.2.2 Linear Seismic Inversion

As discussed in Section 3.2.1, the goal of seismic inversion is to infer rock properties ($\mathbf{m} \in X$) through an indirect set of seismic reflection measurements ($\mathbf{d} \in Y$). For example, seismic inversion can be used to infer acoustic impedance from zero-offset seismic data, which is then used to estimate the porosity of the subsurface. Furthermore, elastic impedance inversion enables the characterization of the subsurface from an elastic point of view as we have seen in Section 4.2.1 which reveals more details about the lithology of the subsurface. For simplicity, we will present a linear inversion method for acoustic impedance estimation from seismic reflection data [118, 18]. A similar formulation can be used for elastic inversion.

A classical approach to solve the seismic inversion problem is to linearize equation (3.1) the forward based on the seismic model in (4.3) as follows:

$$\mathbf{d} = \mathbf{WDm} + \mathbf{n}, \tag{4.10}$$

where $\mathbf{d}$ is the seismic data, $\mathbf{m}$ is the logarithm of acoustic impedance, $\mathbf{n}$ is a random variable that represents noise in the measurements, $\mathbf{D}$ is the first derivative operator, and $\mathbf{W}$ is the wavelet convolution operator. Thus, comparing (4.10) with (3.1), the

forward operator $\mathcal{F}$ is approximated by $\mathbf{WD}$.

Using an initial model, $\mathbf{m}_0$, which is a guess of the acoustic impedance of the subsurface that is obtained from the acoustic impedance logs, the trace-by-trace inversion is carried out for $\Delta\mathbf{m} = \mathbf{m} - \mathbf{m}_0$ as follows:

$$\hat{\mathbf{m}} = \mathbf{m}_0 + \arg\min_{\Delta\mathbf{m}} \|\mathbf{d} - \mathbf{WD}(\mathbf{m}_0 + \Delta\mathbf{m})\|_2^2. \tag{4.11}$$

Since seismic data does not contain the low-frequency component of the $\mathbf{m}$, an initial model $\mathbf{m}_0$ is needed to enable the inversion algorithm to estimate the absolute values of the model's parameters rather than relative ones.

The minimization problem in (4.11) can be solved using a least-squares approach which works great for noise-free synthetic data. However, if the data is noisy or the forward operator does not produce the same observed data, a good solution is not guaranteed. Thus, the inverse problem is often regularized using Laplacian and Tikhonov's regularization methods [119, 120]. Moreover, additional constraints can be used to incorporate prior knowledge in the optimization objective. Moreover, the presented formulation is known as linear seismic inversion. Other approaches have been proposed to solve the seismic inversion problem using nonlinear methods. Relevant literature about such methods was presented in Section 3.2.1

Alternatively, as discussed in Section 3.2.1, machine learning algorithms can be used to solve the seismic inversion problem. Using neural networks, one can learn inverse mapping $\mathcal{F}_\Theta^\dagger$ (in (3.3)) in either a supervised manner or an unsupervised manner [121]. In supervised learning, the machine learning algorithm is given a set of measurement-model pairs $(\mathbf{d}^*, \mathbf{m}^*)$ (e.g., seismic traces and their corresponding property traces from well-logs). Then, the algorithm learns the inverse mapping by minimizing the following loss function:

$$\mathcal{L}(\Theta) := \mathcal{D}\left(\mathbf{m}^*, \mathcal{F}_\Theta^\dagger(\mathbf{d}^*)\right), \tag{4.12}$$

where $\mathcal{D}$ is a distance measure that compares the estimated rock property to the true property. Therefore, supervised-learning algorithms seek a solution that minimizes the inversion error over the given measurement-model pairs. Note that (4.12) is computed only over a subset of all traces in the survey. This subset includes the traces for which a corresponding property trace is available from well-logs.

The inverse mapping can also be sought in an unsupervised-learning scheme where the learning algorithm is given a set of measurements ($\mathbf{d}$), an initial model ($\mathbf{m}_0$), and a forward model $\mathcal{F}$ [122]. The algorithm then learns by minimizing the following loss function:

$$\mathcal{L}(\Theta) := \mathcal{D}\left(\mathcal{F}\left(\mathcal{F}_{\Theta}^{\dagger}(\mathbf{d}) + \mathbf{m}_0\right), \mathbf{d}\right), \tag{4.13}$$

which is computed over all seismic traces in the survey. The loss in (4.13) is known as data misfit. It measures the distance between the input seismic traces and the synthesized seismograms from the estimated property traces using the forward model. Unsupervised-learning inversion follows a similar formulation of the classical inversion presented earlier.

Although supervised methods are superior to unsupervised ones in various learning tasks (e.g., image segmentation and object recognition), they require large labeled datasets. In the case of the seismic inversion, the labels (i.e., property traces) are sparse since they come from well-logs. Unsupervised methods, on the other hand, do not require labeled data, and thus can be trained on all available seismic data only. However, they require a good initial estimate of the subsurface model and do not integrate well-logs directly in the learning.

A key difference between classical inversion methods and learning-based methods is the optimization objective. In classical inversion, the objective is to find rock properties, $\mathbf{m}$, a posterior probability density function of the rock properties. On the other hand, learning methods aim to find a mapping from the seismic domain to rock property domain. This difference between the two approaches plays a key role in

their computational efficiency. The number of parameters to be estimated in classical inversion is the total number of measurements in the survey. In contrast, the number of parameters to be estimated in learning-based inversion is equal to the number of parameters in the neural network which is independent of the survey size.

Moreover, a key difference between the classical, the unsupervised, and the supervised methods is their sensitivity to the initial model. In classical inversion and unsupervised methods, the initial guess plays an important role in the convergence of the method, and in the final solution. On the other hand, supervised-learning methods are randomly initialized, and prior knowledge is integrated into the objective function and is inferred by the learning algorithm from the training data. Thus, learning-based inversion is less sensitive to the initial guess.

In addition, classical and unsupervised-learning inversion methods are based on a physical model (the seismic model) as we have seen earlier. On the other hand, learning-based inversion methods do not often utilize or enforce a physical model that can be used to check the validity of their outputs. In other words, there is a high dependence on machine learning algorithms to understand the inherent properties of the target output without explicitly specifying a physical model. Such dependence can lead to undesirable or incorrect results, especially when training data is limited.

Finally, one limitation of both the classical inversion and learning-based inversion methods presented so far is that they estimate the subsurface properties from the seismic at the seismic resolution. Well-log data are often downsampled to match that of the seismic. Then, the low-resolution well-log data is used to construct the initial model (in classical inversion and unsupervised-learning methods), or they are used to train the model in the supervised-learning methods.

4.3 Proposed Framework

Given that well-log data is limited in a given survey area, the number of training samples is limited. With such limitation, a combination of regularization techniques must be used to train a learning-based model properly and ensure it generalizes beyond the training dataset [110]. In addition, the data shortage limits the number of layers (and hence parameters) that can be used in learning-based models. Therefore, using deeper networks to capture the highly non-linear inverse mapping from seismic data to rock properties is not be feasible using supervised learning. Thus, in [123, 124], we proposed a semi-supervised learning framework for seismic inversion that integrates both well-log data in addition to data misfit in learning the parameters of the inversion network without requiring an initial model. We utilize a seismic forward model as another form of supervision in addition to well-log data. Formally, the loss function of the proposed semi-supervised inversion framework is written as

$$\mathcal{L}(\Theta) := \alpha \cdot \underbrace{\mathcal{L}_1(\Theta)}_{\text{property loss}} + \beta \cdot \underbrace{\mathcal{L}_2(\Theta)}_{\text{seismic loss}}, \tag{4.14}$$

$$\mathcal{L}_1(\Theta) := \mathcal{D}\left(\mathbf{m}^*, \mathcal{F}_\Theta^\dagger(\mathbf{d}^*)\right), \tag{4.15}$$

$$\mathcal{L}_2(\Theta) := \mathcal{D}\left(\mathcal{F}\left(\mathcal{F}_\Theta^\dagger(\mathbf{d})\right), \mathbf{d}\right), \tag{4.16}$$

where $\alpha, \beta \in \mathbb{R}$ are tuning parameters that govern the influence of each of the property loss and seismic loss, respectively. For example, if the input seismic data is noisy, or well-log data is corrupted, the values of α and β can be used to limit the role of the corrupted data in the learning process. $\mathcal{F}$ is a physics-based forward model that can be used to synthesize seismograms from the estimated property traces, and $\mathcal{F}_\Theta^\dagger$ is the inverse mapping that is learned by the neural network (inversion network).

The property loss is computed over the traces for which we have access to rock

properties form well-logs, $\mathbf{d}^*, \mathbf{m}^*$. The seismic loss, on the other hand, is computed over all seismic traces in the survey, $\mathbf{d}$. Naturally, the size of well-log data is small compared to seismic data. The two losses are minimized simultaneously despite the different number of data points over which they are computed. The seismic loss learns from many more data points that are indirect measurements of the subsurface lithology. On the other hand, the property loss learns from limited but direct and high-resolution measurements of the lithology. Moreover, the vertical resolution of well-log data is finer than that of the seismic traces. Hence, the inversion network is not only required to estimate the lithology from seismic data, but to also learns to interpolate lithology using knowledge from well-logs. It is worth noting that the unsupervised seismic loss does not require an initial model. This is only possible because the property loss is computed directly from well-logs which can help estimate the low-frequency component of $\mathbf{m}$ (i.e. $\mathbf{m}_0$).

The inversion network ($\mathcal{F}_\Theta^\dagger$) can, in principle, be trained using well-log data and their corresponding seismic data only. However, as we have discussed earlier, a deep neural network requires a large dataset of labeled data to train properly which is not possible in a practical setting where the number of well-logs is limited. Integrating a forward model allows the inversion network to learn from the seismic data without requiring their corresponding property traces, in addition to learning from the few available property traces from well-logs. The proposed framework is shown in Figure 4.3

The framework in Figure 4.3 consists of two main modules: the inversion network ($\mathcal{F}_\Theta^\dagger$) with learnable parameters, and a predefined physical forward model ($\mathcal{F}$). The proposed framework takes seismic traces as inputs, and outputs the best estimate of a given rock property. Then, the forward model is used to synthesize seismograms from the estimated rock property traces. The error (data misfit) is computed between the synthesized seismogram and the input seismic traces using the seismic loss. Further-

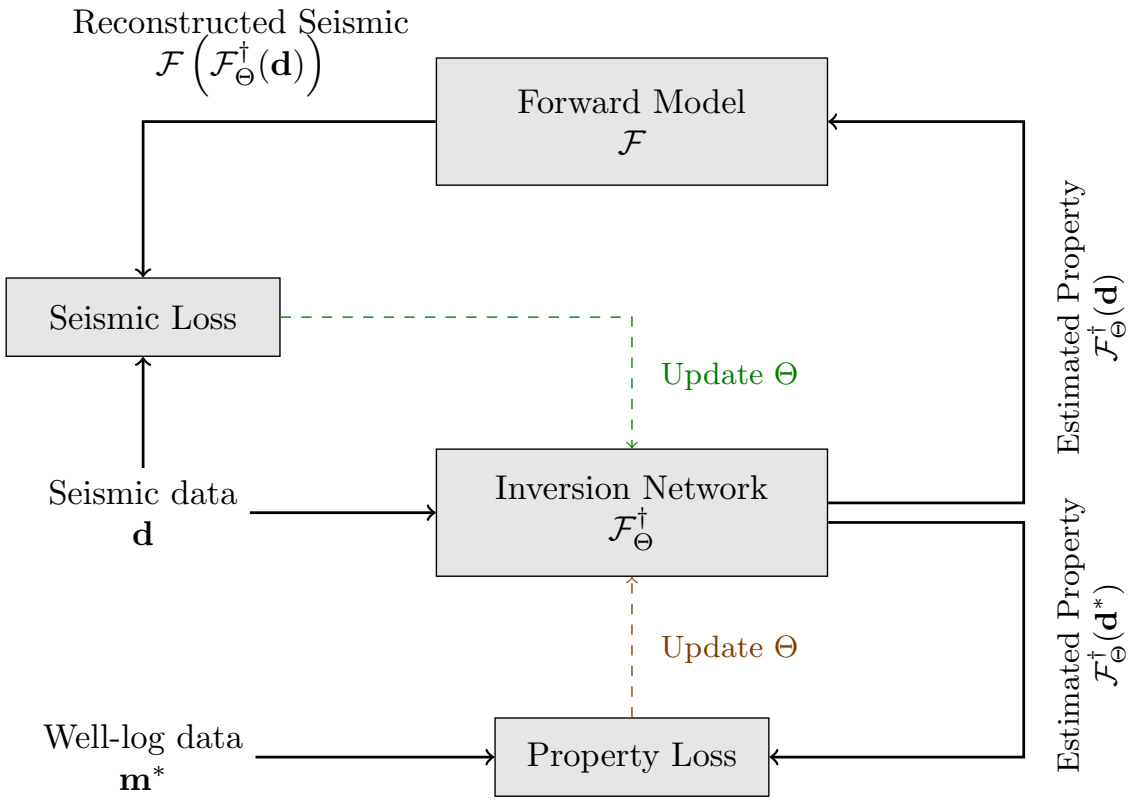


Figure 4.3: The proposed semi-supervised lithology characterization framework with a predefined forward model.

more, the property loss is computed between estimated and true rock properties on traces for which the true property values are available well-logs. The parameters of the inversion network are learned by combining both losses as in (4.14). The details of the inversion network are presented in the next section.

It is important to note that the choice of the forward model is critical in the proposed framework for two reasons. First, the forward model must be able to synthesize at a speed comparable to the speed at which the inversion network processes data. Since deep learning models, in general, are capable of processing large amounts of data in a very short time with GPU technology, the forward model must be fast. Second, the proposed inversion network, like all other deep learning models, learns its parameters according to the gradients with respect to a defined loss function, there-

fore, the forward model must be differentiable in order to compute gradients with respect to the seismic loss. Therefore, in this work, we chose a convolutional forward model based on Zoeppritz equations since we are interested in impedance inversion due to its simplicity and efficiency to reduce computation time. Other choices of the forward model are possible as long as they satisfy the two conditions stated above. We also choose Mean Squared Error as a distance measure ($\mathcal{D}$), and $\alpha = \beta = 1$. Thus, (4.14) can be re-written as,

$$\mathcal{L}(\Theta) := \frac{1}{N^*} \sum_i \left\| \mathcal{F}_\Theta^\dagger(\mathbf{d}_i^*) - \mathbf{m}_i^* \right\|_2^2 + \frac{1}{N} \sum_i \left\| \mathcal{F}\left(\mathcal{F}_\Theta^\dagger(\mathbf{d}_i)\right) - \mathbf{d}_i \right\|_2^2, \tag{4.17}$$

where N^*, and N is the number of samples in $\mathbf{d}^*$, and $\mathbf{d}$, respectively.

The proposed framework offers two advantages over other methods in the literature. First, it allows the incorporation of geophysics into a machine learning paradigm to ensure that the outputs of the networks are obeying physical laws. Second, it learns from all available data without requiring labels for each data sample. The details of the proposed framework are discussed in the next sections.

4.3.1 The Inversion Network

We propose a deep inversion network that consists of a combination of recurrent and convolutional layers. The proposed inversion network (shown in Figure 4.4) consists of four main submodules. These submodules are labeled as *sequence modeling*, *local pattern analysis*, *upscaling*, and *regression*. Each of the four submodules performs a different task in the inversion network.

The *sequence modeling* submodule models temporal dynamics of seismic traces and produces features that best represent the low-frequency content of rock property traces. The *local pattern analysis* submodule extracts local attributes from seismic

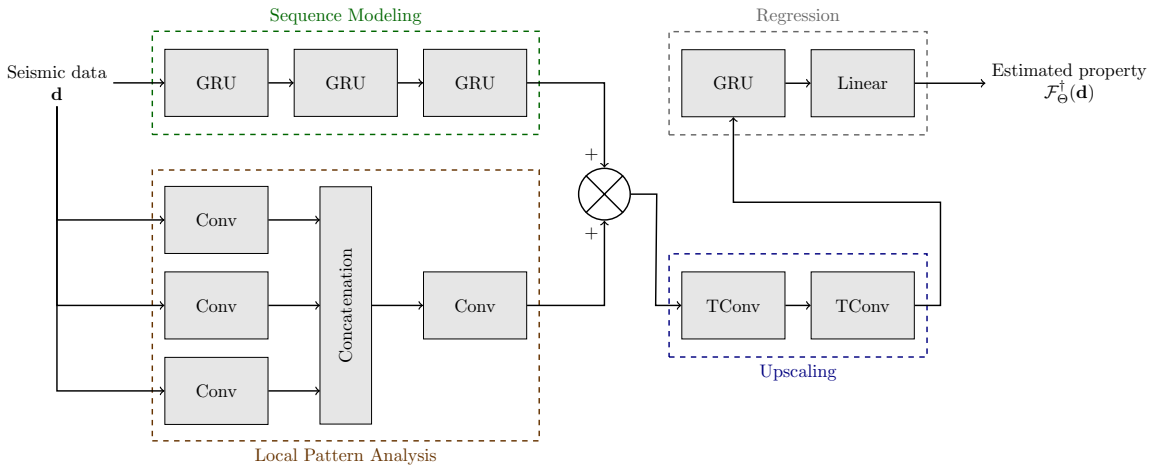


Figure 4.4: The architecture of the inversion network. GRU: Gated Recurrent Unit, Conv: convolutional layer + group normalization + non-linearity, TConv: transposed convolutional layer + group normalization + non-linearity, Linear: fully connected layer.

traces that best model high-frequency trends of the rock property traces. The *upscaling* submodule takes the sum of the features produced by the previous modules and upscales them vertically. This module is added based on the assumption that seismic data are sampled (vertically) at a lower resolution than that of well-log data. Finally, the *regression* submodule maps the upscaled outputs from features domain to target domain (i.e., a given rock property). The details of each of the submodules are discussed next.

Sequence Modeling

The *sequence modeling* submodule consists of a series of bidirectional Gated Recurrent Units (GRU). Each bidirectional GRU computes a state variable from future and past predictions and is equivalent to 2 GRUs where one models the trace from shallow to deeper layers, and the other models the reverse trace. Assuming each input seismic traces have c channels (one channel for each incident angle), the First GRU takes these c channels as input features and computes temporal features based on the temporal variations of the processed traces. The next two GRUs perform a similar

task on the outputs of their respective preceding GRU. The series of all three GRUs is equivalent to a 3-layer deep GRU. Deeper networks are able to model complex input-output relationships that shallow networks might not capture. Moreover, deep GRUs generally produce smooth outputs than shallow ones. Hence, the output of the *sequence modeling* submodule is considered as the low-frequency trend of the target property.

Local Pattern Analysis

Another submodule of the inversion network is the *local pattern analysis* submodule which consists of a set of 1-dimensional convolutional blocks with different dilation factors in parallel. The output features of each of the parallel convolutional blocks are then combined using another convolutional block. Dilation refers to the spacing between convolution kernel points in the convolutional layers [125]. Multiple dilation factors of the kernel extract multiscale features by incorporating information from trace samples that are direct neighbors to a reference sample (i.e., the center sample), in addition to the samples that are further from it. An illustration of dilated convolution is shown in Figure 4.5 for a convolution kernel of size 5 and dilation factors dilation $= 1, 2$ and 3.

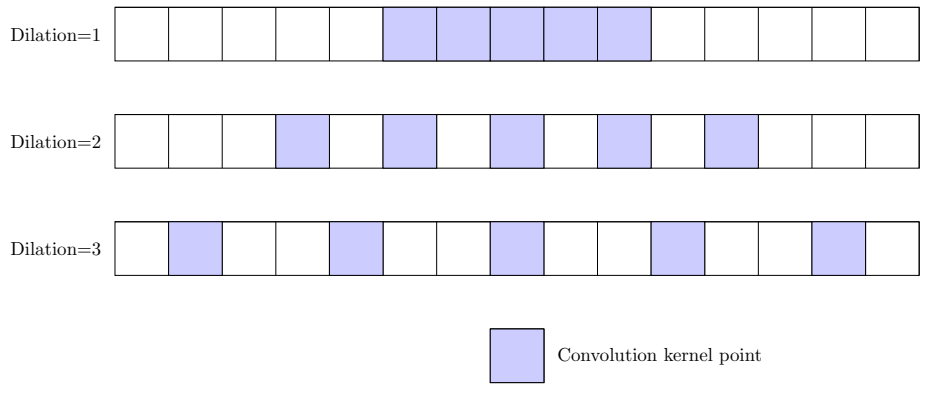


Figure 4.5: An illustration of dilated convolution for multiscale feature extraction (kernel size = 5, dilation factors = $1, 2$, and 3).

A convolutional block (*Conv*) consists of a convolutional layer followed by group

normalization [126] and an activation function. Group normalization divides the output of the convolutional layers into groups, and normalizes each group using a learned mean and standard deviation. They have been shown to reduce the covariant shift in the learned features and speed up the learning. In addition, activation functions are one of the building blocks of any neural network. They are a source of non-linearity that allows the neural networks to approximate highly non-linear functions. In this work, we chose the Rectified Linear Unit (ReLU) as the activation function.

Convolutional layers operate on small windows of the input trace due to their small kernel sizes. Therefore, they capture high-frequency (local) content. Since convolutional layers do not have a state variable (memory-less), they can not estimate the low-frequency content of property traces. The outputs of the *local pattern analysis* submodule have the same dimensions as the outputs of the *sequence modeling* submodule. Hence, the outputs of the two modules are added to obtain a full-band frequency content.

Upscaling

Seismic data is sampled at a lower rate than that of well-logs data. The role of the *upscaling* submodule is to compensate for this resolution mismatch. This submodule consists of two transposed convolution blocks ($T\ Conv$) with different kernel strides. The stride controls the factor by which the inputs are upscaled. A stride of ($s = 2$) deconvolutional block produces an output that has twice the number of the input samples (vertically).

Transposed convolutional layers (also known as fractionally-strided convolutional layers) are upscaling modules with learnable kernel parameters unlike classical interpolation methods with fixed kernel parameters (e.g., linear interpolation). They learn kernel parameters from the training data. They have been used for various applications like semantic segmentation and seismic structure labeling [127, 15]. The

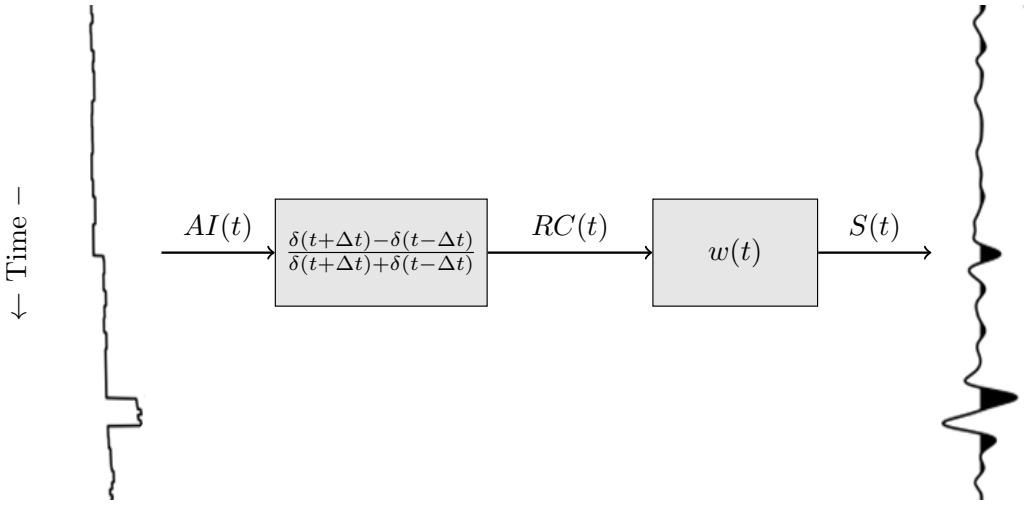
transposed convolutional blocks in Figure 4.4 have a similar structure as the convolutional blocks introduced earlier. They are a series of deconvolutional layer followed by a group normalization module and an activation function.

Regression

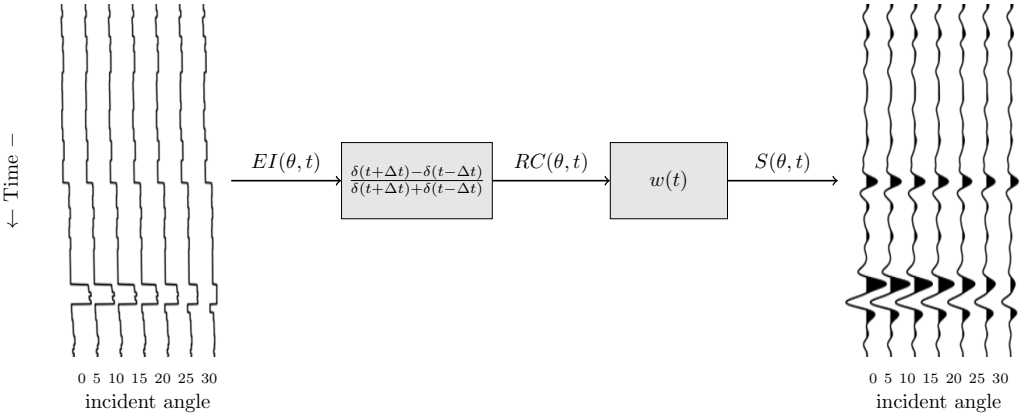
The final submodule in the inversion network is the *regression* submodule which consists of a GRU followed by a linear mapping layer (fully-connected layer). Its role is to regress the extracted features from the other submodules to the target domain (rock property). The GRU in this module is a simple 1-layer GRU that augments the upscaled outputs using global temporal features. Finally, a linear affine transformation layer (fully-connected layer) takes the output features from the GRU and maps them to the same number of features in the target domain.

4.3.2 The Forward Model

Forward modeling is the process of synthesizing seismograms from elastic properties of the earth (i.e., P-wave velocity, S-wave velocity, and density) or from a function of the elastic properties such as elastic impedance. In our work, we choose a convolutional forward model based on Zoeppritz equations. When estimating acoustic impedance, the forward model simply takes acoustic impedance and produces a seismogram. However, when estimating elastic impedance, the forward model takes multi-angle elastic impedance, and generates corresponding multi-angle seismograms. The forward models used to synthesize seismograms from acoustic and elastic impedance are shown in Figure 4.6. It is worth noting that the wavelet of the forward model is assumed to be known. Thus, we refer to this model as a predefined forward model. We use Ormsby wavelet (5-10-60-80 Hz) in the forward model as described in [23].



(a) Acoustic

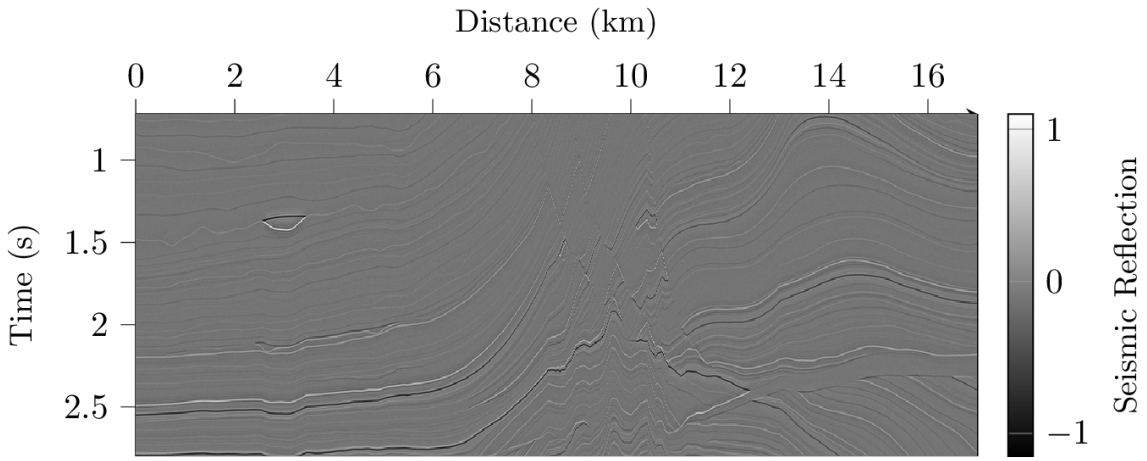


(b) Elastic

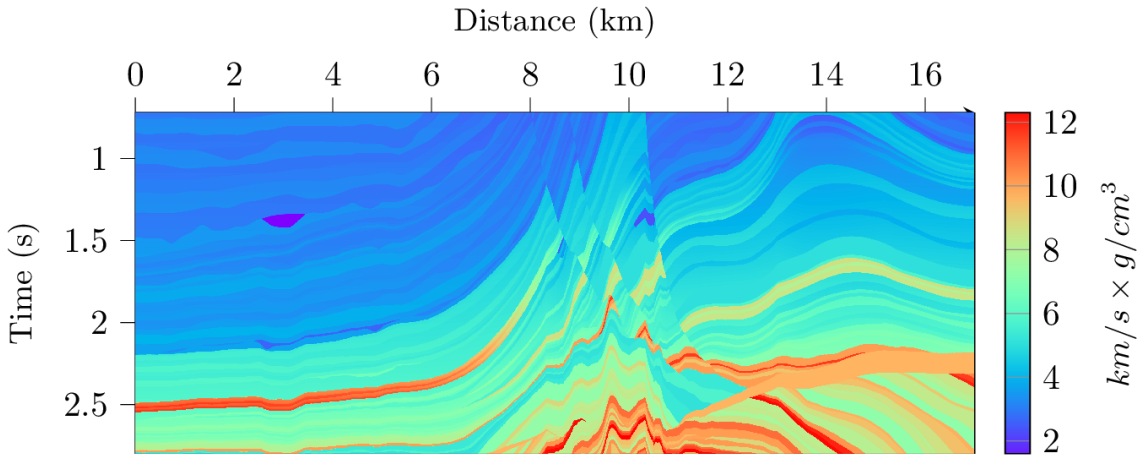
Figure 4.6: The forward models used to synthesize seismograms from impedance.

4.4 Case Study: Acoustic Impedance Estimation

In order to validate the proposed framework, we present a case study on acoustic impedance estimation from post-stack seismic data from Marmousi 2 model as a case study [23]. However, since the forward model assumes inputs in time instead of depth, we perform the estimation in the time domain. Furthermore, we compare the results obtained using the proposed framework with the results obtained using the classical trace-by-trace inversion presented in Section 4.2.2. The synthetic seismic section and its corresponding acoustic impedance profile are shown in Figure 4.7.



(a) Synthetic zero-offset seismic section



(b) Acoustic impedance

Figure 4.7: A Synthetic seismic section (in time) from Marmousi 2 [23] and its corresponding acoustic impedance section.

In this experiment, we use zero-offset synthetic seismic data with 15 dB Gaussian noise. Furthermore, the seismic data was downsampled by a factor of six to evaluate the performance of the proposed framework for estimating a high-resolution acoustic impedance from low-resolution seismic data.

4.4.1 Training the Network

We train the network using 10 evenly-spaced seismic traces and their corresponding acoustic impedance traces (labels) ($<$ 0.4% of the total number of traces in the

model). All seismic traces are also used without labels. The inversion network is initialized with random parameters. Seismic traces are fed to the inversion network to get an estimate of acoustic impedance. If the input seismic trace is part of the training dataset (i.e., a label is available), the error is computed between the estimated acoustic impedance and the true one using the property loss. However, if the seismic traces are not labeled, the forward model is used to synthesize a seismogram from the estimated acoustic impedance trace. The seismic loss is computed as the MSE between the seismogram and the input seismic trace. The total loss is computed as the sum of the two losses. Then, the gradients of the total loss are computed, and the parameters of the inversion network are updated accordingly. The process is repeated until a convergence criterion is met.

The proposed framework was implemented in Python using PyTorch deep learning library [128]. For optimization, we used Adam [129] which is a gradient-based stochastic optimization technique with an adaptive learning rate that was designed specifically for training deep neural networks.

4.4.2 Results

The trained network is used to estimate the acoustic impedance for all seismic traces in the model. In addition, we invert the seismic data for acoustic impedance using the classical trace-by-trace linear inversion. We used a blurred version of the acoustic impedance section as an initial smooth model for the linear inversion methods. Furthermore, since the seismic data is noisy, we used Tikhonov regularization. In addition, the estimated acoustic impedance is then used as an initial model and the inversion algorithm is run again with additional Laplacian regularization to reduce the effect of the noise. The estimated acoustic impedance sections using the proposed framework and linear inversion methods are shown in Figure 4.8.

The results show that the proposed framework was able to generate an acoustic

impedance profile that is more detailed compared to the classical inversion approaches. This is clearly visible in thin layers. However, linear inversion produces results that are laterally smoother compared to our approach. This smoothness comes from the smooth initial model used. Furthermore, the lateral discontinuity in the estimated acoustic impedance using the proposed framework is due to the use of 1-dimensional neural networks that perform the inversion for each trace independently. Although the estimated acoustic impedance using the proposed framework are superior in terms of resolution, there are a few areas in the estimated section that were not correctly resolved. For example, the gas-charged pocked at $t = 1.4s,\ x = 3$ was not correctly estimated. The error from this error propagated to nearby samples due to the use of recurrent neural networks. However, the network recovers from this error for deeper layers and estimates the acoustic impedance correctly.

To quantitatively evaluate the results, we compute the Pearson Correlation Coefficient (**PCC**) and coefficient of determination ($\mathbf{r}^2$) for each trace in the section. The results averaged over all traces in the model are reported in Table 4.1. We also report the metrics computed on the smooth initial model use for linear inversion.

Table 4.1: Quantitative evaluation of the estimated acoustic impedance from synthetic seismic data with a 15dB noise.

Method	**PCC**	$\mathbf{r}^2$
Smooth model	0.8957	0.7895
Inversion (Tikhonov)	0.9457	0.8890
Inversion (Tikhonov + Laplacian)	0.9534	0.9008
Ours [124]	**0.9701**	**0.9158**

The results in Table 4.1 show that the proposed method outperforms both inversion approaches in **PCC** and $\mathbf{r}^2$ metrics. It is worth noting that the initial model, before inversion, gives a good estimate of the low-frequency content of the acoustic impedance as shown in the table. This smooth model was used as a starting point for the two classical inversion approaches. However, our proposed method does not

require an initial model. Instead, the network is randomly initialized.

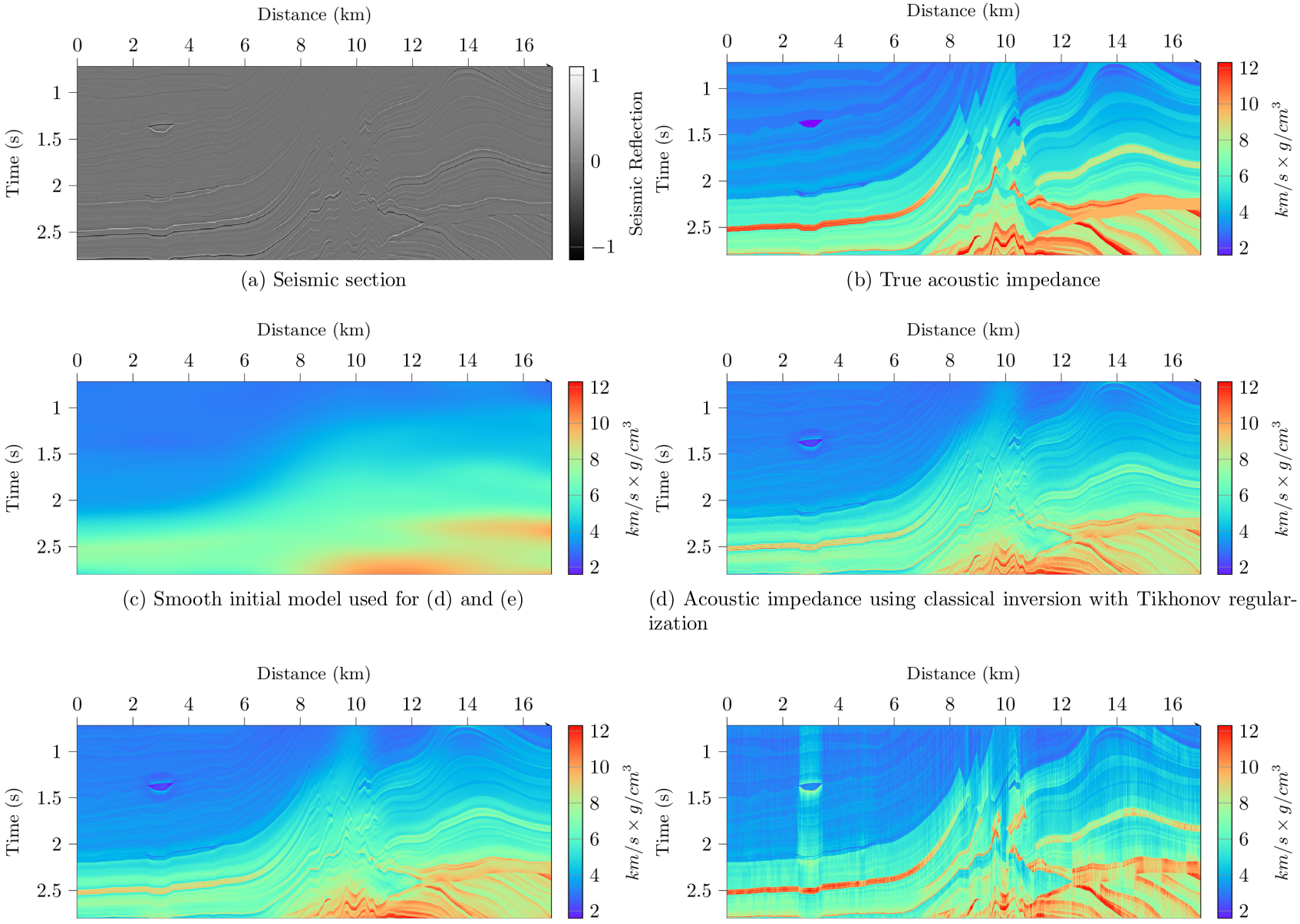
Figure 4.9 shows scatter plots of the estimated acoustic impedance versus the true acoustic impedance. It is evident from the scatter plots that the estimated acoustic impedance using the proposed framework is linearly correlated with the true acoustic impedance which was indicated by the high **PCC** value. Moreover, the proposed framework estimates the high-impedance values better than the classical approach as indicated by the scatter plots. Areas of the section that have high-impedance are mostly thin layers that are not be clearly resolved in the low-resolution input seismic. Using the upscaling module, our proposed framework can interpolate the seismic using the learned interpolation kernels, which helps resolve such thin layers.

Note that in our experiments, we assumed that the true forward model is known, including the wavelet used to generate the data. With the exception of noise present in the data, the synthesized seismic data matches the input seismic data in both the classical and learning-based approaches. However, the estimation of an accurate forward model might not be possible for field data where the wavelet is not fully known. Therefore, we repeat the experiments using wave-equation-migrated seismic data. We use the same forward model that was used in the synthetic seismic case, which does not match the one used to generate the data.

Linear inversion approaches, with regularization, did not converge when using wave-equation-migrated data because the forward model did not match the one used in the data. However, in our proposed framework, since we can control how much the network learns directly from well-log data and the forward model through the two-term loss, we used a small β value in (4.17) ($\beta = 0.01$). This limits the role of the forward model in the learning in the case of wavelet mismatch. The estimation acoustic impedance from wave-equation-migrated seismic data is shown in Figure 4.10.

The results are significantly worse when we used wave-equation-migrated seismic

data instead of synthetic data. This is expected since the two terms of the loss in (4.17) have different assumptions. The property loss assumes that the input is migrated seismic data based on the input traces. On the other hand, the seismic loss assumes that the inputs are synthetic based on the forward model. To resolve this discrepancy, we used a small value of β to limit its role. The **PCC** and $\mathbf{r}^2$ for the estimated acoustic impedance using wave-equation-migrated seismic data are 0.9091, and 0.7892, respectively. We address the issue of mismatching forward model in the next chapter. Although the results with migrated seismic data are not as good as the results using synthetic data, the network converges to a valid solution guided by the property loss. The thin layers and deep parts of the estimation acoustic impedance were not estimated accurately since they were not clearly visible in the migrated seismic data. Note that no spatial regularization was used in our approach, and the results shown are the direct output of the network with no post-processing.



(a) Seismic section

(b) True acoustic impedance

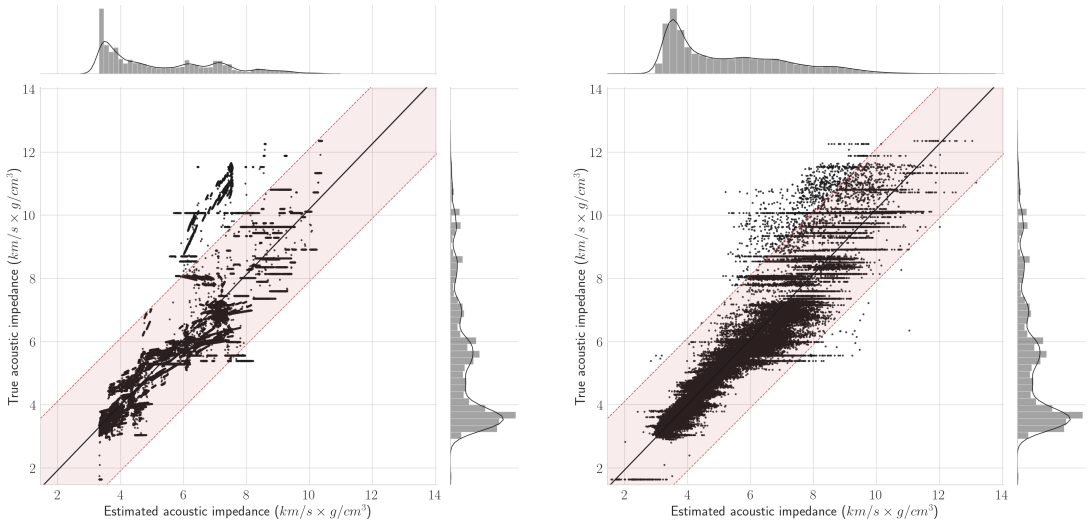
(c) Smooth initial model used for (d) and (e)

(d) Acoustic impedance using classical inversion with Tikhonov regularization

(e) Acoustic impedance using classical inversion with additional Laplacian regularization

(f) Acoustic impedance using the proposed semi-supervised framework

Figure 4.8: Estimated acoustic impedance from a synthetic seismic section with 15dB noise.



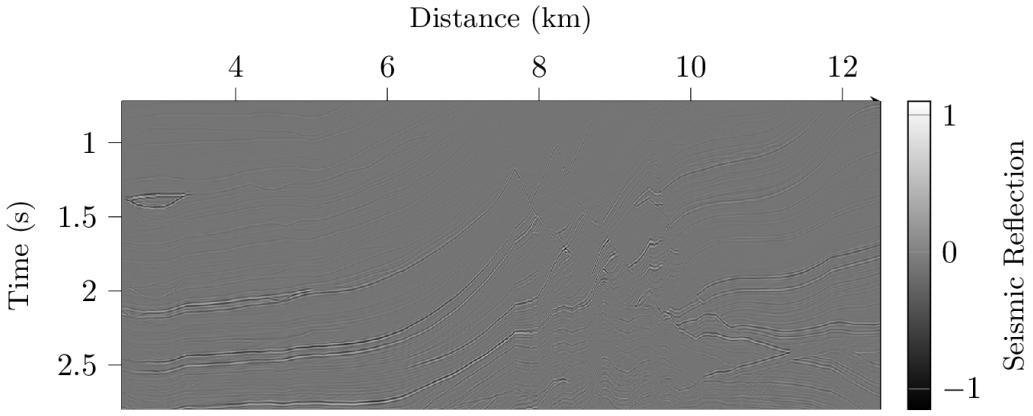
(a) Smooth model

(b) Classical inversion (Tikhonov)

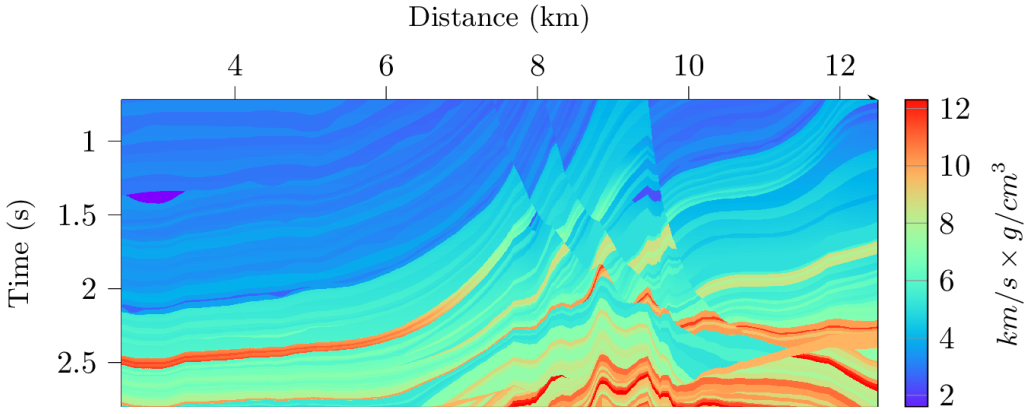
(c) Classical inversion (Tikhonov+Laplacian)

(d) Ours

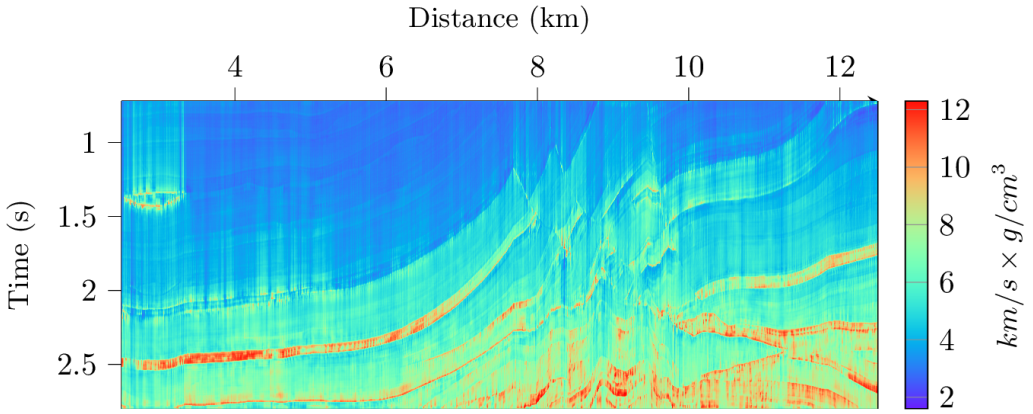
Figure 4.9: Scatter plots of the true acoustic impedance versus the estimated acoustic impedance. The shaded region include all points that are within one standard deviation of the true acoustic impedance. The black line is the best linear fit.



(a) Wave-equation-migrated seismic section



(b) True acoustic impedance



(c) Estimated acoustic impedance using the proposed semi-supervised framework

Figure 4.10: Estimated acoustic impedance from a migrated seismic section.

4.5 Case Study: Elastic Impedance Estimation

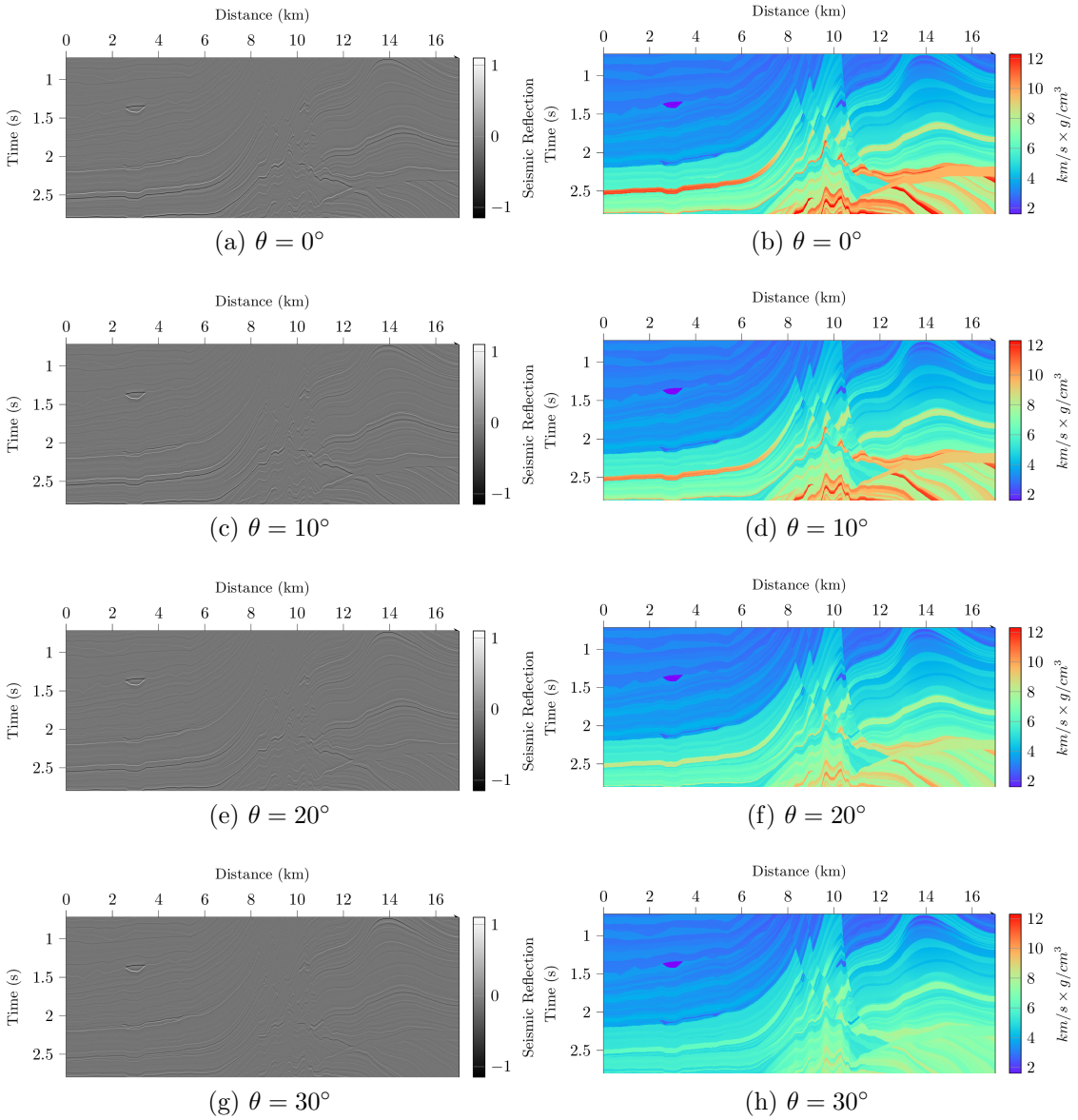
The applications of the proposed framework are not limited to acoustic impedance inversion. We can change the number of channels in the inversion network to estimate more than one property simultaneously. In this case study, we evaluate the performance of the proposed framework for elastic impedance inversion from prestack synthetic seismic data (in time) from Marmousi 2 dataset.

We used the elastic model of Marmousi 2 to compute elastic impedance for 4 incident angles $\theta = 0°$,$10°$,$20°$, and $30°$. Multi-angle seismic data (a total of $N = 2720$ traces) is then generated from the elastic impedance using a forward model with Ormsby wavelet (5-10-60-80 Hz) following the synthesis procedure in [23]. The seismic traces are then downsampled by a factor of six to simulate the resolution difference between seismic and well-log data. Finally, a 15 dB white Gaussian noise is added to assess the robustness of the proposed framework to noise. The multi-angle seismic sections are the corresponding elastic impedance sections are shown in Figure 4.11

4.5.1 Results

To train the proposed inversion framework, we follow the same procedure that was used in Section 4.4. We chose 10 evenly-spaced pseudo-wells for training. We assume we have access to both elastic impedance and seismic data for the training traces. For all other traces in the survey, we assume we have access to seismic data only. The inversion network used in this case study is similar to the one used in Section 4.4. However, we change the number of input and output channels of the neural network to enable the network to estimate multi-angle elastic impedance. Furthermore, we use the elastic version of the forward model shown in Figure 4.6 to synthesize multi-angle seismic from the estimated elastic impedance.

We also compare with classical linear inversion approaches. We used a blurred



(a) $\theta = 0°$ (b) $\theta = 0°$

(c) $\theta = 10°$ (d) $\theta = 10°$

(e) $\theta = 20°$ (f) $\theta = 20°$

(g) $\theta = 30°$ (h) $\theta = 30°$

Figure 4.11: Synthetic multi-angle seismic sections (left), and their corresponding true elastic impedance sections.

initial model of the elastic impedance sections as an initial model for the inversion. Also, since the seismic data is noisy, we used Tikhonov regularization and Laplacian regularization to reduce the effect of the noise. The results using the proposed framework in addition to linear inversion results are shown in Figures 4.12-4.15 for different incident angles.

The results obtained using the proposed framework show more details compared

to classical inversion approaches. This is visible in thin layers that are not clearly resolved in the seismic. The high-resolution results are the result of using the learned-interpolation network (*upscaling* submodule). Furthermore, we can also see that the results for $\theta = 0^\circ$ are better than the acoustic impedance results presented in the previous case study. This improvement is the result of the simultaneous inversion of all multi-angle traces. The changes of the seismic response between the multiple angles are used in the proposed framework to estimate the elastic properties of the subsurface which in turn improved the acoustic impedance estimate ($\theta = 0^\circ$).

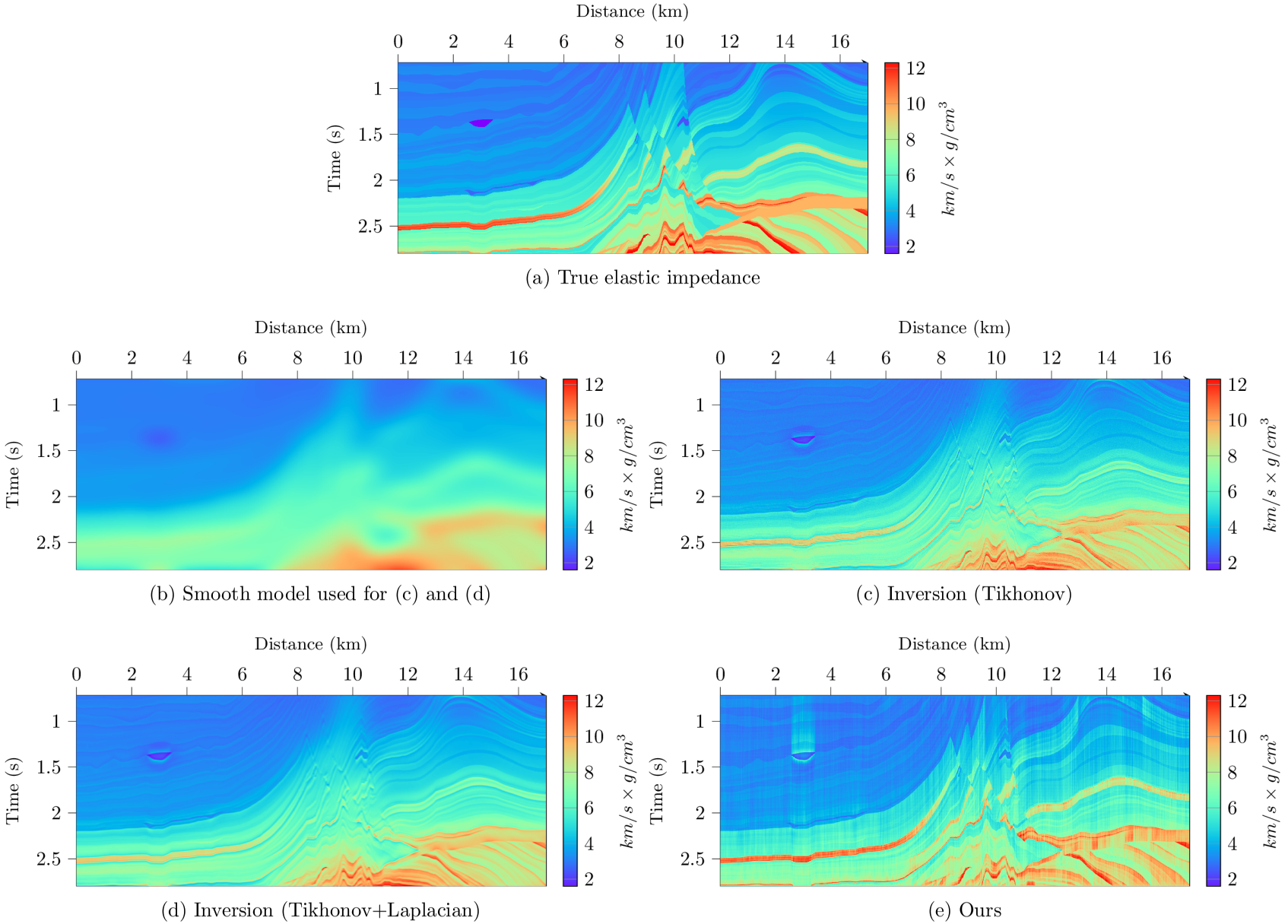
In order to evaluate the results quantitatively, we computed **PCC** and $\mathbf{r}^2$ coefficients between the true and estimated elastic impedance for all incident angles. The results are summarized in Table 4.2. The numbers reported in Table 4.2 show that the proposed framework outperforms linear inversion methods on all incident angles on both **PCC** and $\mathbf{r}^2$. It is worth noting that the smooth model was only used for the linear inversion methods. In the proposed framework, no initial model was used.

Finally, we show scatter plots of the true elastic impedance versus the estimated elastic impedance for a few traces for all angles in Figure 4.16. It can be seen in the figure that the classical inversion approaches do not estimate high impedance values as well as the proposed framework. In addition, the majority of the points estimated using the proposed framework are within the shaded region which includes all points that are within one standard deviation of the true acoustic impedance. It is worth noting that the high impedance values are mostly thin layers. Since seismic data do not resolve thin layers due to their low-resolution, it is expected that all methods do not estimate their impedance values accurately.

4.6 Summary

In summary, we presented a semi-supervised lithology characterization framework based on recurrent and convolutional networks. The proposed framework uses a

predefined physical forward model that enables it to learn from unlabeled data (i.e., seismic data without well-logs). Furthermore, the proposed framework does require an initial model. Instead, the initial model is inferred using the training well-logs. The proposed framework was used in two case studies for acoustic and elastic impedance inversion from noisy synthetic seismic data. The semi-supervised framework was trained on 10 impedance traces. Then, it was used to estimate the impedance for the entire section. The results obtained using the proposed framework were more detailed, especially in thin layers and complex geology, compared to classical inversion methods. Furthermore, the proposed framework was evaluated on migrated seismic data to estimate the acoustic impedance. Although the used forward model in the migrated seismic experiment was not accurate, the two-term loss of the proposed framework was used to limit the role of the forward model in the learning process.



(a) True elastic impedance

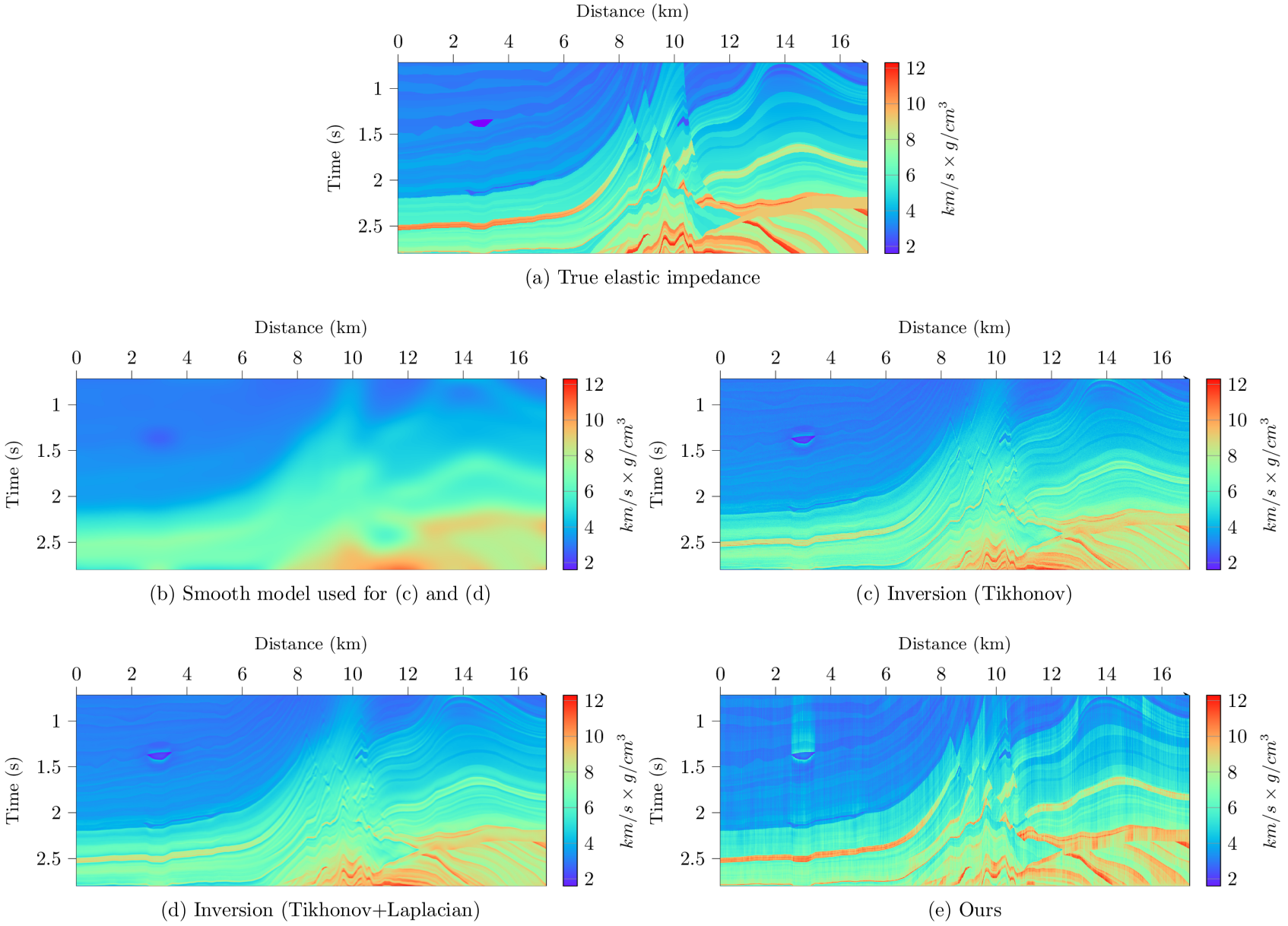
(b) Smooth model used for (c) and (d)

(c) Inversion (Tikhonov)

(d) Inversion (Tikhonov+Laplacian)

(e) Ours

Figure 4.12: Estimated elastic impedance ($\theta = 0°$) from synthetic multi-angle seismic data.



(a) True elastic impedance

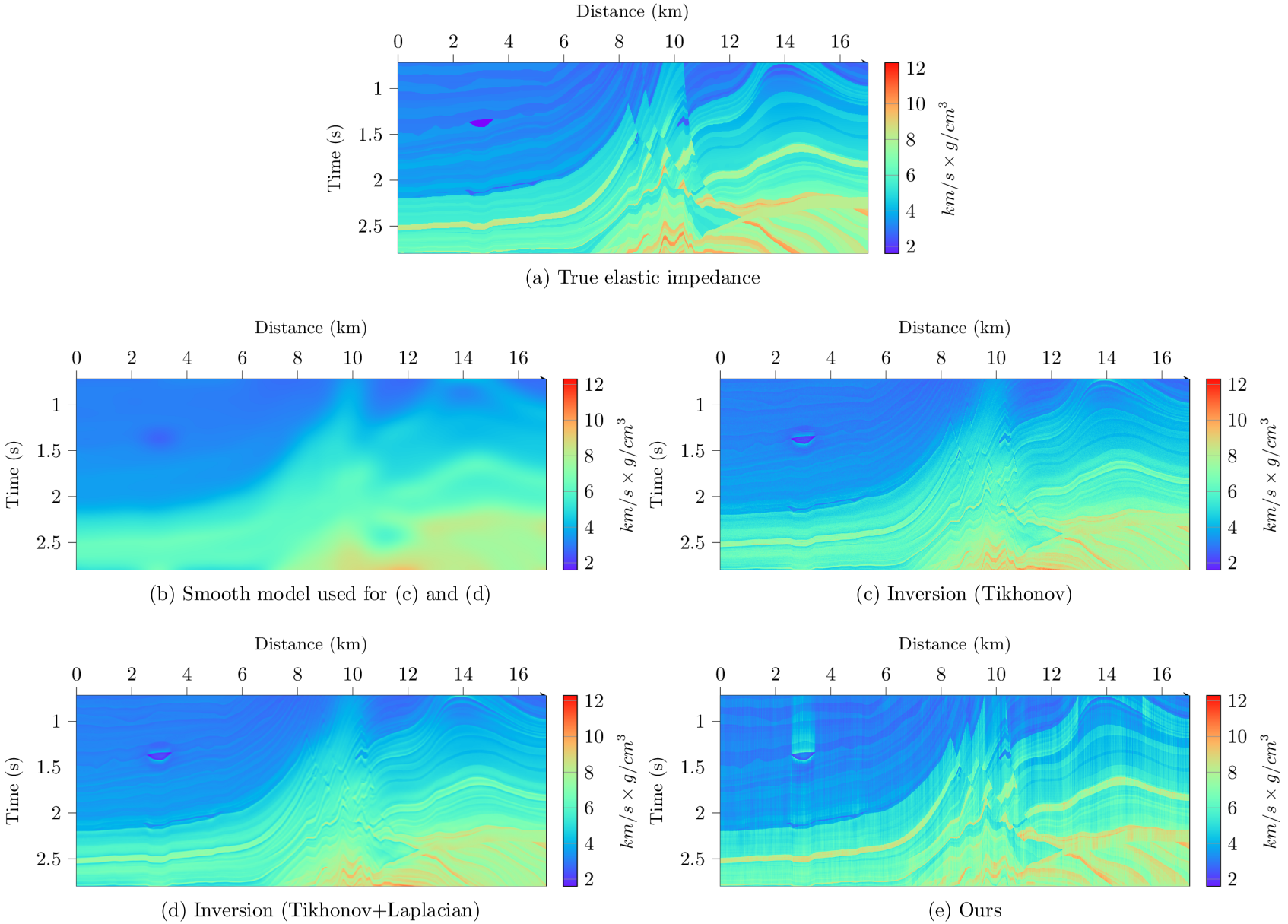
(b) Smooth model used for (c) and (d)

(c) Inversion (Tikhonov)

(d) Inversion (Tikhonov+Laplacian)

(e) Ours

Figure 4.13: Estimated elastic impedance ($\theta = 10°$) from synthetic multi-angle seismic data.



(a) True elastic impedance

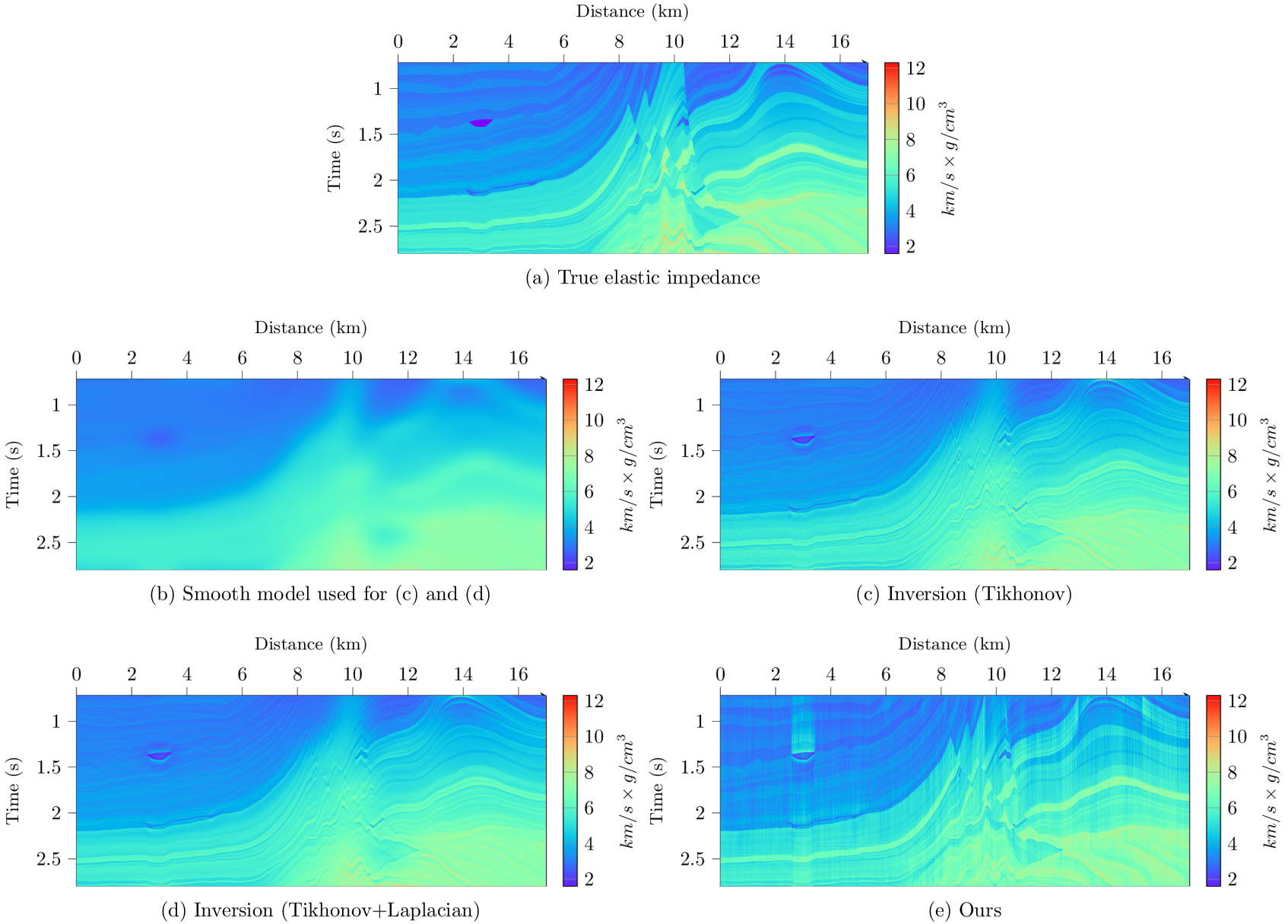
(b) Smooth model used for (c) and (d)

(c) Inversion (Tikhonov)

(d) Inversion (Tikhonov+Laplacian)

(e) Ours

Figure 4.14: Estimated elastic impedance ($\theta = 20°$) from synthetic multi-angle seismic data.



(a) True elastic impedance

(b) Smooth model used for (c) and (d)

(c) Inversion (Tikhonov)

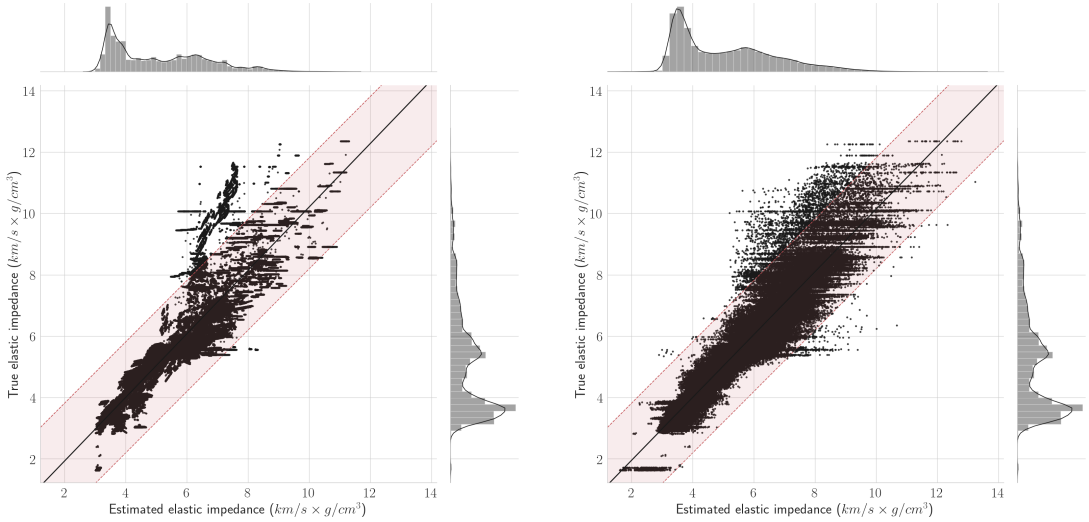
(d) Inversion (Tikhonov+Laplacian)

(e) Ours

Figure 4.15: Estimated elastic impedance ($\theta = 30°$) from synthetic multi-angle seismic data.

Table 4.2: Quantitative evaluation of the estimated elastic impedance from multi-angle synthetic seismic data with a 15dB noise.

	PCC					**r^2**				
Method	0°	10°	20°	30°	Average	0°	10°	20°	30°	Average
Smooth model	0.9101	0.9163	0.9325	0.9472	0.9265	0.8212	0.8329	0.8635	0.8899	0.8519
Inversion (Tikhonov)	0.9467	0.9494	0.9564	0.9612	0.9534	0.8907	0.8962	0.9096	0.9169	0.9034
Inversion (Tikhonov + Laplacian)	0.9534	0.9562	0.9635	0.9698	0.9607	0.9007	0.9067	0.9218	**0.9331**	0.9156
Ours [124]	**0.9789**	**0.9801**	**0.9817**	**0.9775**	**0.9795**	**0.9425**	**0.9434**	**0.9409**	0.9189	**0.9364**



(a) Smooth model

(b) Inversion (Tikhonov)

(c) Inversion (Tikhonov+Laplacian)

(d) Ours

Figure 4.16: Scatter plots of the true elastic impedance versus the estimated elastic impedance. The shaded region include all points that are within one standard deviation of the true acoustic impedance. The black line is the best linear fit.

CHAPTER 5
SEMI-SUPERVISED LITHOLOGY CHARACTERIZATION WITH A LEARNED FORWARD MODEL

5.1 Overview

In Chapter 4, we presented a semi-supervised lithology characterization workflow that uses a predefined physical model to enable the inversion network to learn from unlabeled data. Although the forward model improved the estimation results for both synthetic and migrated seismic data, estimating the parameters of the forward model is not a trivial task. For example, when using a predefined forward model, it was assumed that the wavelet used to generate the seismic data is known. When the wavelet used in the forward model does not match that of the seismic data, linear inversion methods did not converge to a valid solution. The proposed semi-supervised framework in Chapter 4 was able to learn a valid inverse mapping by limiting the role of the forward model in the learning process.

In addition, the forward model must be defined differently depending on the estimated property. For instance, when estimating acoustic or elastic impedance, the forward model was defined based on Zoeppritz equations as we have seen in Chapter 4. However, when estimating other properties, the forward model might not be easily defined using a physics-based equation. Thus, we believe that the use of a predefined model might limit the applications of the proposed semi-supervised lithology characterization framework. Therefore, in this chapter, we propose a semi-supervised framework that can learn both the inverse and forward mapping simultaneously. The forward model is learned from the data through a shallow neural network, and it enables the inversion network to learn from unlabeled data.

In Section 5.2, we present an overview of relevant literature from the fields of machine learning and deep learning. In 5.3, we present the details of the proposed semi-supervised lithology characterization framework with a learned forward model. Section 4.4 presents a case study of acoustic impedance estimation from seismic data using the proposed framework. Then, we incorporate multi-offset prestack seismic data for elastic impedance estimation in Section 4.5. Finally, we present a summary of this chapter in Section 4.6.

5.2 Background

Deep neural networks have achieved better-than-human performance in many computer vision tasks such as image segmentation and image recognition [130]. Such networks are often trained on a large dataset of labeled examples in a supervised-learning scheme. However, the absence or shortage of such large labeled datasets severely limits the ability of neural networks to learn.

In general, deep neural networks are trained to find a mapping $\mathcal{F}_\Theta : X \rightarrow Y$ from one domain (source domain, X) to another (target domain, Y). Alternatively, they can be trained to find an inverse mapping, $\mathcal{F}_\Theta^\dagger : Y \rightarrow X$ where $Y = \mathcal{F}(X)$, as we have seen in Chapters 3 and 4. Such inverse mapping can be often well-approximated by a deep neural network given the availability of a large dataset of training examples. However, with limited data, neural networks might overfit to training data, and fail to generalize for new samples. As we have seen in Chapter 4, knowledge of the forward model ($\mathcal{F} : X \rightarrow Y$) can be used as a learning constraint to enable the network to learn the task at hand without overfitting [121, 124]. However, a precise forward model can often be difficult or impossible to define for computer vision problems where the task is to densely estimate physical properties from visual data. This gives rise to what is known as *cycle consistency learning*, where a neural network is used to learn a forward mapping that helps the network in learning a stable

representation of the data. Cycle consistency learning has been successfully applied to various computer vision applications such as style-transfer, domain adaptation, correspondence learning, and many others [131, 132, 133, 134, 135].

For $\mathbf{x} \in X, \mathbf{y} \in Y$, cycle inconsistency is defined as the distance between $\mathbf{y}$ and its reconstructed version (i.e., $\mathcal{F}_{\Theta_2}\left(\mathcal{F}^{\dagger}_{\Theta_1}(y)\right)$), where $\mathcal{F}^{\dagger}_{\Theta_1}$ and $\mathcal{F}_{\Theta_2}$ are the learned forward and inverse mapping, respectively. Cycle inconsistency loss can be formally written as follows:

$$\mathcal{L} := \mathcal{D}\left(y, \mathcal{F}_{\Theta_2}\left(\mathcal{F}_{\Theta_1}(y)\right)\right), \tag{5.1}$$

where $\mathcal{D}$ is a distance measure. Figure 5.1 illustrates the idea of cycle inconsistency for inverse problems.

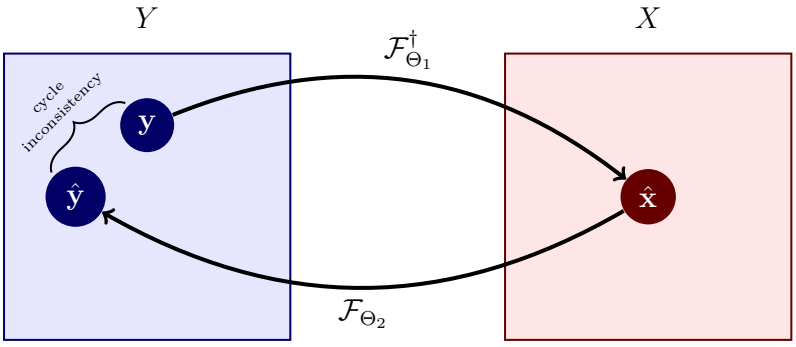


Figure 5.1: An illustration of cycle inconsistency loss for inverse problems.

Minimizing the cycle inconsistency loss helps the inversion network to better represent the data and, therefore, generalize well beyond the training data. In addition, it enables deep networks to learn data representation from unlabeled data since cycle inconsistency is computed between inputs and their reconstructed versions through the learned forward network.

5.3 Proposed Framework

In Chapter 4, we used a variant of the cycle inconsistency loss using a physical forward model. However, it is also possible to learn the forward model from the data using

a neural network. Learning the forward model enables neural networks to work on other types of data for which a forward model is not well defined. Thus, in [123], we proposed a semi-supervised lithology characterization framework using a learned forward model instead of a predefined one. The proposed framework is shown in Figure 5.2

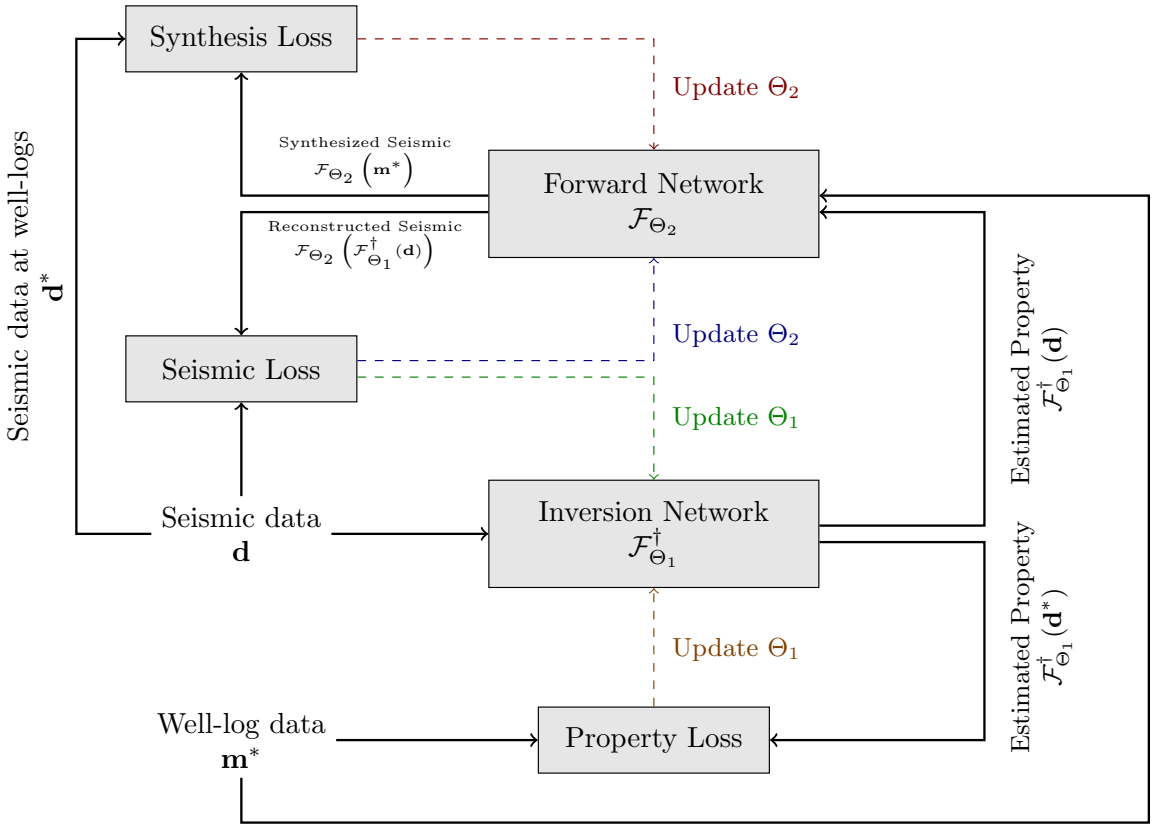


Figure 5.2: The proposed semi-supervised lithology characterization framework with a learned forward model.

The proposed framework learns from a three-term loss function that is used to update the parameters of the inverse and forward model simultaneously. The proposed loss is given as follows:

$$\mathcal{L}(\Theta) := \alpha \cdot \underbrace{\mathcal{L}_1(\Theta_1)}_{\text{property loss}} + \beta \cdot \underbrace{\mathcal{L}_2(\Theta_1, \Theta_2)}_{\text{seismic loss}} + \gamma \cdot \underbrace{\mathcal{L}_3(\Theta_2)}_{\text{synthesis loss}}, \tag{5.2}$$

$$\mathcal{L}_1(\Theta_1) := \mathcal{D}\left(\mathbf{m}^*, \mathcal{F}^{\dagger}_{\Theta_1}(\mathbf{d}^*)\right), \tag{5.3}$$

$$\mathcal{L}_2(\Theta_1, \Theta_2) := \mathcal{D}\left(\mathcal{F}_{\Theta_2}\left(\mathcal{F}^{\dagger}_{\Theta_1}(\mathbf{d})\right), \mathbf{d}\right), \tag{5.4}$$

$$\mathcal{L}_3(\Theta_2) := \mathcal{D}\left(\mathcal{F}_{\Theta_2}(\mathbf{m}^*), \mathbf{d}\right), \tag{5.5}$$

where $\alpha, \beta, \gamma \in \mathbb{R}$ are tuning parameters that govern the influence of each of the three terms of the loss function. $\mathcal{F}_{\Theta_2}$ is a learned forward mapping that can be used to synthesize seismograms from the property traces, and $\mathcal{F}^{\dagger}_{\Theta_1}$ is the learned inverse mapping.

The property loss is computed over the traces for which we have access to rock property measurements form well-logs, $\mathbf{d}^*, \mathbf{m}^*$. The property loss updates the parameters of the inversion network only since it is independent of the forward network. The seismic loss, on the other hand, is computed over all traces in the survey by comparing reconstructed seismic traces from the estimated rock property with the input seismic traces. The seismic loss, which is the cycle inconsistency loss, updates the parameters of the inverse and the forward networks because it is a function of both. Finally, the synthesis loss is computed between the input seismic traces and the synthesized seismic traces from well-logs. The synthesis loss is not a function of the inversion network. Hence, it is only used to update the parameters of the forward network.

In this work, we used MSE loss for all three terms, and we use $\alpha = \beta = \gamma = 1$. Thus, (5.2) can be re-written as follows:

$$\mathcal{L}(\Theta) := \frac{1}{N^*}\sum_i \left\| \mathcal{F}^{\dagger}_{\Theta_1}(\mathbf{d}^*_i) - \mathbf{m}^*_i \right\|_2^2 + \frac{1}{N}\sum_i \left\| \mathcal{F}_{\Theta_2}\left(\mathcal{F}^{\dagger}_{\Theta_1}(\mathbf{d}_i)\right) - \mathbf{d}_i \right\|_2^2 + \frac{1}{N^*}\sum_i \left\| \mathcal{F}_{\Theta_2}(\mathbf{m}^*_i) - \mathbf{d}^*_i \right\|_2^2, \tag{5.6}$$

where $\mathbf{d}$, is the seismic data, $\mathbf{m}^*$ and $\mathbf{d}^*$ are rock property traces from well-logs and their corresponding seismic data, respectively, N^*, and N is the number of samples in $\mathbf{d}^*$, and $\mathbf{d}$, respectively.

The proposed framework offers two advantages over the framework presented in Chapter 4. First, it enables learning from unlabeled data for which a forward model can not be easily defined. Second, it requires no modeling of the data such as estimating wavelet for seismic data. The forward model is replaced with a neural network (the forward network) to learn the forward model from the training data. However, the inversion network in the proposed framework is identical to the one used in Chapter 4.

5.3.1 Learning the Forward Model

The seismic forward model, defined in Section 4.2.2, is comprised of two operations. The first operation computes the reflection coefficients from input elastic or acoustic impedance. Then, the next step convolves the reflection coefficients with a predefined wavelet. In order to use a neural network to approximate the forward model, we use a 3-layer neural network. The choice of a shallow network is because of the simple function of the forward model compared to the inverse function. Furthermore, since the seismic response is local, unlike rock properties, we use only convolutional layers in the forward network. The proposed forward network is shown in Figure 5.3.

Similar to the predefined forward model, the proposed forward network consists of two modules. The first module is the feature extraction module which comprises two

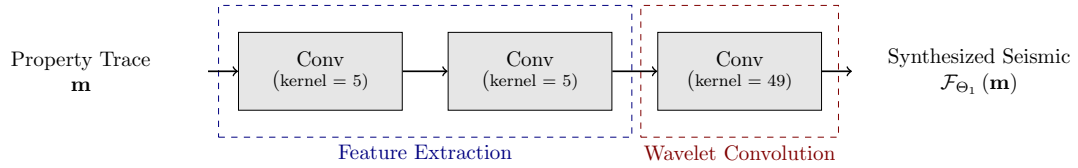


Figure 5.3: The architecture of the forward network in the semi-supervised lithology characterization framework.

convolutional blocks (Conv). A convolutional block consists of a convolutional layer followed by a batch normalization layer and a ReLU activation. We choose a small kernel size ($k = 5$) for the convolutional layers in the features extraction. The choice of the small kernel is to enable the feature extraction module to compute local features that resemble the reflection coefficients computation (high-pass filter). The other module is the wavelet convolution module which consists of a single convolutional block with a kernel of size 50 (low-pass filter). The large-kernel-convolution enables the forward network to learn the characteristics of the wavelet.

The proposed forward network is applicable to single- and multi-angle synthesis of seismograms from acoustic or elastic impedance, respectively. The only parameter to be adjusted in the number of input and output channels of the network. Furthermore, the proposed forward network can be used to synthesize seismic from properties other than acoustic and elastic impedance where a forward model is not well defined.

5.4 Case Study: Acoustic Impedance Estimation

We evaluate the semi-supervised lithology characterization framework with a learned forward model on a case study of acoustic impedance estimation from post-stack seismic data from Marmousi 2 model. We follow a similar procedure in this case study as the one detailed in Section 4.4. The proposed framework is trained on a zero-offset synthetic seismic data with 15 dB Gaussian noise and a 10 uniformly sampled acoustic impedance traces with a resolution that is six times higher than that of the seismic traces. The trained network is then used to estimate the acoustic

impedance for all seismic traces in the model.

We show the results obtained using the proposed framework with the results presented in Chapter 4 using classical approaches in addition to using the proposed framework with a predefined forward model in Figure 5.4. The results show that using a learned forward model gives a similar estimate of the impedance to one obtained using a predefined forward model. The forward network is able to learn the behavior of the forward model without being explicitly modeled as in the case of using a predefined forward model.

To quantitatively evaluate the results, we compute **PCC** and $\mathbf{r}^2$ for each trace in estimation acoustic impedance section. The results, averaged over all traces in the section, are reported in Table 5.1. We also report the metrics computed on the smooth initial model used for linear inversion approaches. The results in Table 5.1 indicate that the proposed framework with a learned forward model resulted in better estimation compared to classical approaches. The results, however, are slightly worse than the ones obtained using the predefined forward model. Note that in the case of a predefined forward model, it was assumed that the seismic wavelet is known. Therefore, we do not expect the learned model to work as well as the predefined one in cases where the predefined forward model matches the ones used to generate the data. However, producing comparable results indicates that the learned forward model is able to learn the true forward model. The distributions of **PCC** and $\mathbf{r}^2$ values over all traces are shown in Figure 5.5 which indicate that using the proposed framework (with a predefined or a learned model) results in better overall metrics values for all traces.

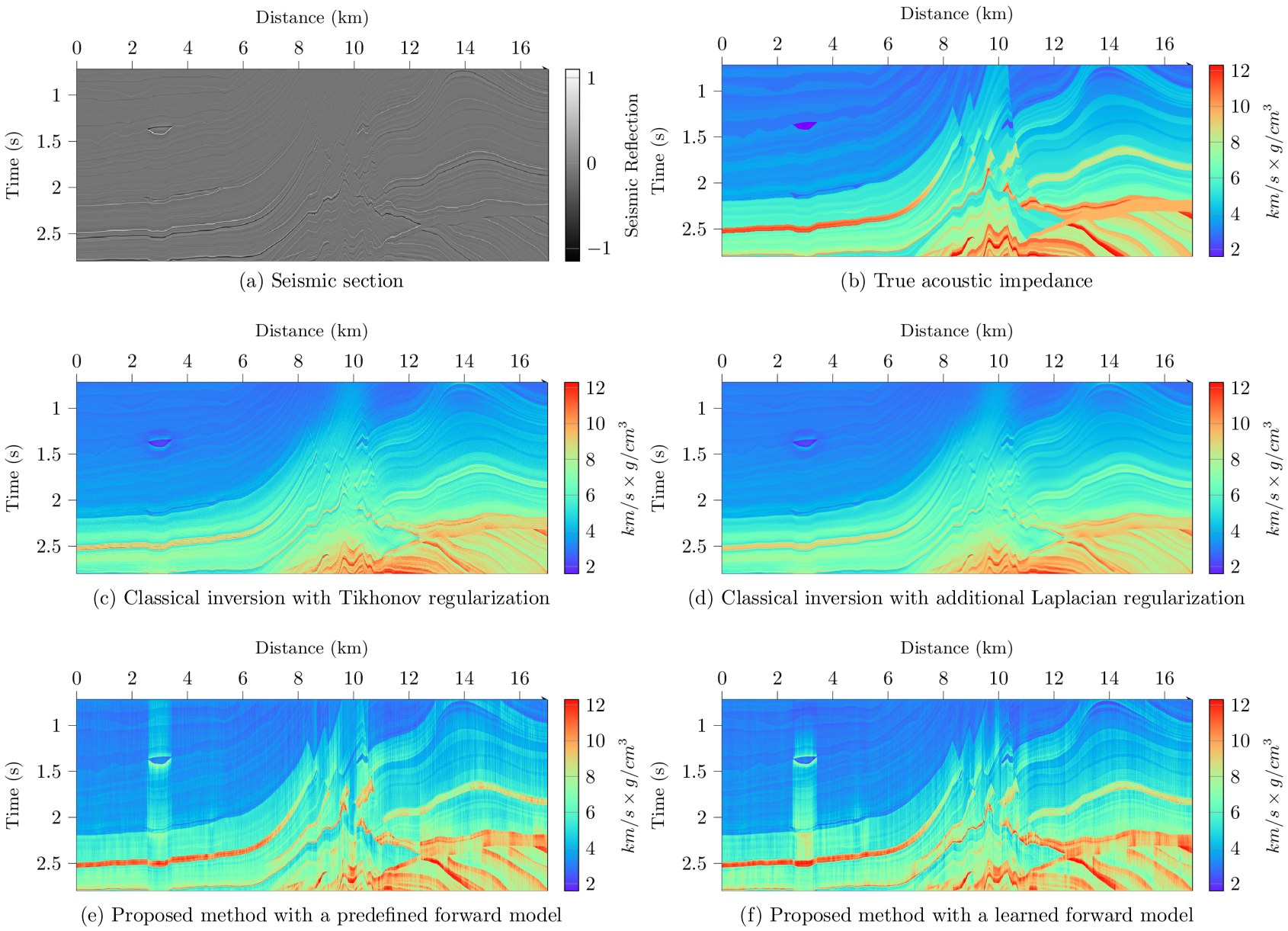
We also show scatter plots of the true acoustic impedance versus the estimated acoustic impedance in Figure 5.6. The figure shows that the estimated acoustic impedance using the learned forward model is linearly correlated with the true acoustic impedance. This is true even for high-impedance values which are not well esti-

Table 5.1: Quantitative evaluation of the estimated acoustic impedance from synthetic seismic data with a 15dB noise.

Method	**PCC**	$\mathbf{r}^2$
Smooth model	0.8957	0.7895
Inversion (Tikhonov)	0.9457	0.8890
Inversion (Tikhonov + Laplacian)	0.9534	0.9008
Ours (predefined)[124]	**0.9701**	**0.9158**
Ours (learned)[123]	0.9656	0.9020

mated using the classical inversion approaches. However, we can see in the figure that low-impedance points are not well estimated using the learned forward model. These points are mostly the points close to the gas-charged pocket ($t = 1.4s,\ x = 3km$). The gas-charged pocket is an anomalous phenomenon in the data that is not present in the training data. Since we are using a recurrent neural network, the error is expected to propagate to nearby samples.

Figure 5.7 shows selected traces from the estimated acoustic impedance section using the two classical inversion approaches and the proposed method with a learned forward model or a predefined forward model. The estimated traces show the advantage of using the proposed workflow, especially for estimating high-contrast impedance values. Both classical inversion approaches do not estimate the acoustic impedance values correctly for thin layers. However, the proposed frameworks (learned and predefined) provide better estimates of these layers. By examining the trace at $x = 15km$, we can see that the estimated acoustic impedance using the proposed framework the thin layer at $t = 1.6s$ matches the true acoustic impedance with high accuracy. In addition, although the results of using the predefined and learned forward model are very comparable, we do expect the predefined forward model to produce better estimates since it is using the true forward model that describes the data.



(a) Seismic section

(b) True acoustic impedance

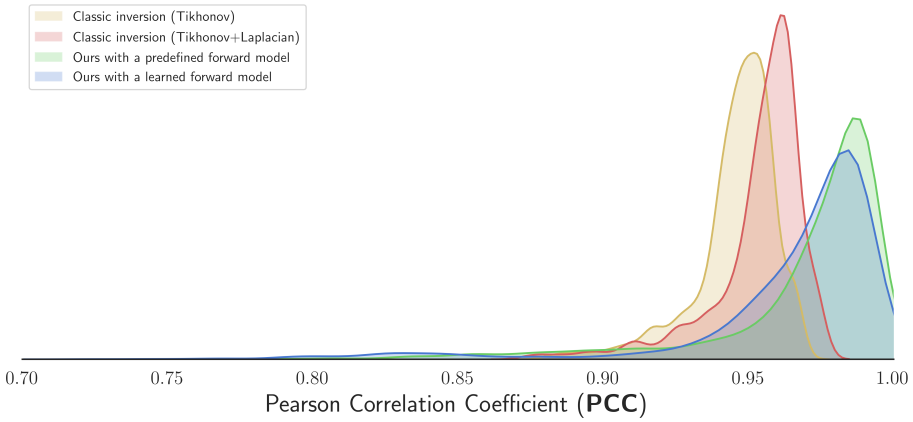
(c) Classical inversion with Tikhonov regularization

(d) Classical inversion with additional Laplacian regularization

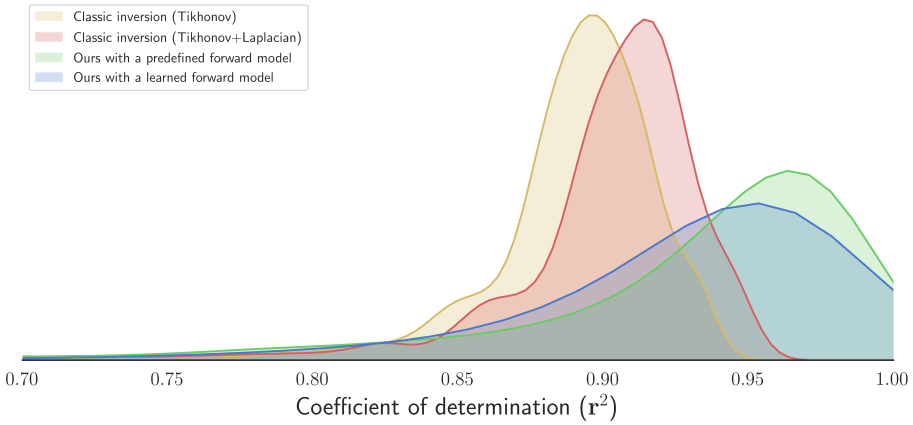
(e) Proposed method with a predefined forward model

(f) Proposed method with a learned forward model

Figure 5.4: Estimated acoustic impedance from a synthetic seismic section with a 15dB noise.

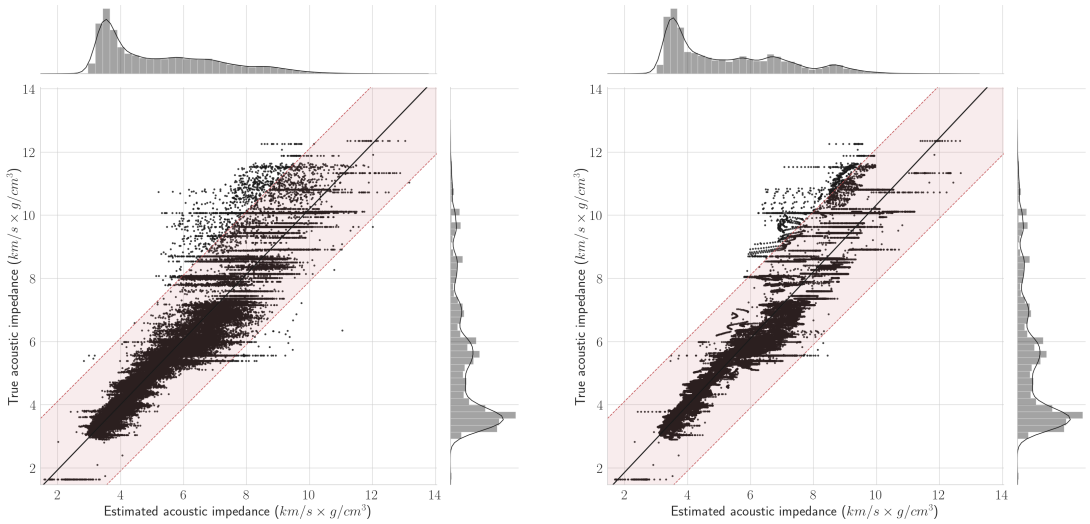


(a) Pearson Correlation Coefficient (**PCC**)



(b) Coefficient of determination ($\mathbf{r^2}$)

Figure 5.5: The distribution of the **PCC** and $\mathbf{r}^2$ for the estimated acoustic impedance from synthetic seismic data.



(a) Inversion (Tikhonov)

(b) Inversion (Tikhonov+Laplacian)

(c) Ours with a predefined forward model

(d) Ours with a learned forward model

Figure 5.6: Scatter plots of the true acoustic impedance versus the estimated acoustic impedance. The shaded region include all points that are within one standard deviation of the true acoustic impedance. The black line is the best linear fit.

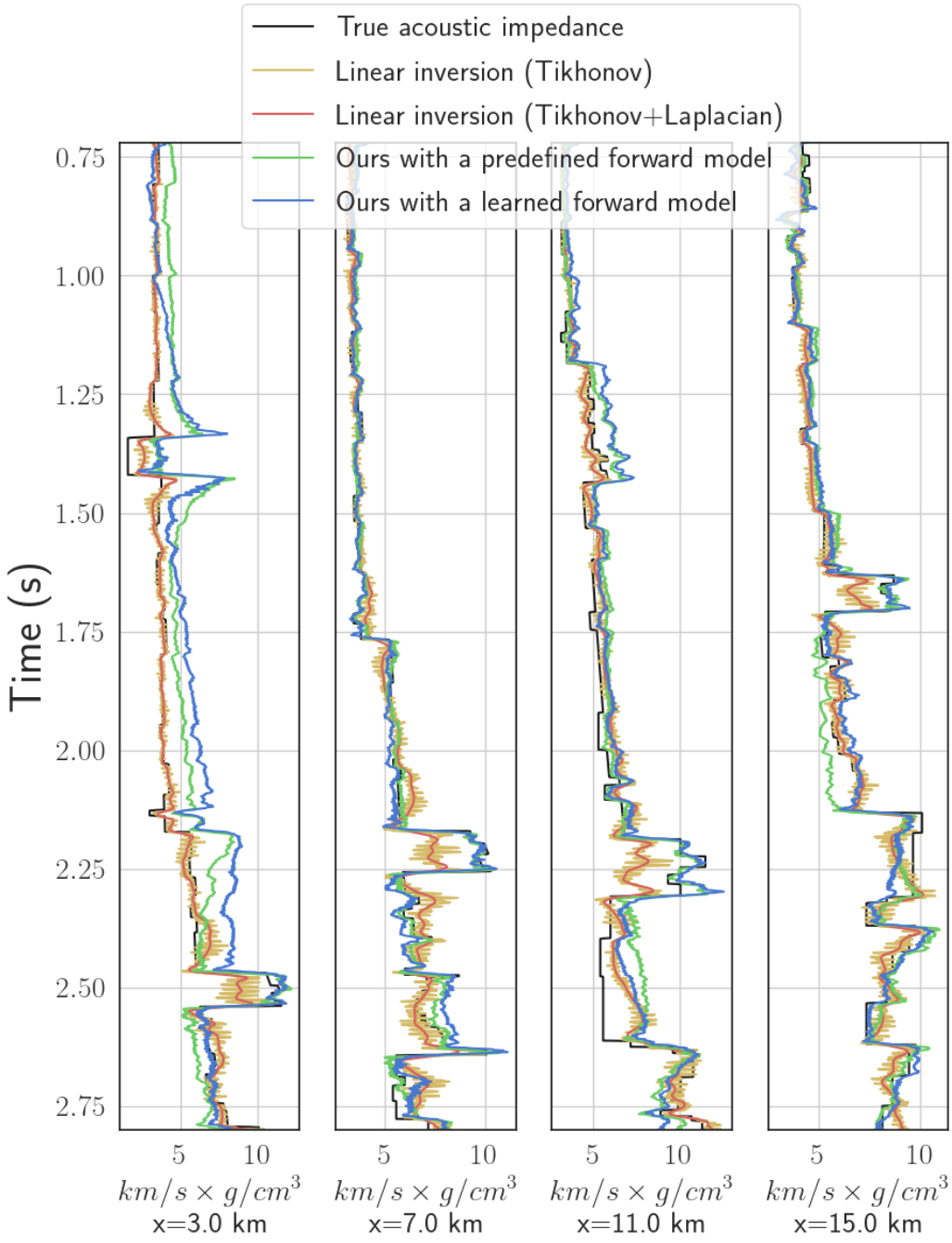


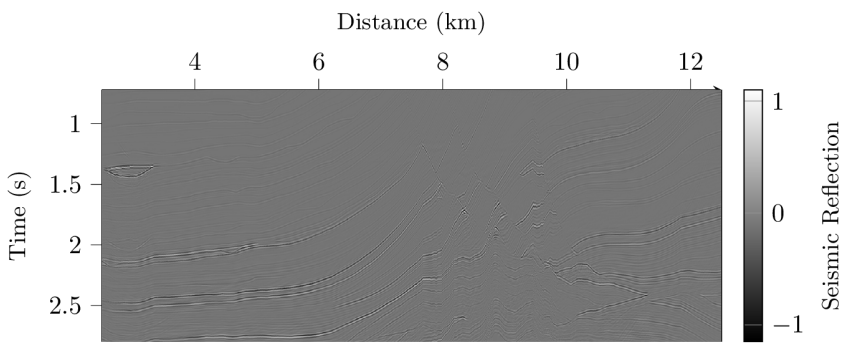
Figure 5.7: Selected traces from the estimated acoustic impedance section.

To further analyze the performance of the proposed framework, we repeat the experiments using wave-equation-migrated seismic data instead of noisy synthetic seismic data. In our experiments, classical inversion approaches did not converge to a solution using migrated data. Thus, we show the results obtained using the proposed framework for both cases of a predefined and a learned forward model in Figure 5.8. In the case of wave-equation-migrated seismic, the predefined forward model was misleading the inversion network since the seismic data did not match the synthesized seismic. Thus, we limited the role of the predefined forward model by using a small weight for the seismic loss. On the other hand, the learned forward network can learn how to synthesize seismic data that matches the input seismic using the seismic loss and the synthesis loss. Thus, we use equal weights for all three terms of the loss function ($\alpha = \beta = \gamma = 1$). The acoustic impedance estimated with a learned forward model has a better resolution of the layers, although they were not clearly visible in the seismic data. This improvement is clear in the faulty region on the model ($x = 8km$), and in the gas-charged pocket ($t = 1.4s,\ x = 3km$). The quantitative results (reported in Table 5.2) also reflect the gains of using the learned forward model. Both **PCC** and $\mathbf{r}^2$ improved by about 2.7% compared to results with a predefined forward model. The distributions of **PCC** and $\mathbf{r}^2$ values over all traces are shown in Figure 5.9.

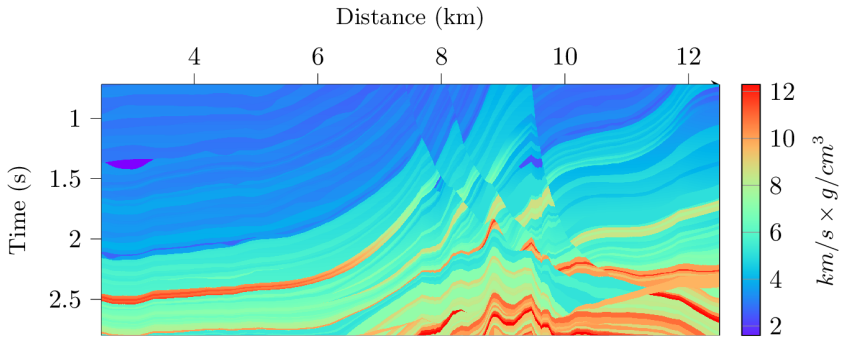
Table 5.2: Quantitative evaluation of the estimated acoustic impedance from migrated seismic data.

Method	**PCC**	$\mathbf{r}^2$
Ours with a predefined forward model [124]	0.9091	0.7892
Ours with a learned forward model [123]	**0.9321**	**0.8168**

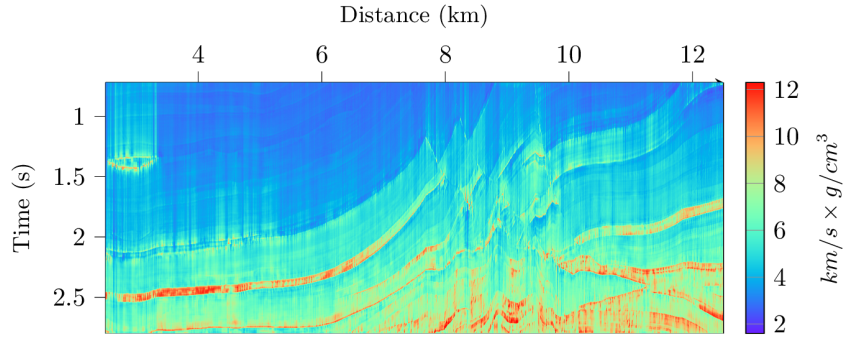
Finally, we show selected traces from the estimated acoustic impedance section using migrated seismic data in Figure 5.10. The figure shows that the estimated traces using a learned forward model are comparable to the estimated traces using



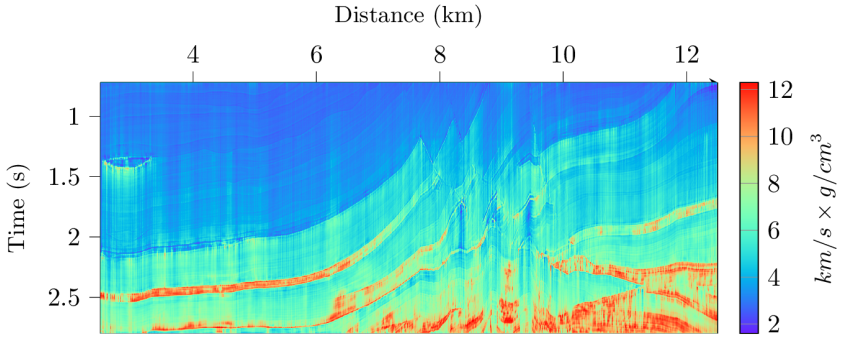
(a) Wave-equation-migrated seismic section



(b) True acoustic impedance

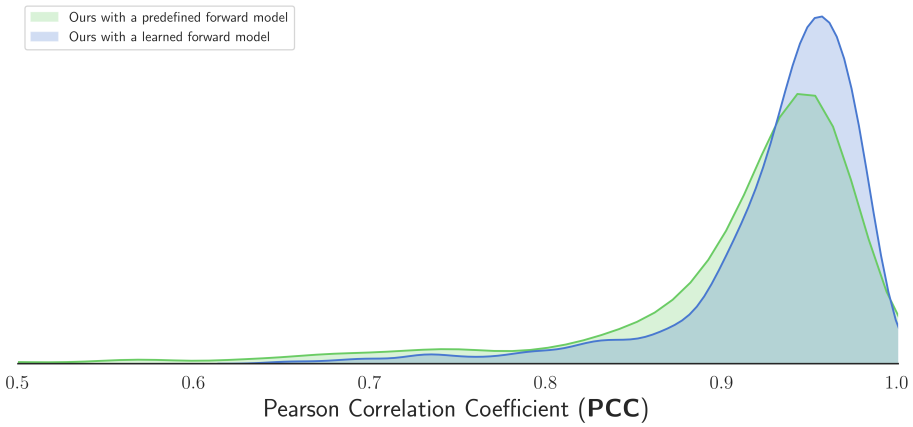


(c) Acoustic impedance using the proposed semi-supervised framework

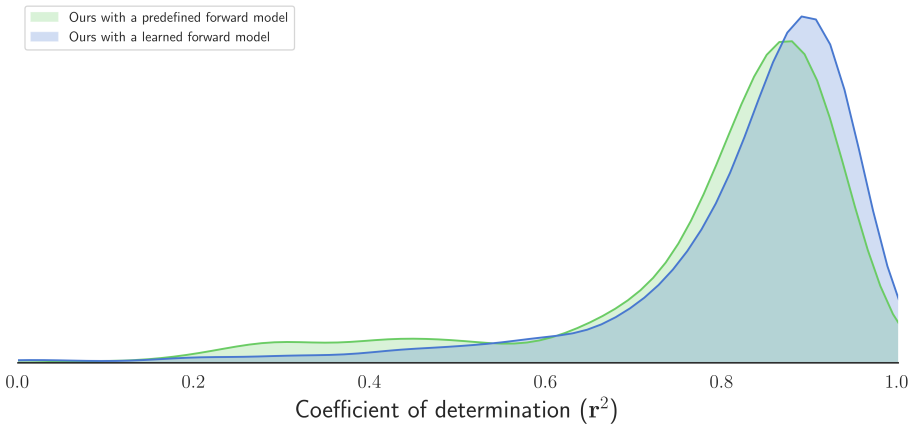


(d) Acoustic impedance using the proposed semi-supervised framework

Figure 5.8: Estimated acoustic impedance from a migrated seismic section.



(a) Pearson Correlation Coefficient (**PCC**)



(b) Coefficient of determination ($\mathbf{r}^2$)

Figure 5.9: The distribution of the **PCC** and $\mathbf{r}^2$ for the estimated acoustic impedance from migrated seismic data.

the predefined model. However, the traces estimated with a learned forward model show a better match to the true acoustic impedance in complex or thin layers. For example, the trace at $x = 3km$ indicated that the forward model is closer to the true impedance values for the areas near the gas-charged pocket ($t$between0.8s and 1.6s). Since both methods use an identical inversion network, we expect the high-frequency components of their estimates to be very similar. However, for the low-frequency components, they have small differences where the learned forward model produces better results.

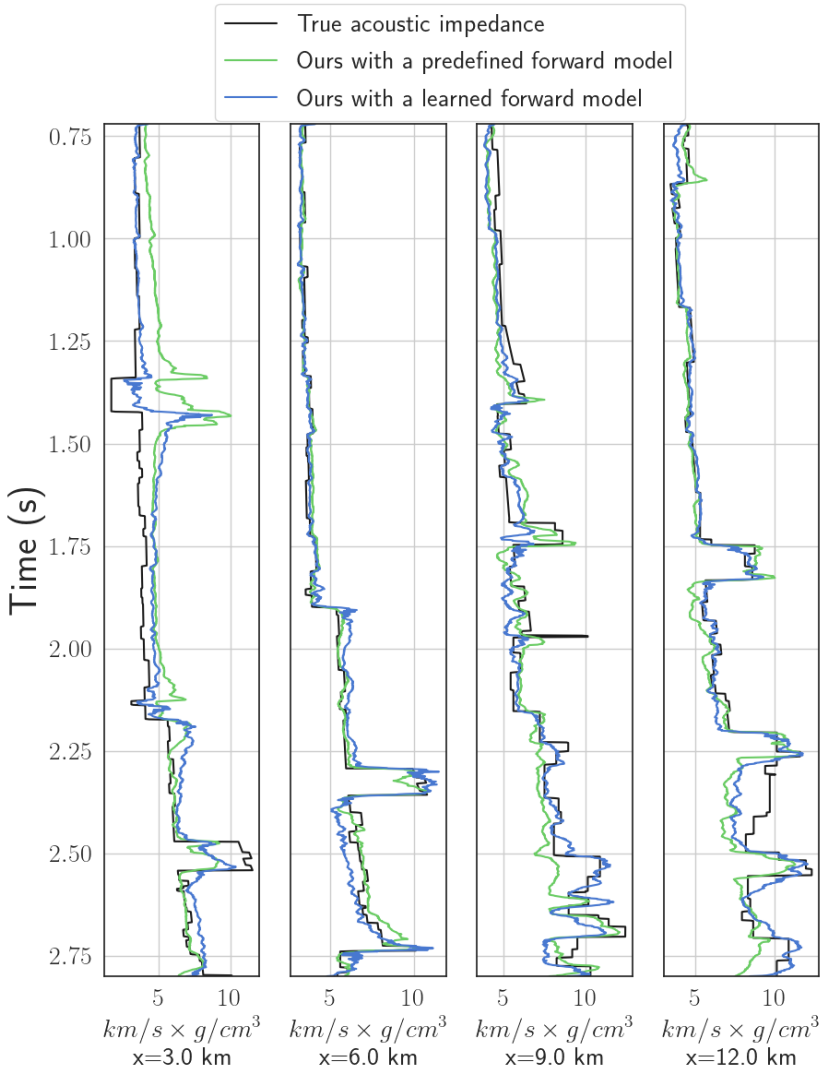


Figure 5.10: Selected traces from the acoustic impedance estimated from section from wave-equation-migrated seismic data.

5.5 Case Study: Elastic Impedance Estimation

We also extend the proposed framework to estimate the elastic impedance from prestack seismic data. We use Synthetic noisy seismic data (SNR=15dB) from Marmousi 2 with incident angles $\theta = 0°, 10°, 20°, \text{ and}, 30°$ to estimate the corresponding elastic impedance. The seismic data and corresponding elastic impedance sections are shown in Figure 4.11. We use 10 uniformly-sampled elastic impedance traces for training. Then, the trained network is used to estimate elastic impedance for all traces in the model. The resolution of the seismic data is six times lower than that of the elastic impedance. The number of input and output channels of both the inverse and forward networks is changed to 4.

Furthermore, we compare the results with results from linear inversion approaches and the proposed framework with a predefined forward model in Chapter 4. The results are shown in Figures 5.11-5.14 for different incident angles.

Similar to the acoustic impedance case, the learned forward model gives a similar high-resolution estimation of the elastic impedance. The quantitative results on **PCC** and $\mathbf{r}^2$ are summarized in Table 5.3.

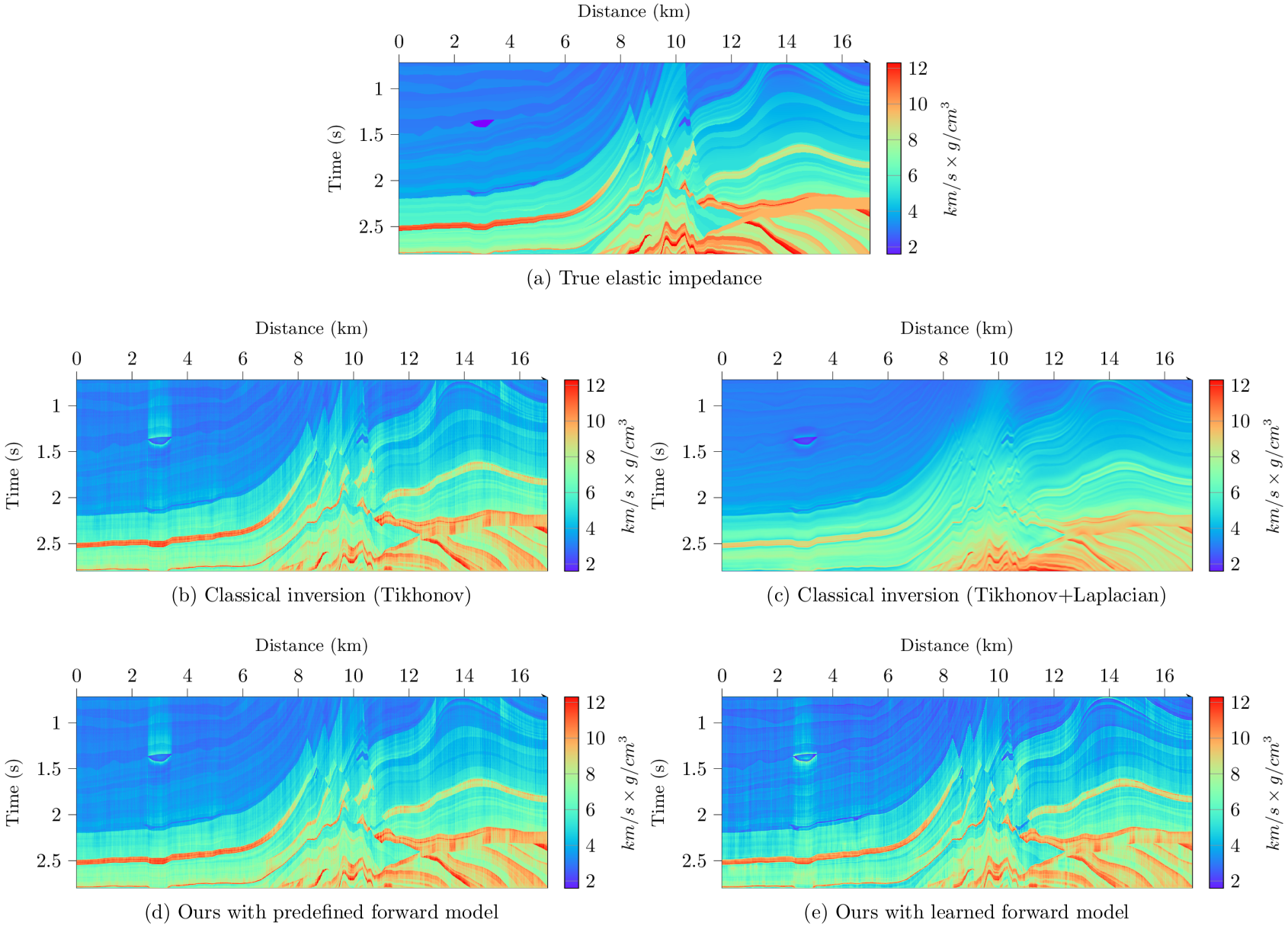
The numbers reported in Table 5.3 show that the proposed framework with a learned forward model performs better than the classic inversion approaches. In addition, its estimates are comparable to the estimates using the predefined forward model. We also show the scatter plots of the true acoustic impedance versus the estimated acoustic impedance in Figure 5.16. Comparing the scatter plot of the learned and predefined forward models, the latter shows a slightly better fit due to the use of the true forward model. Overall, both the learned and predefined forward model show a better fit than the classical inversion methods which is indicated by the large number of points that are within the shaded area of the scatter plot. The distributions of **PCC** and $\mathbf{r}^2$ values over all traces are shown in Figure 5.15.

Selected traces from the estimated elastic impedance sections are shown in Figure 5.17. Similar to the results presented in Section 5.4, the estimated traces using both the learned and the predefined forward models match the true elastic impedance better than the classical inversion approaches, especially in thin layers. Moreover, results using the predefined forward model are better since the true model predefined forward model since we assumed the true wavelet was known for the predefined forward model.

5.6 Summary

In summary, we presented a learning-based semi-supervised lithology characterization that can learn the inverse and forward model using the training data. The use of the learned forward model enables the proposed framework to learn from data without any modeling requirements as in the case of a predefined forward model. The proposed framework learns from a three-term loss function that utilizes all training data in addition to unlabeled seismic data to better estimate the true lithology of the subsurface. The proposed framework was evaluated in two case studies for acoustic and elastic impedance inversion from noisy synthetic seismic data. The semi-supervised framework was trained on 10 impedance traces, and was then used to estimate the impedance for the entire section. Furthermore, the proposed framework was evaluated on migrated seismic data to estimate the acoustic impedance. Results obtained using the introduced framework with a learned forward model were close to results obtained using a predefined forward model. The predefined forward model worked better in cases where the forward model accurately described the data (i.e., synthetic seismic data). However, when the wavelet is unknown or modeled incorrectly, the learned forward model produces better results as in the case of migrated seismic data. Although we have used the proposed framework for acoustic and impedance estimation, it can be used to estimate any given rock property from seismic data given that a few

training examples are available.



(a) True elastic impedance

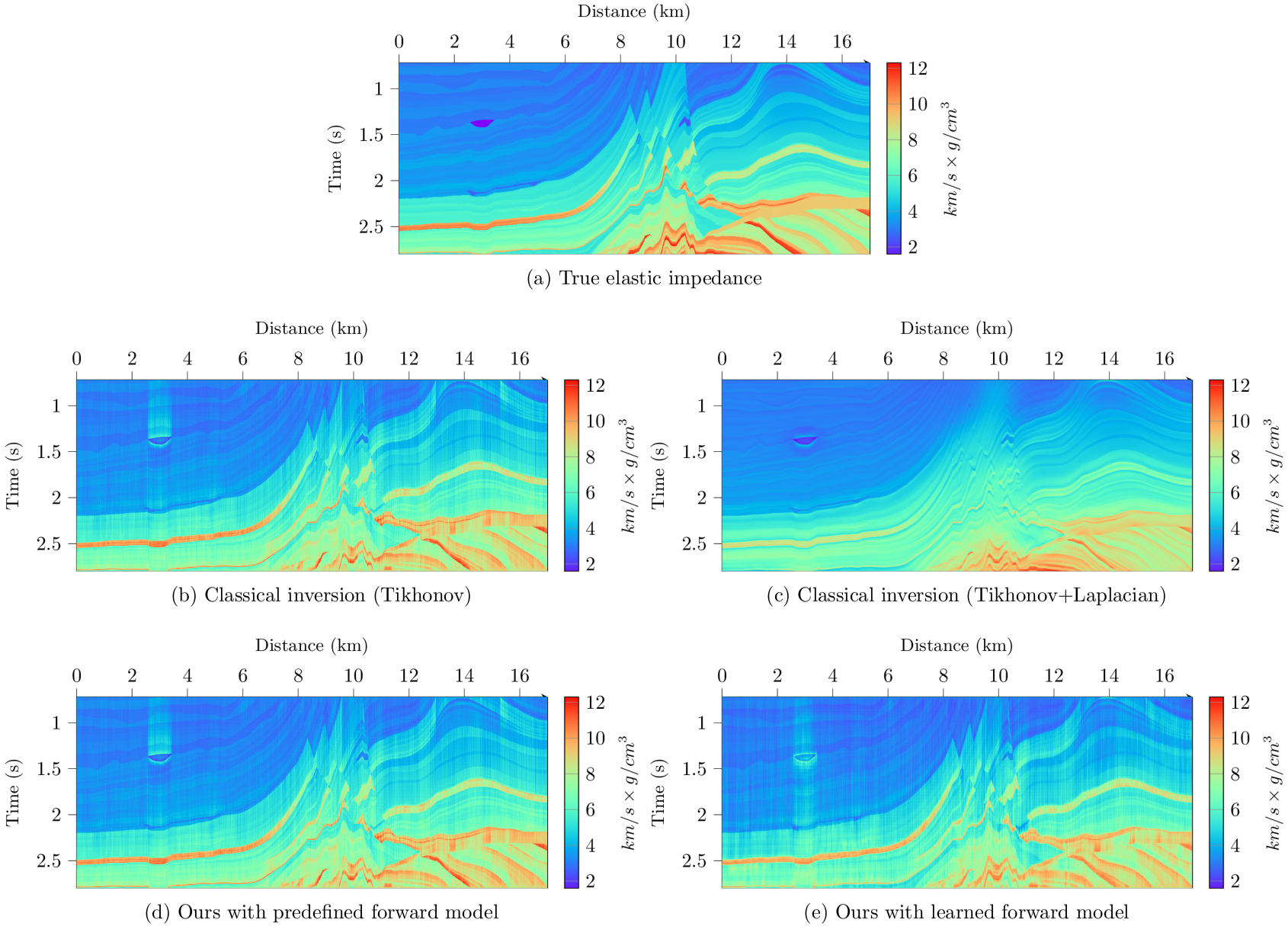
(b) Classical inversion (Tikhonov)

(c) Classical inversion (Tikhonov+Laplacian)

(d) Ours with predefined forward model

(e) Ours with learned forward model

Figure 5.11: Estimated elastic impedance ($\theta = 0°$) from synthetic seismic data with a 15dB noise.



(a) True elastic impedance

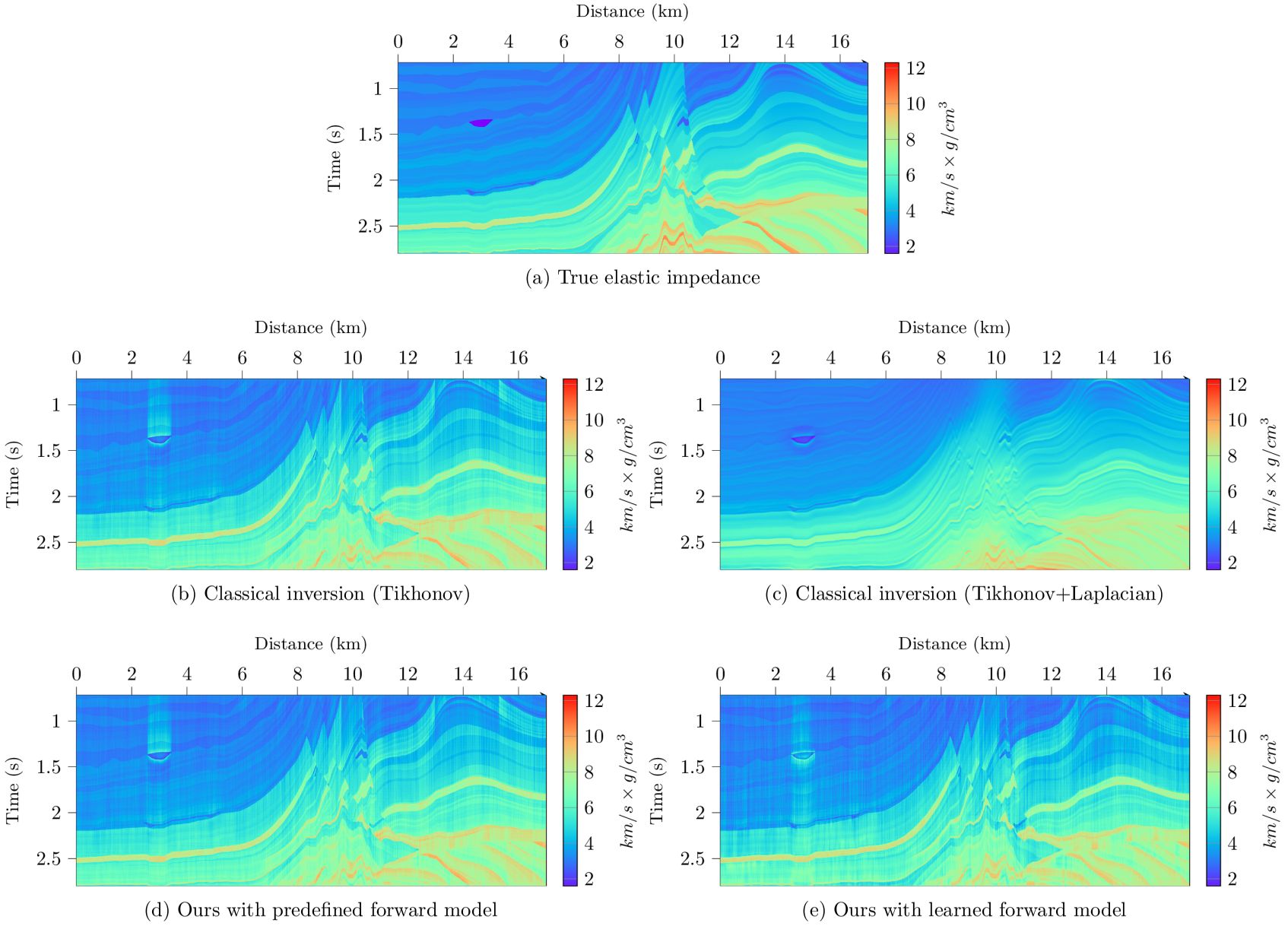
(b) Classical inversion (Tikhonov)

(c) Classical inversion (Tikhonov+Laplacian)

(d) Ours with predefined forward model

(e) Ours with learned forward model

Figure 5.12: Estimated elastic impedance ($\theta = 10°$) from synthetic seismic data with a 15dB noise.



(a) True elastic impedance

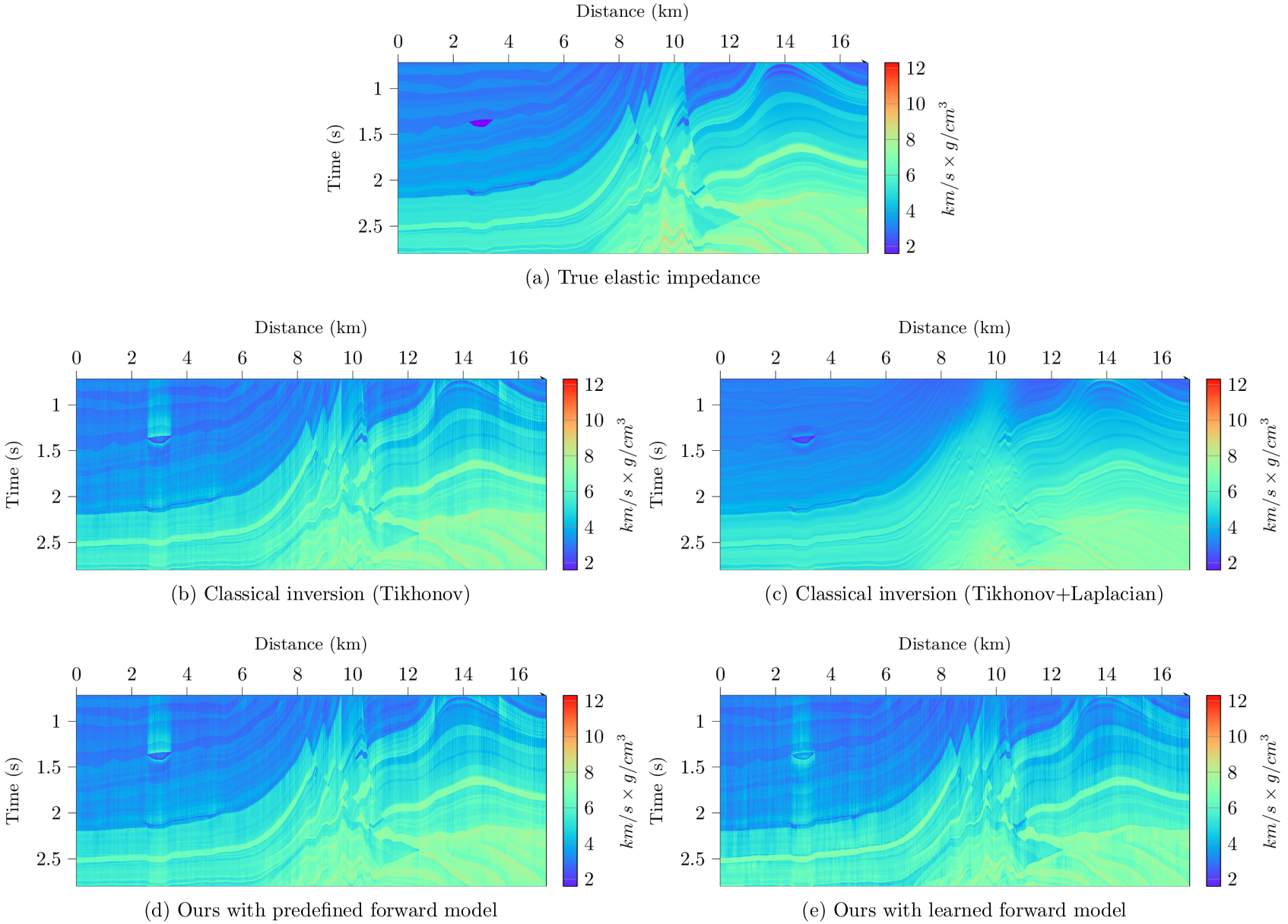
(b) Classical inversion (Tikhonov)

(c) Classical inversion (Tikhonov+Laplacian)

(d) Ours with predefined forward model

(e) Ours with learned forward model

Figure 5.13: Estimated elastic impedance ($\theta = 20°$) from synthetic seismic data with a 15dB noise.



(a) True elastic impedance

(b) Classical inversion (Tikhonov)

(c) Classical inversion (Tikhonov+Laplacian)

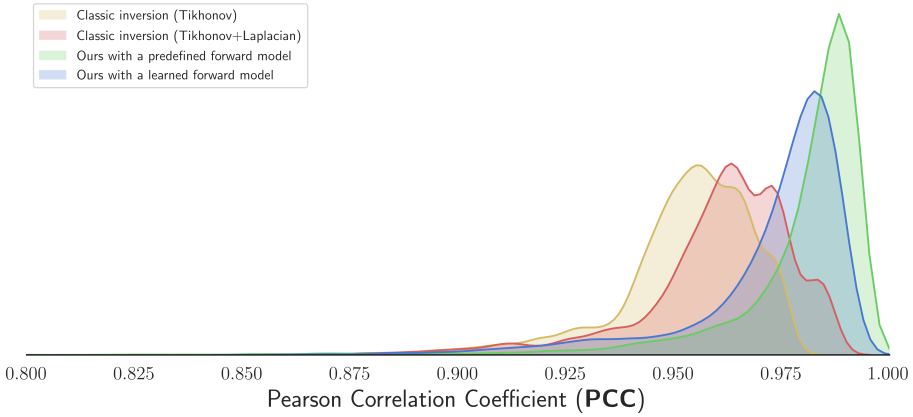
(d) Ours with predefined forward model

(e) Ours with learned forward model

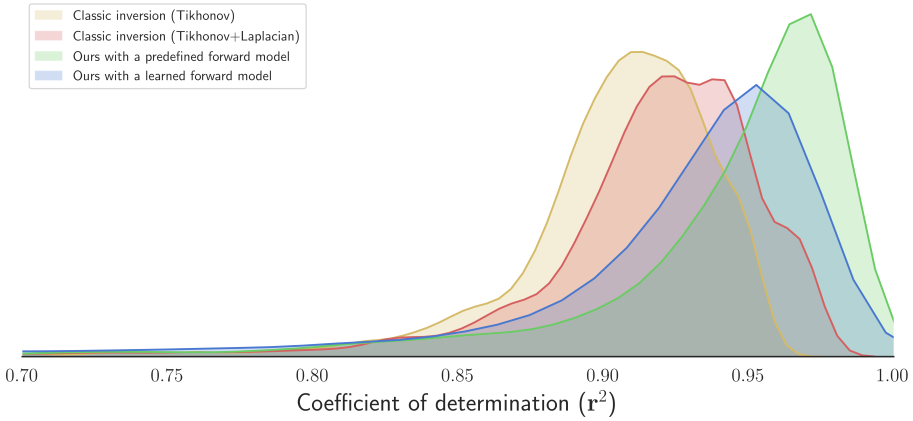
Figure 5.14: Estimated elastic impedance ($\theta = 30°$) from synthetic seismic data with a 15dB noise.

Table 5.3: Quantitative evaluation of the estimated elastic impedance from synthetic multi-angle seismic data with a 15dB noise.

	PCC					**r^2**				
Method	0°	10°	20°	30°	Average	0°	10°	20°	30°	Average
Smooth model	0.9101	0.9163	0.9325	0.9472	0.9265	0.8212	0.8329	0.8635	0.8899	0.8519
Inversion (Tikhonov)	0.9467	0.9494	0.9564	0.9612	0.9534	0.8907	0.8962	0.9096	0.9169	0.9034
Inversion (Tikhonov + Laplacian)	0.9534	0.9562	0.9635	0.9698	0.9607	0.9007	0.9067	0.9218	**0.9331**	0.9156
Ours with a predefined forward model [124]	**0.9789**	**0.9801**	**0.9817**	**0.9775**	**0.9795**	**0.9425**	**0.9434**	**0.9409**	0.9189	**0.9364**
Ours with a learned forward model [123]	0.9702	0.9735	0.9766	0.9702	0.9726	0.9246	0.9279	0.9252	0.8790	0.9142

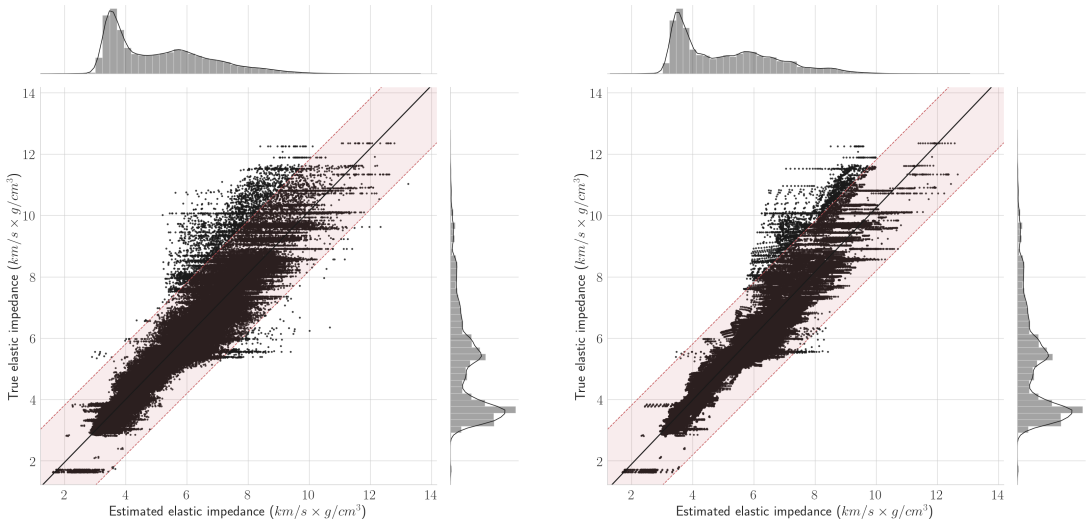


(a) Pearson Correlation Coefficient (**PCC**)



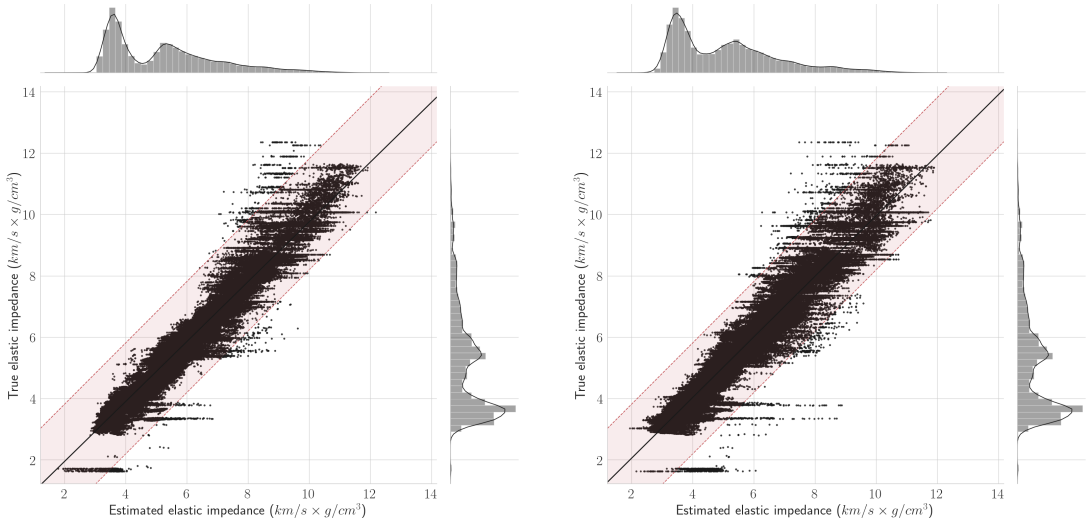
(b) Coefficient of determination ($\mathbf{r}^2$)

Figure 5.15: The distribution of **PCC** and $\mathbf{r}^2$ values for the estimated elastic impedance from synthetic multi-angle seismic data.



(a) Inversion (Tikhonov)

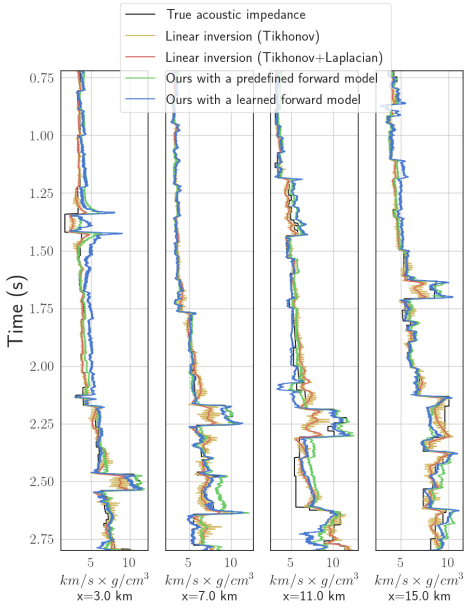
(b) Inversion (Tikhonov+Laplacian)



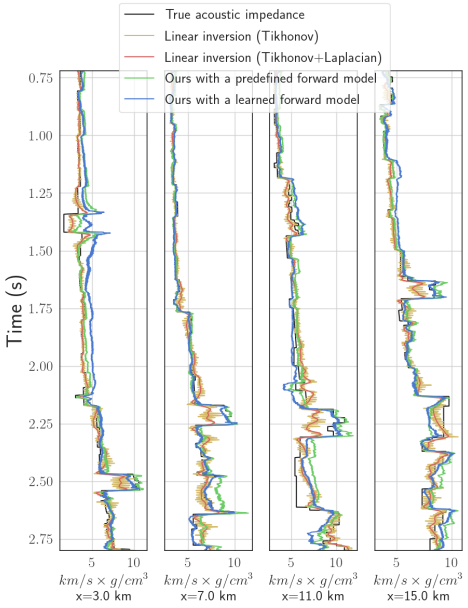
(c) Ours with a predefined forward model

(d) Ours with a learned forward model

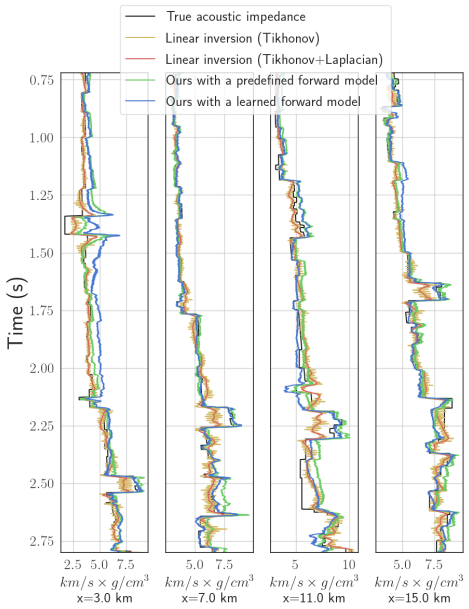
Figure 5.16: Scatter plots of the true elastic impedance versus the estimated elastic impedance. The shaded region include all points that are within one standard deviation of the true acoustic impedance. The black line is the best linear fit.



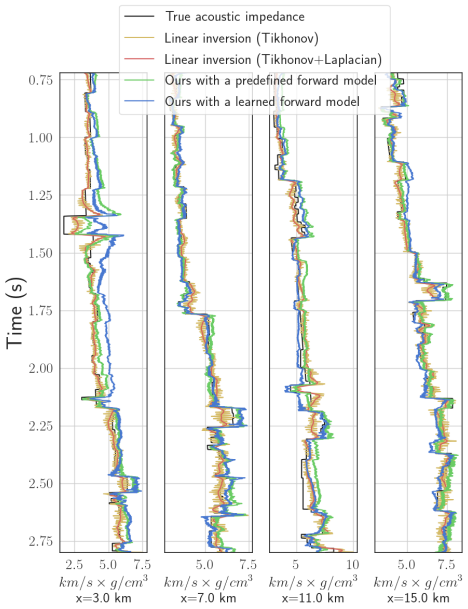
(a) $\theta = 0°$



(b) $\theta = 10°$



(c) $\theta = 20°$



(d) $\theta = 30°$

Figure 5.17: Selected traces from the estimated elastic impedance sections for all incident angles.

CHAPTER 6
CONCLUSION

Subsurface volume characterization is a labor-intensive and time-consuming process through which various geological events are inferred from indirect measurements of the subsurface properties. The exponential growth of collected data from different seismic surveys makes it impossible for interpreters to manually inspect, analyze and annotate all collected data. While deep learning has proved to be a viable solution for big data problems in various computer vision tasks such as image classification, image captioning, and semantic segmentation, deep learning applications are limited in the field of subsurface volume characterization due to the limited availability of large annotated seismic datasets. Obtaining such annotations by manual interpretation is a labor-intensive process that requires field knowledge. Also, due to the limited resolution of the seismic data, seismic interpreters might arrive at different interpretations of the subsurface that are valid (i.e., they agree with the data). In addition, obtaining such annotations from true measurements of the subsurface (e.g., well-log data) is expensive due to the high costs of drilling. Furthermore, since deep neural networks are mainly data-driven, they might fail to generalize and adequately represent noisy and low-resolution data such as seismic data. Injecting the physics governing the seismic data allows deep neural networks to learn a geologically valid representation of the data. Also, it enables deep neural networks to learn from unlabeled data. Therefore, to successfully utilize deep learning methods in this field, one must address and circumvent the lack or shortage of annotated consistent datasets. Therefore, in this dissertation, we have presented a physics-guided learning-based subsurface volume characterization framework that can learn from a few annotated data samples including manually annotated examples or true subsurface measurements such as well-logs

and core data. First, we presented a framework to characterize seismic images from a structural point of view using multiresolution texture analysis. In this framework, we introduced a novel texture similarity measure that was used to obtain a large dataset of labeled seismic images from an unlabeled seismic volume and a few labeled examples. The obtained dataset was then used to train a machine learning framework to characterize a large seismic volume from a structural point of view. Furthermore, we presented a semi-supervised sequence modeling framework that models seismic data as time series using recurrent neural networks. The introduced framework uses the sparsely-available rock property measurements from well-logs as labels for seismic data. In addition, it utilizes physical seismic models to learn from unlabeled seismic data. The trained framework can then be used to estimate rock properties for the entire seismic volume. The presented framework, unlike classical inversion methods, does not require an initial model to be manually estimated. Instead, it can infer such a model from the training data. Furthermore, the introduced framework learns to map low-resolution seismic traces to high-resolution rock property traces using a learned upscaling network. It was shown to outperform classical seismic inversion frameworks in case studies of elastic and acoustic impedance estimation from noisy synthetic and migrated seismic data. Finally, we presented an end-to-end learning-based semi-supervised sequence modeling framework for lithology characterization framework that can also learn the seismic physical model from the training data instead of being explicitly modeled. The presented framework was shown to be robust to noise and artifacts that are often present in migrated seismic data.

6.1 Contributions

The main contributions of this dissertation can be summarized as follows:

- A state-of-the-art texture-based image similarity measure for seismic structures. We show that the introduced similarity measure can characterize seismic struc-

tures present in seismic images with high accuracy. In addition, we show that the introduced similarly measure maintains a robust performance across different datasets of natural and synthetic texture images in addition to seismic images.

- A novel method to model seismic traces as sequential data using recurrent neural networks. We show that modeling seismic traces as time series enable neural networks to learn global and local features which play a key role in lithology characterization of subsurface volumes. We also show that recurrent neural networks outperform feed-forward ones such as convolutional neural networks to characterize seismic data from a lithology point of view.

- A semi-supervised physics-guided deep sequence modeling lithology characterization framework. The framework uses well-log data as labels for the seismic data In addition, the physical seismic forward model enables neural networks to learn from unlabeled seismic data. The introduced framework learns to map low-resolution seismic data to high-resolution lithology characterization of the subsurface using a specialized upscaling neural network. We show that the introduced framework outperforms classical seismic inversion methods without requiring low-frequency content of the true lithology of the subsurface as in the case of classical inversion.

- A semi-supervised deep sequence modeling lithology characterization framework with a learned forward model. The introduced framework requires no physical modeling of the seismic data. Instead, it learns the inverse and forward models from seismic data to rock properties from the limited well-logs as annotations for seismic data. We have shown that it can estimate rock properties from seismic data even with the presence of noise and migration artifacts.

6.2 Future Research Directions

In this dissertation, we have focused on applications of image processing and deep learning for subsurface volume characterization with the limited availability of labeled data. In light of what we have presented in this dissertation, we identify promising research directions that can bridge the gap between the fields deep learning and subsurface volume characterization. These research directions are summarized next.

- In Chapter 2, we focused on 2-dimensional image-based characterization of seismic data. However, since the structures of the subsurface are 3-dimensional, it is expected that 3-dimensional characterization can help identify these structures better. Furthermore, the use of multi-angle (prestack) seismic data adds another dimension to the three spatial ones. Prestack seismic data reveal details that are not visible in the poststack data. This can be achieved using high-dimensional signal processing and analysis tools.

- In this thesis, we utilized 1-dimensional recurrent neural networks as sequence models for seismic data. Although the results obtained using the introduced framework outperform classical methods, incorporating spatial constraints in the introduced framework would improve the lateral resolution of the results. This can be achieved by incorporating multi-dimensional recurrent neural networks or a combination of 3-dimensional convolutional neural networks with 1-dimensional sequence models.

- In Chapter 4, we introduced the use of a physical forward model to enable the neural network to learn from unlabeled data. However, the forward model we used was a relatively simple one that might not describe the data accurately. The forward model must be fast to keep up with the deep neural inversion network. Accurate forward models such as wave-propagating simulation are

computationally expensive. A fast physics- or learning-based realistic forward models can be easily used in the presented semi-supervised framework, which would help the neural network to learn to characterize seismic data more accurately.

- The applications of the introduced semi-supervised sequence modeling framework presented in Chapters 4 and 5 are not only limited to lithology characterization of subsurface volumes. It can be applied for structure and stratigraphy characterization of the subsurface.

- The introduced physics-based semi-supervised framework can be used to inject physical knowledge to solve various problems in a learning-based paradigm. This includes applications of geothermal exploration, medical imaging, metrical characterization, and structure integrity testing.

CHAPTER 7
THESIS PRODUCTS

7.1 Magazines Articles

1. G. AlRegib, M. Deriche, Z. Long, H. Di, Z. Wang, Y. Alaudah, M. A. Shafiq, and **M. Alfarraj**, "Subsurface Structure Analysis Using Computational Interpretation and Learning," in IEEE Signal Processing Magazine, vol. 35, no. 2, pp. 82-98, Mar. 2018.

7.2 Journal Articles

1. Y. Hu, Z. Long, A. Sundaresan, **M. Alfarraj**, G. AlRegib, and S. Jayaraman, "Assessment of Deep Learning-based Texture Representations Using a Challenging Dataset", Submitted to *IEEE Transactions on Multimedia*, Aug 2019.

2. **M. Alfarraj**, and G. AlRegib, "Semi-supervised Sequence Modeling for Elastic Impedance Inversion", in *Interpretation*, Aug 2019.

3. Y. Alaudah, P. Michalowicz, **M. Alfarraj**, G. AlRegib, "A Machine Learning Benchmark for Facies Classification", in *Interpretation*, Aug 2019.

4. **M. Alfarraj**, Y. Alaudah, Z. Long, and G. AlRegib, "Multiresolution Analysis and Learning for Computational Seismic Interpretation," in *The Leading Edge*, 2018.

5. Y. Alaudah, **M. Alfarraj**, and G. AlRegib, "Structure Label Prediction Using Similarity-Based Retrieval and Weakly-Supervised Label Mapping," in *Geophysics*, vol. 84, no. 1, pp. V67-V79, Dec. 2018.

6. H. Di, **M. Alfarraj**, and G. AlRegib, "Three-Dimensional Curvature Analysis of Seismic Waveforms and Its Interpretational Implications," in *Geophysical Prospecting*, vol. 67, no. 2, pp. 265-281, Dec. 2018.

7. Z. Long, Y. Alaudah, M.A. Qureshi, Y. Hu, Z. Wang, **M. Alfarraj**, G. AlRegib, A. Amin, M. Deriche, S. Al-Dharrab, and H. Di, "A Comparative Study of Texture Attributes for Characterizing Subsurface Structures in Seismic Volumes," in *Interpretation*, vol. 6, no. 4, pp. 1055-1066, Oct. 2018.

7.3 Conference Papers

1. **M. Alfarraj** and G. AlRegib, "Semi-Supervised Learning for Acoustic Impedance Inversion," in *Expanded Abstracts of the SEG Annual Meeting*, Sep. 15-20 2019.

2. A. Mustafa, **M. Alfarraj**, and G. AlRegib, "Estimation of Acoustic Impedance From Seismic Data Using Temporal Convolutional Network," in *Expanded Abstracts of the SEG Annual Meeting*, Sep. 15-20 2019.

3. **M. Alfarraj** and G. AlRegib, "Petrophysical Property Estimation From Seismic Data Using Recurrent Neural Networks," in *Expanded Abstracts of the SEG Annual Meeting*, Anaheim, CA, Oct. 14-19 2018.

4. **M. Alfarraj**, H. Di, and G. AlRegib, "Multiscale Fusion for Seismic Geometric Attribute Enhancement," in *Expanded Abstracts of the SEG Annual Meeting*, Houston, Texas, Sep. 24-26 2017.

5. Di, H., **M. Alfarraj**, and G. AlRegib, 2017, "3D Curvature Analysis of Seismic Waveform and Its Interpretational Implications," in E*xpanded Abstracts of the SEG Annual Meeting*, Houston, Texas, Sep. 24-26 2017, pp. 2255-2259.

6. **M. Alfarraj**, H Di, and G. AlRegib, "Multiscale Fusion for Improved Instantaneous Attribute Analysis," in *EAGE Annual Conference & Exhibition, Paris*,

France, Jun. 12-15 2017.

7. **M. Alfarraj**, Y. Alaudah, and G. AlRegib , "Content-Adaptive Non-Parametric Texture Similarity Measure," in *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, Montreal, Canada, Sep. 21-23 2016.

8. Z. Long, Y. Alaudah, M. Qureshi, **M. Alfarraj**, Z. Wang, A. Amin, M. Deriche, and G. AlRegib, "Characterization of Migrated Seismic Volumes Using Texture Attributes: A Comparative Study," in *Expanded Abstracts of the SEG Annual Meeting*, New Orleans, LA, Oct. 18-23 2015.

9. **M. Alfarraj**, N. Keni, and G. AlRegib, "Property Prediction From Seismic Attributes Using a Boosted Ensemble Machine Learning Scheme," in *SBGf/SEG Machine Learning Workshop*, Rio De Janeiro, Brazil, 2018.

7.4 Datasets

1. CoMMonS: Challenging Microscopic Material Surface Dataset, Jul. 2019. [Link]

2. A Machine Learning Benchmark for Facies Classification, Apr. 2019. [Link]

3. LANDMASS: Large North Sea Dataset of Migrated Aggregated Seismic Structures, Dec. 2014. [Link]

7.5 Software

1. Dippykit: A Python Library for Digital Image Processing, Version 2.0.2, Aug. 2018. [Link]

2. CLeVER: CLoud Visual ExploreR, Nov 2017. [Link]

Appendices

APPENDIX A
EVALUATION METRICS

A.1 Image Retrieval

To assess the retrieval performance of a similarity measure, we set each image $\mathbf{x}_i$ in the dataset as a query and use the similarity measure to retrieve the images that are similar to the query image. Then, information retrieval metrics are used to quantify the performance of the retrieval method. Let us first define the following sets:

- $\mathcal{I}$ is the set of all images in the dataset.
- $\mathcal{I}_k$ is the set of all images in the dataset that belong to the k^{th} class.
- $\mathcal{C}_i = \mathcal{I}_{\text{class}(\mathbf{x}_i)} \backslash \{\mathbf{x}_i\} = \{\mathbf{z} : \mathbf{z} \in \mathcal{I}_{\text{class}(\mathbf{x}_i)}, \mathbf{z} \notin \{\mathbf{x}_i\}\}$ is the set of all images that are of the same class as $\mathbf{x}_i$; excluding the image itself.
- $\mathcal{R}_i^{(j)} = \left\{\mathbf{r}_i^{(1)}, \mathbf{r}_i^{(2)}, \ldots, \mathbf{r}_i^{(j)}\right\}$ is the set top j images in terms of their similarity to $\mathbf{x}_i$. Note that the elements of $\mathcal{R}_i^{(j)}$ are sorted according to their similarity to $\mathbf{x}_i$ such that: $\mathsf{Similarity}\left(\mathbf{x}_i, \mathbf{r}_i^{(k)}\right) \geq \mathsf{Similarity}\left(\mathbf{x}_i, \mathbf{r}_i^{(k+1)}\right)$.
- $\mathcal{R}_i^{(j)} \cap \mathcal{C}_i$ is the set that contains all images that are of the same class as the query image $\mathbf{x}_i$ in the set of retrieved images $\mathcal{R}_i^{(j)}$.

The following information retrieval metrics are used to assess the performance of the similarity measures:

- **Precision at n (P@n)** is the average percentage of correctly retrieved images out of n retrieved images. **Precision at n (P@n)** is defined as follows:

$$\mathbf{P@}n = \frac{1}{|\mathcal{I}|} \sum_{i=1}^{|\mathcal{I}|} \frac{\left|\mathcal{R}_i^{(n)} \cap \mathcal{C}_i\right|}{\left|\mathcal{R}_i^{(n)}\right|}, \tag{A.1}$$

where $|\cdot|$ is the number of elements in the set.

- **Retrieval Accuracy (RA)** is the P@n when n is equal to the number of elements that are of the same class the query images, i.e. $n = |\mathcal{C}_i|$.

$$\text{RA} = \frac{1}{|\mathcal{I}|} \sum_{i=1}^{|\mathcal{I}|} \frac{\left| \mathcal{R}_i^{(|\mathcal{C}_i|)} \cap \mathcal{C}_i \right|}{\left| \mathcal{R}_i^{(|\mathcal{C}_i|)} \right|}. \tag{A.2}$$

- **Average Precision (AP)** for query image $\mathbf{x}_i$ is a measure of precision that takes into account the order of which the correct images are retrieved. It is defined as:

$$\text{AP}_i = \frac{1}{|\mathcal{C}_i|} \sum_{j=1}^{N_s-1} \frac{\left| \mathcal{R}_i^{(j)} \cap \mathcal{C}_i \right|}{\left| \mathcal{R}_i^{(j)} \right|} \times 1_{\{r_i^{(j)} \in \mathcal{C}_i\}}, \tag{A.3}$$

where $1_{\{r_i^{(j)} \in \mathcal{C}_i\}}$ is the indicator function and it is equal to 1 if and only if $r_i^{(j)} \in \mathcal{C}_i$, and 0 otherwise. **Mean Average Precision (MAP)** is the mean value of AP for all images in the dataset.

- **Receiver Operating Characteristics (ROC)** is a plot of the True Positive Rate (TPR) versus False Positive Rate (FPR) for different similarity thresholds. TPR is the percentage of pairs of images that are correctly identified as similar by the similarity measure. FPR is the percentage of pairs of images that are not similar but were identified as similar by the similarity measure. The area under the ROC curve, denoted as **AUC**, is used as a measure of the discriminative power of a similarity measure. The ideal ROC curve would have perfect TPR (TPR=1) for all values of FPR, and in this case, the area under the curve would be maximum AUC = 1.

A.2 Clustering

To evaluate the quality of clusters obtained with a clustering algorithm, we use **Rand index** [136] and **adjusted Rand index** [137].

Rand index is defined as the percentage of pairs of objects that have been either correctly clustered. Formally, for each pair of images, $\mathbf{x}_i$ and $\mathbf{x}_j$, in the dataset, is said to be correctly clustered in the following cases:

- $\mathbf{x}_i$ and $\mathbf{x}_j$ are of the same class and are also put in the same cluster by the clustering algorithm (true positive).
- $\mathbf{x}_i$ and $\mathbf{x}_j$ are of different classes and are in different clusters and are also put in different clusters by the clustering algorithm (true negative).

Before we define Rand index mathematically, let us define the contingency table which is used to compute Rand index. Let $\mathcal{I}$ be a dataset of N samples with a ground-truth clustering of its elements into L clusters, $\{\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_K\}$, where $\mathcal{I}_i$ is the set of elements that belong to the i^{th} cluster. Let $\{\mathcal{J}_1, \mathcal{J}_2, \ldots, \mathcal{J}_L\}$ be the clustering obtained by a clustering algorithm. The contingency table (T), is defined as follows:

	$\mathcal{J}_1$	$\mathcal{J}_2$	$\ldots$	$\mathcal{J}_l$	Sum
$\mathcal{I}_1$	$T_{1,1}$	$T_{1,2}$	$\ldots$	$T_{1,l}$	$a_1 = \sum_i T_{1,i}$
$\mathcal{I}_2$	$T_{2,1}$	$T_{2,2}$	$\ldots$	$T_{2,l}$	$a_2 = \sum_i T_{2,i}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
$\mathcal{I}_K$	$T_{K,1}$	$T_{K,2}$	$\ldots$	$T_{K,L}$	$a_K = \sum_i T_{K,i}$
Sum	$\sum_i b_1 = T_{i,1}$	$b_2 = \sum_i T_{i,2}$	$\ldots$	$b_L = \sum_i T_{i,L}$	$N = \sum_{i,j} T_{i,j}$

where $T_{i,j}$ is the set of elements are are in both $\mathcal{I}_i$ and $\mathcal{J}_j$. Note that $\sum_{i,j} T_{i,j} = |\mathcal{I}|$ which is the total number of elements in the dataset. Then, we define the true positive

(TP), false positive (FP), false negative (FN), and true negative (TN) as follows:

$$\begin{aligned}
\text{TP} &= \sum_{i,j} \binom{T_{i,j}}{2}, \\
\text{FP} &= \sum_{i} \binom{a_i}{2} - \text{TP}, \\
\text{FN} &= \sum_{i} \binom{b_i}{2} - \text{TP}, \\
\text{TN} &= \binom{N}{2} - (\text{TP} + \text{FP} + \text{FN}),
\end{aligned}$$

where $\binom{n}{k} = \frac{n!}{k!(n-k)!}$, and $n!$ is the factorial of n. Rand index is then defined as:

$$\textbf{Rand Index} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FN} + \text{FP}}. \tag{A.4}$$

Rand index gives a value in the range $[0, 1]$ with 1 being a perfect clustering that matches the ground truth. However, Rand index mostly lies in a narrow range of $[0.5 - 1]$. In addition, it does not take a constant value when comparing two random clustering partitions [138]. The adjusted Rand index was then proposed in [137] to address this issue. The adjusted Rand index is defined as follows:

$$\text{Adjusted Rand Index} = \frac{\sum_{i,j} \binom{T_{i,j}}{2} - \sum_i \binom{a_i}{2} \sum_j \binom{b_i}{2} / \binom{N}{2}}{\frac{1}{2}\left[\sum_i \binom{a}{2} \sum_i \binom{b}{i}\right] - \left[\sum_i \binom{a_i}{2} \sum_i \binom{b_i}{2}\right] / \binom{N}{2}}. \tag{A.5}$$

The adjusted Rand index is no longer bounded by zero. However, it gives a value of zero when the clustering is as good as a random one.

A.3 Segmentation

To evaluate the quality of a labeling of an image or volume with respect to the ground-truth labeling of the image, we use semantic segmentation metrics. Before we define these metrics, we will define the following sets and constants:

- $\mathcal{G}_i$ is the set of all pixels that belong to the i^{th} in the ground-truth labeling.
- $\mathcal{H}_i$ is the set of all pixels that labeled as i by a classification or a segmentation algorithm.
- N_c is the number of classes, or the number of possible labels for a pixel.
- $|\mathcal{A}|$ is the number of items in a set $\mathcal{A}$.

These sets are used to determine the quality of the labeling obtained using a classification or a segmentation algorithm with the following metrics:

- Pixel Accuracy (**PA**) is defined as the percentage of pixels that are labeled correctly.

$$\mathbf{PA} = \frac{\sum_i |\mathcal{G}_i \cap \mathcal{H}_i|}{\sum_i |\mathcal{G}_i|}. \tag{A.6}$$

- Class Accuracy for the i^{th} class ($\mathbf{CA}_i$) is the percentage of pixels that belong to the i^{th} and have been correctly labeled as i,

$$\mathbf{CA}_i = \frac{|\mathcal{G}_i \cap \mathcal{H}_i|}{|\mathcal{G}_i|}. \tag{A.7}$$

  Then, the average of **CA** over all classes gives Mean Class Accuracy (MCA),

$$\mathbf{MCA} = \frac{1}{N_c} \frac{|\mathcal{G}_i \cap \mathcal{H}_i|}{|\mathcal{G}_i|}. \tag{A.8}$$

- Intersection over Union ($\mathbf{IoU}_i$) measures the overlap between $\mathcal{G}_i$ and $\mathcal{H}_i$. It is defined as the ratio between the size of the intersection set of $\mathcal{G}_i$ and $\mathcal{H}_i$ to the size of their union set,

$$\mathbf{IoU}_i = \frac{|\mathcal{G}_i \cap \mathcal{H}_i|}{|\mathcal{G}_i \cup \mathcal{H}_i|}. \tag{A.9}$$

Averaging $\mathbf{IoU}_i$ over all classes give Mean Intersection over Union (**MIoU**),

$$\mathbf{MIoU}_i = \frac{1}{N_c} \sum_i \frac{|\mathcal{G}_i \cap \mathcal{H}_i|}{|\mathcal{G}_i \cup \mathcal{H}_i|}. \tag{A.10}$$

Alternatively, we can use Frequency-Weighted Mean Intersection over Union (**FW-MIoU**) which gives more weight to the more frequent classes,

$$\mathbf{FW\text{-}MIoU}_i = \frac{1}{|\mathcal{G}_i|} \sum_i |\mathcal{G}_i| \cdot \frac{|\mathcal{G}_i \cap \mathcal{H}_i|}{|\mathcal{G}_i \cup \mathcal{H}_i|}. \tag{A.11}$$

A.4 Goodness-of-fit

To measure the goodness-of-fit of one 1-d discrete signal to another, we use Pearson Correlation Coefficient (**PCC**), and the coefficient of determination. Let $\mathbf{y}$ be the reference signal, and $\hat{\mathbf{y}}$ be an estimation of $\mathbf{y}$. Pearson correlation coefficient is defined as follows:

$$\mathbf{PCC} = \frac{\sum_i (\mathbf{y}[i] - \mu_{\mathbf{y}}) (\hat{\mathbf{y}}[y] - \mu_{\hat{\mathbf{y}}})}{\sigma_{\mathbf{y}} \sigma_{\hat{\mathbf{y}}}}, \tag{A.12}$$

where $\mu_{\mathbf{y}}$ and $\mu_{\hat{\mathbf{y}}}$ are the average values of $\mathbf{y}$ and $\hat{\mathbf{y}}$, respectively, and $\sigma_{\mathbf{y}}$ and $\sigma_{\hat{\mathbf{y}}}$, are their standard deviation values, respectively. **PCC** is in the range $[-1, 1]$, with values of -1 and 1 indicating perfect positive and negative linear correlation, respectively. A value of 0 indicates no correlation.

The coefficient of determination ($\mathbf{r}^2$) is defined as follows:

$$\mathbf{r}^2 = 1 - \frac{\sum_i \left(\mathbf{y}[i] - \hat{\mathbf{y}}[i]\right)^2}{\sum_i \left(\mathbf{y}[i] - \mu_{\mathbf{y}}\right)^2}. \tag{A.13}$$

The range of coefficient of determination is $[-\infty, 1]$, with a value of 1 indicating a perfect match between $\mathbf{y}$ and $\hat{\mathbf{y}}$.

REFERENCES

[1] K. Boman, *Big data growth continues in seismic surveys*, 2015.

[2] Ö. Yilmaz, *Seismic data analysis: Processing, inversion, and interpretation of seismic data*. Society of exploration geophysicists, 2001.

[3] D. Morton-Thompson, A. M. Woods, *et al.*, *Development geology reference manual: AAPG methods in exploration series, no. 10*, 10. AAPG, 1993.

[4] U. EIA, "Trends in us oil and natural gas upstream costs," *US Energy Information Administration*, 2016.

[5] B. dGB Earth Sciences, *The Netherlands Offshore, The North Sea, F3 Block - Complete*, 1987.

[6] L. Zhu, E. Liu, and L. Zhu, *Seismic simulation, survey, and imaging (sssi)*, `https://github.com/zhulingchen/SSSI`, 2013.

[7] K. Barry, D. Cavers, and C. Kneale, "Recommended standards for digital tape formats," *Geophysics*, vol. 40, no. 2, pp. 344–352, 1975.

[8] M. T. Taner, "Seismic attributes," *CSEG recorder*, vol. 26, no. 7, pp. 49–56, 2001.

[9] R. M. Haralick, K. Shanmugam, *et al.*, "Textural features for image classification," *IEEE Transactions on systems, man, and cybernetics*, no. 6, pp. 610–621, 1973.

[10] S. Chopra and V. Alexeev, "Applications of texture attribute analysis to 3d seismic data," *The Leading Edge*, vol. 25, no. 8, pp. 934–940, 2006.

[11] D. Gao, "Latest developments in seismic texture analysis for subsurface structure, facies, and reservoir characterization: A review," *Geophysics*, vol. 76, no. 2, W1–W13, 2011.

[12] C Eichkitz, J Amtmann, and M. Schreilechner, "Application of glcm-based seismic attributes for anisotropy detection," in *76th EAGE Conference and Exhibition 2014*, 2014.

[13] C. G. Eichkitz, J. Davies, J. Amtmann, M. G. Schreilechner, and P. de Groot, "Grey level co-occurrence matrix and its application to seismic data," *first break*, vol. 33, no. 3, pp. 71–77, 2015.

[14] T. Coléou, M. Poupon, and K. Azbel, "Unsupervised seismic facies classification: A review and comparison of techniques and implementation," *The Leading Edge*, vol. 22, no. 10, pp. 942–953, 2003.

[15] Y. Alaudah, S. Gao, and G. AlRegib, "Learning to label seismic structures with deconvolution networks and weak labels," in *SEG Technical Program Expanded Abstracts 2018*, Society of Exploration Geophysicists, 2018, pp. 2121–2125.

[16] G. AlRegib, M. Deriche, Z. Long, H. Di, Z. Wang, Y. Alaudah, M. A. Shafiq, and M. Alfarraj, "Subsurface structure analysis using computational interpretation and learning: A visual signal processing perspective," *IEEE Signal Processing Magazine*, vol. 35, no. 2, pp. 82–98, 2018.

[17] Z. Wang, H. Di, M. A. Shafiq, Y. Alaudah, and G. AlRegib, "Successful leveraging of image processing and machine learning in seismic structural interpretation: A review," *The Leading Edge*, vol. 37, no. 6, pp. 451–461, 2018.

[18] S. Chopra and J. P. Castagna, *AVO*. Society of Exploration Geophysicists, 2014.

[19] V. Das and T. Mukerji, "Petrophysical properties prediction from pre-stack seismic data using convolutional neural networks," in *SEG Technical Program Expanded Abstracts 2019*, Society of Exploration Geophysicists, 2019, pp. 2328–2332.

[20] V. Das, A. Pollack, U. Wollner, and T. Mukerji, "Convolutional neural network for seismic impedance inversion," in *SEG Technical Program Expanded Abstracts 2018*, Society of Exploration Geophysicists, 2018, pp. 2071–2075.

[21] S. Chaki, A. Routray, and W. K. Mohanty, "A novel preprocessing scheme to improve the prediction of sand fraction from seismic attributes using neural networks," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 8, no. 4, pp. 1808–1820, 2015.

[22] M. Araya-Polo, J. Jennings, A. Adler, and T. Dahlke, "Deep-learning tomography," *The Leading Edge*, vol. 37, no. 1, pp. 58–66, 2018.

[23] G. S. Martin, R. Wiley, and K. J. Marfurt, "Marmousi2: An elastic upgrade for marmousi," *The Leading Edge*, vol. 25, no. 2, pp. 156–166, 2006.

[24] P. Aigrain, H. Zhang, and D. Petkovic, "Content-based representation and retrieval of visual media: A state-of-the-art review," *Multimedia tools and applications*, vol. 3, no. 3, pp. 179–202, 1996.

[25] C. C. Venters and M. Cooper, "A review of content-based image retrieval systems," *JISC Technology Applications Programme, http://www. jtap. ac. uk/reports/htm/jtap-054. html*, 2000.

[26] A. W. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 12, pp. 1349–1380, 2000.

[27] H. Müller, N. Michoux, D. Bandon, and A. Geissbuhler, "A review of content-based image retrieval systems in medical applications—clinical benefits and future directions," *International journal of medical informatics*, vol. 73, no. 1, pp. 1–23, 2004.

[28] T. Yao, Y. Pan, Y. Li, Z. Qiu, and T. Mei, "Boosting image captioning with attributes," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 4894–4902.

[29] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo, "Image captioning with semantic attention," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4651–4659.

[30] C.-Y. Ma, A. Kadav, I. Melvin, Z. Kira, G. AlRegib, and H. Peter Graf, "Attend and interact: Higher-order object interactions for video understanding," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6790–6800.

[31] G Castellano, L Bonilha, L. Li, and F Cendes, "Texture analysis of medical images," *Clinical radiology*, vol. 59, no. 12, pp. 1061–1069, 2004.

[32] B Singh and B. Mazumdar, "Content retrieval from xray images using color & texture features," *Methodology*, vol. 1, p. 6, 2010.

[33] L. Nanni, A. Lumini, and S. Brahnam, "Local binary patterns variants as texture descriptors for medical image analysis," *Artificial intelligence in medicine*, vol. 49, no. 2, pp. 117–125, 2010.

[34] I. Pitas and C. Kotropoulos, "A texture-based approach to the segmentation of seismic images," *Pattern Recognition*, vol. 25, no. 9, pp. 929–945, 1992.

[35] K. Röster and M. Spann, "A system for seismic data processing," in *EUSIPCO 1998*, 1998.

[36] M. A. Shafiq, Z. Wang, G. AlRegib, A. Amin, and M. Deriche, "A texture-based interpretation workflow with application to delineating salt domes," *Interpretation*, vol. 5, no. 3, SJ1–SJ19, 2017.

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

[38] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 7, pp. 971–987, 2002.

[39] S. Liao, M. W. Law, and A. C. Chung, "Dominant local binary patterns for texture classification," *IEEE transactions on image processing*, vol. 18, no. 5, pp. 1107–1118, 2009.

[40] B. Zhang, Y. Gao, S. Zhao, and J. Liu, "Local derivative pattern versus local binary pattern: Face recognition with high-order local pattern descriptor," *IEEE transactions on image processing*, vol. 19, no. 2, pp. 533–544, 2010.

[41] Y. HU, Z. Long, and G. AlRegib, "Completed local derivative pattern for rotation invariant texture classification," in *Proceedings of the International Conference on Image Processing (ICIP)*, 2016.

[42] M. P. Sampat, Z. Wang, S. Gupta, A. C. Bovik, and M. K. Markey, "Complex wavelet structural similarity: A new image similarity index," *IEEE transactions on image processing*, vol. 18, no. 11, pp. 2385–2401, 2009.

[43] J. Zujovic, T. N. Pappas, and D. L. Neuhoff, "Structural texture similarity metrics for image analysis and retrieval," *IEEE Transactions on Image Processing*, vol. 22, no. 7, pp. 2545–2558, 2013.

[44] D. Zhang, M. M. Islam, G. Lu, and I. J. Sumana, "Rotation invariant curvelet features for region based image retrieval," *International journal of computer vision*, vol. 98, no. 2, pp. 187–201, 2012.

[45] S. Selvan and S. Ramakrishnan, "Svd-based modeling for image texture classification using wavelet transformation," *Image Processing, IEEE Transactions on*, vol. 16, no. 11, pp. 2688–2696, 2007.

[46] S Arivazhagan, L Ganesan, and T. S. Kumar, "Texture classification using curvelet statistical and co-occurrence features," in *18th International Conference on Pattern Recognition (ICPR'06)*, IEEE, vol. 2, 2006, pp. 938–941.

[47] M. N. Do and M. Vetterli, "Texture similarity measurement using kullback-leibler distance on wavelet subbands," in *Proceedings 2000 International Conference on Image Processing (Cat. No. 00CH37101)*, IEEE, vol. 3, 2000, pp. 730–733.

[48] T. Hegazy* and G. AlRegib, "Texture attributes for detecting salt bodies in seismic data," in *SEG Technical Program Expanded Abstracts 2014*, Society of Exploration Geophysicists, 2014, pp. 1455–1459.

[49] H. Di, M. A. Shafiq, and G. AlRegib, "Seismic-fault detection based on multiattribute support vector machine analysis," in *SEG Technical Program Expanded Abstracts 2017*, Society of Exploration Geophysicists, 2017, pp. 2039–2044.

[50] H. Al-Marzouqi and G. Al Regib, "Texture similarity using periodically extended and adaptive curvelets," in *2014 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, IEEE, 2014, pp. 951–955.

[51] Z. Long, Z. Wang, and G. AlRegib, "Seisim: Structural similarity evaluation for seismic data retrieval," 2014.

[52] Y. Alaudah and G. AlRegib, "A curvelet-based distance measure for seismic images," in *Image Processing (ICIP), 2015 IEEE International Conference on*, IEEE, 2015, pp. 4200–4204.

[53] Z. Long, Y. Alaudah, M. A. Qureshi, Y. Hu, Z. Wang, M. Alfarraj, G. AlRegib, A. Amin, M. Deriche, S. Al-Dharrab, *et al.*, "A comparative study of texture attributes for characterizing subsurface structures in seismic volumes," *Interpretation*, vol. 6, no. 4, T1055–T1066, 2018.

[54] A. B. Mattos, R. S. Ferreira, R. M. D. G. e Silva, M. Riva, and E. V. Brazil, "Assessing texture descriptors for seismic image retrieval," in *Graphics, Patterns and Images (SIBGRAPI), 2017 30th SIBGRAPI Conference on*, IEEE, 2017, pp. 292–299.

[55] M. Shafiq, Y Alaudah, and G AlRegib, "Salt dome delineation using edge- and texture-based attributes," in *79th EAGE Conference and Exhibition 2017*, 2017.

[56] E. H. Adelson, C. H. Anderson, J. R. Bergen, P. J. Burt, and J. M. Ogden, "Pyramid methods in image processing," *RCA engineer*, vol. 29, no. 6, pp. 33–41, 1984.

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

[58] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 20, no. 11, pp. 1254–1259, 1998.

[59] I. Daubechies, *Ten lectures on wavelets*. SIAM, 1992.

[60] J. G. Daugman, "Complete discrete 2-d gabor transforms by neural networks for image analysis and compression," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 36, no. 7, pp. 1169–1179, 1988.

[61] R. Mehrotra, K. R. Namuduri, and N. Ranganathan, "Gabor filter-based edge detection," *Pattern recognition*, vol. 25, no. 12, pp. 1479–1494, 1992.

[62] A. K. Jain and F. Farrokhnia, "Unsupervised texture segmentation using gabor filters," *Pattern recognition*, vol. 24, no. 12, pp. 1167–1186, 1991.

[63] T. Randen and L. Sønneland, "Atlas of 3d seismic attributes," in *Mathematical Methods and Modelling in Hydrocarbon Exploration and Production*, Springer, 2005, pp. 23–46.

[64] E. Candes, L. Demanet, D. Donoho, and L. Ying, "Fast discrete curvelet transforms," *Multiscale Modeling & Simulation*, vol. 5, no. 3, pp. 861–899, 2006.

[65] E. J. Candès and D. L. Donoho, "Curvelets - A surprisingly effective nonadaptive representation for objects with edges," *Curves and Surfaces*, vol. C, no. 2, pp. 1–10, 2000.

[66] O. Roy and M. Vetterli, "The effective rank: A measure of effective dimensionality," in *European signal processing conference (EUSIPCO)*, 2007, pp. 606–610.

[67] (). Curet: Columbia-utrecht reflectance and texture database.

[68] F. Halley, "Perceptually relevant browsing environments for large texture databases," PhD thesis, Heriot-Watt University, 2012.

[69] (). Forrest texture library.

[70] D. Zhang, M. M. Islam, G. Lu, and I. J. Sumana, "Rotation invariant curvelet features for region based image retrieval," *International journal of computer vision*, vol. 98, no. 2, pp. 187–201, 2012.

[71] M. Alfarraj, Y. Alaudah, and G. AlRegib, "Content-adaptive non-parametric texture similarity measure," in *2016 IEEE 18th International Workshop on Multimedia Signal Processing (MMSP)*, IEEE, 2016, pp. 1–6.

[72] CeGP, "LANDMASS: large north-sea dataset of migrated aggregated seismic structures," 2015.

[73] J. B. Kruskal, "Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis," *Psychometrika*, vol. 29, no. 1, pp. 1–27, 1964.

[74] Y. Alaudah and G. AlRegib, "Weakly-supervised labeling of seismic volumes using reference exemplars," in *Image Processing (ICIP), 2016 IEEE International Conference on*, IEEE, 2016, pp. 4373–4377.

[75] M. Alfarraj, Y. Alaudah, Z. Long, and G. AlRegib, "Multiresolution analysis and learning for computational seismic interpretation," *The Leading Edge*, vol. 37, no. 6, pp. 443–450, 2018.

[76] A. Duijndam, "Bayesian estimation in seismic inversion. part i: Principles," *Geophysical Prospecting*, vol. 36, no. 8, pp. 878–898, 1988.

[77] P. M. Doyen, "Porosity from seismic data: A geostatistical approach," *Geophysics*, vol. 53, no. 10, pp. 1263–1275, 1988.

[78] A. Duijndam, "Bayesian estimation in seismic inversion. part ii: Uncertainty analysis," *Geophysical Prospecting*, vol. 36, no. 8, pp. 899–918, 1988.

[79] T. J. Ulrych, M. D. Sacchi, and A. Woodbury, "A bayes tour of inversion: A tutorial," *Geophysics*, vol. 66, no. 1, pp. 55–69, 2001.

[80] A. Buland and H. Omre, "Bayesian linearized avo inversion," *Geophysics*, vol. 68, no. 1, pp. 185–198, 2003.

[81] A. Gholami, "Nonlinear multichannel impedance inversion by total-variation regularization," *Geophysics*, vol. 80, no. 5, R217–R224, 2015.

[82] A. Tarantola, *Inverse problem theory and methods for model parameter estimation*. siam, 2005, vol. 89.

[83] A. Lucas, M. Iliadis, R. Molina, and A. K. Katsaggelos, "Using deep neural networks for inverse problems in imaging: Beyond analytical methods," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 20–36, 2018.

[84] A. Richardson, "Seismic full-waveform inversion using deep learning tools and techniques," *arXiv preprint arXiv:1801.07232*, 2018.

[85] S. Chaki, A. Routray, and W. K. Mohanty, "Well-log and seismic data integration for reservoir characterization: A signal processing and machine-learning perspective," *IEEE Signal Processing Magazine*, vol. 35, no. 2, pp. 72–81, 2018.

[86] A. Al-Anazi and I. Gates, "Support vector regression to predict porosity and permeability: Effect of sample size," *Computers & geosciences*, vol. 39, pp. 64–76, 2012.

[87] G. Röth and A. Tarantola, "Neural networks and inversion of seismic data," *Journal of Geophysical Research: Solid Earth*, vol. 99, no. B4, pp. 6753–6768, 1994.

[88] A. Gholami and H. R. Ansari, "Estimation of porosity from seismic attributes using a committee model with bat-inspired optimization algorithm," *Journal of Petroleum Science and Engineering*, vol. 152, pp. 238–249, 2017.

[89] S. Yuan and S. Wang, "Spectral sparse Bayesian learning reflectivity inversion," *Geophysical Prospecting*, vol. 61, no. 4, pp. 735–746, 2013.

[90] L. Mosser, W. Kimman, J. Dramsch, S. Purves, A De la Fuente Briceño, and G. Ganssle, "Rapid seismic domain transfer: Seismic velocity inversion and modeling using deep generative neural networks," in *80th EAGE Conference and Exhibition 2018*, 2018.

[91] A. Stolcke and E. Shriberg, "Statistical language modeling for speech disfluencies," in *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, IEEE, vol. 1, 1996, pp. 405–408.

[92] X. Liu and T. Cheng, "Video-based face recognition using adaptive hidden markov models," in *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, IEEE, vol. 1, 2003, pp. I–I.

[93] W. C. Krumbein and M. F. Dacey, "Markov chains and embedded markov chains in geology," *Journal of the International Association for Mathematical Geology*, vol. 1, no. 1, pp. 79–96, 1969.

[94] L. E. Baum and T. Petrie, "Statistical inference for probabilistic functions of finite state markov chains," *The annals of mathematical statistics*, vol. 37, no. 6, pp. 1554–1563, 1966.

[95] F. Jelinek, "Markov source modeling of text generation," in *The Impact of Processing Techniques on Communications*, Springer, 1985, pp. 569–591.

[96] D. Bruckner, B. Sallans, and G. Russ, "Hidden markov models for traffic observation," in *2007 5th IEEE International Conference on Industrial Informatics*, IEEE, vol. 2, 2007, pp. 989–994.

[97] S. Vogel, H. Ney, and C. Tillmann, "Hmm-based word alignment in statistical translation," in *Proceedings of the 16th conference on Computational linguistics-Volume 2*, Association for Computational Linguistics, 1996, pp. 836–841.

[98] C Lomnitz, "Statistical prediction of earthquakes," *Reviews of Geophysics*, vol. 4, no. 3, pp. 377–393, 1966.

[99] R. Godfrey, F. Muir, and F. Rocca, "Modeling seismic impedance with markov chains," *Geophysics*, vol. 45, no. 9, pp. 1351–1372, 1980.

[100] A. Amin, M. Deriche, and B. Liu, "A novel approach for salt dome detection in seismic surveys using a hidden markov model," in *SEG Technical Program Expanded Abstracts 2016*, Society of Exploration Geophysicists, 2016, pp. 1688–1692.

[101] D. E. Rumelhart, G. E. Hinton, R. J. Williams, *et al.*, "Learning representations by back-propagating errors," *Cognitive modeling*, vol. 5, no. 3, p. 1, 1988.

[102] T. Mikolov, M. Karafiát, L. Burget, J. Černockỳ, and S. Khudanpur, "Recurrent neural network based language model," in *Eleventh Annual Conference of the International Speech Communication Association*, 2010.

[103] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Acoustics, speech and signal processing (icassp), 2013 ieee international conference on*, IEEE, 2013, pp. 6645–6649.

[104] C.-Y. Ma, M.-H. Chen, Z. Kira, and G. AlRegib, "Ts-lstm and temporal-inception: Exploiting spatiotemporal dynamics for activity recognition," *arXiv preprint arXiv:1703.10667*, 2017.

[105] J. Wiszniowski, B. M. Plesiewicz, and J. Trojanowski, "Application of real time recurrent neural network for detection of small natural earthquakes in poland," *Acta Geophysica*, vol. 62, no. 3, pp. 469–485, 2014.

[106] R. Biswas, A. Vassiliou, R. Stromberg, and M. K. Sen, "Stacking velocity estimation using recurrent neural network," in *SEG Technical Program Expanded Abstracts 2018*, Society of Exploration Geophysicists, 2018, pp. 2241–2245.

[107] P. J. Werbos, "Backpropagation through time: What it does and how to do it," *Proceedings of the IEEE*, vol. 78, no. 10, pp. 1550–1560, 1990.

[108] S. Hochreiter and J. Schmidhuber, "Lstm can solve hard long time lag problems," in *Advances in neural information processing systems*, 1997, pp. 473–479.

[109] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," *arXiv preprint arXiv:1409.1259*, 2014.

[110] M. Alfarraj and G. AlRegib, "Petrophysical property estimation from seismic data using recurrent neural networks," in *SEG Technical Program Expanded Abstracts 2018*, Society of Exploration Geophysicists, 2018, pp. 2141–2146.

[111] R. Versteeg, "The marmousi experience: Velocity model determination on a synthetic complex data set," *The Leading Edge*, vol. 13, no. 9, pp. 927–936, 1994.

[112] K Zoeppritz, "Über reflexion und durchgang seismischer wellen durch unstetigkeitsflächen: Nachrichten von der gesellschaft der wissenschaften zu göttingen," *Mathematisch-Physikalische Klasse*, vol. 1, pp. 66–84, 1919.

[113] P. G. Richards and K. Aki, *Quantitative seismology: theory and methods*. Freeman, 1980.

[114] R. Bortfeld, "Approximations to the reflection and transmission coefficients of plane longitudinal and transverse waves," *Geophysical Prospecting*, vol. 9, no. 4, pp. 485–502, 1961.

[115] R. Shuey, "A simplification of the zoeppritz equations," *Geophysics*, vol. 50, no. 4, pp. 609–614, 1985.

[116] P. Connolly, "Elastic impedance," *The Leading Edge*, vol. 18, no. 4, pp. 438–452, 1999.

[117] D. N. Whitcombe, "Elastic impedance normalization," *Geophysics*, vol. 67, no. 1, pp. 60–62, 2002.

[118] D. A. Cooke and W. A. Schneider, "Generalized linear inversion of reflection seismic data," *Geophysics*, vol. 48, no. 6, pp. 665–676, 1983.

[119] A. N. Tikhonov, "On the solution of ill-posed problems and the method of regularization," in *Doklady Akademii Nauk*, Russian Academy of Sciences, vol. 151, 1963, pp. 501–504.

[120] F. Santosa and W. W. Symes, "Linear inversion of band-limited reflection seismograms," *SIAM Journal on Scientific and Statistical Computing*, vol. 7, no. 4, pp. 1307–1330, 1986.

[121] J. Adler and O. Öktem, "Solving ill-posed inverse problems using iterative deep neural networks," *Inverse Problems*, vol. 33, no. 12, p. 124 007, 2017.

[122] R. Biswas, M. K. Sen, V. Das, and T. Mukerji, "Pre-stack and post-stack inversion using a physics-guided convolutional neural network," *Interpretation*, vol. 7, no. 3, pp. 1–76, 2019.

[123] M. Alfarraj and G. AlRegib, "Semi-supervised learning for acoustic impedance inversion," in *SEG Technical Program Expanded Abstracts 2019*. 2019, pp. 2298–2302. eprint: `https://library.seg.org/doi/pdf/10.1190/segam2019-3215902.1`.

[124] ——, "Semi-supervised sequence modeling for elastic impedance inversion," *Interpretation*, vol. 7, no. 3, pp. 1–65, 2019.

[125] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," *arXiv preprint arXiv:1511.07122*, 2015.

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

[127] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1520–1528.

[128] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," in *NIPS-W*, 2017.

[129] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[130] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural networks*, vol. 61, pp. 85–117, 2015.

[131] T. Zhou, P. Krahenbuhl, M. Aubry, Q. Huang, and A. A. Efros, "Learning dense correspondence via 3d-guided cycle consistency," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 117–126.

[132] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.

[133] J. Hoffman, E. Tzeng, T. Park, J.-Y. Zhu, P. Isola, K. Saenko, A. A. Efros, and T. Darrell, "Cycada: Cycle-consistent adversarial domain adaptation," *arXiv preprint arXiv:1711.03213*, 2017.

[134] Z. Zhang, L. Yang, and Y. Zheng, "Translating and segmenting multimodal medical volumes with cycle-and shape-consistency generative adversarial network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 9242–9251.

[135] J. Song, K. Pang, Y.-Z. Song, T. Xiang, and T. M. Hospedales, "Learning to sketch with shortcut cycle consistency," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 801–810.

[136] W. M. Rand, "Objective criteria for the evaluation of clustering methods," *Journal of the American Statistical association*, vol. 66, no. 336, pp. 846–850, 1971.

[137] L. Hubert and P. Arabie, "Comparing partitions," *Journal of classification*, vol. 2, no. 1, pp. 193–218, 1985.

[138] K. Y. Yeung and W. L. Ruzzo, "Details of the adjusted rand index and clustering algorithms, supplement to the paper an empirical study on principal component analysis for clustering gene expression data," *Bioinformatics*, vol. 17, no. 9, pp. 763–774, 2001.

VITA

Motaz Alfarraj was born in Buraidah, Saudi Arabia in March 1991. He received his B.Sc. degree with first honors in Electrical Engineering in 2013 from King Fahd University of Petroleum and Minerals (KFUPM), Dhahran, Saudi Arabia. He received his M.S. and Ph.D. degrees in Electrical and Computer Engineering with a minor in Math from Georgia Institute of Technology, Atlanta, GA in 2015 and 2019, respectively. His research interests include machine learning, image processing, and computational seismic interpretation. During his studies at Georgia Institute of Technology, Motaz worked as a research assistant in Omni Lab for Intelligent Visual Engineering and Science (OLIVES) and Center for Energy and Geo Processing (CeGP) on various vision-related problems such as machine learning for visual data, texture representations, computational seismic interpretation. He also worked as a teaching assistant for Digital Image Processing class (ECE 6258) in Fall 2015, Fall 2016, and Fall 2017 for which he received the Roger P. Webb Outstanding Teaching Assistant Award. Motaz was also a member of the Students Activity Council (SAC) at the Center for Signal and Information Processing (CSIP) at Georgia Institute of Technology between Fall 2017 and Fall 2019. He received the best service award from CSIP in Spring 2019. Motaz also worked as a Geoscience inter in the Advanced Analytics and Emerging Technology (AAET) group at Anadarko Petroleum Corporation in Summer 2018. Motaz is a member of the IEEE Signal Processing Society (SPS), Society of Exploration Geophysicists (SEG), and Society of Petroleum Engineers (SPE).