UiO : University of Oslo

Jing Sun

Deep learning-based seismic data processing for attenuation of interference noise and deblending in the shot domain

Thesis submitted for the degree of Philosophiae Doctor

Department of Geosciences

Faculty of Mathematics and Natural Sciences

2021

Series of dissertations submitted to the
Faculty of Mathematics and Natural Sciences, University of Oslo
No. 2480

ISSN 1501-7710

Cover: Hanne Baadsgaard Utigard.
Print production: Reprosentralen, University of Oslo.

Preface

This thesis is submitted to the Faculty of Mathematics and Natural Sciences at the University of Oslo in partial fulfillment of the requirements for the degree of Philosophiae Doctor (Ph.D.). The research presented here was conducted at CGG Norway AS and the University of Oslo, under the main supervision of Dr. Vetle Vinje and co-supervision of Prof. Valerie Maupin. This work is part of the Industrial Ph.D. scheme funded by the Norwegian Research council through grant 314179, initiated by CGG Norway AS and the University of Oslo.

The thesis is a collection of four papers, dealing with methods to process seismic data using deep learning, specifically to separate coherent noise from the wanted signal directly in the shot domain, without breaking the coherency of the noise. I am first author of all papers. The first chapter of the thesis covers the motivation, objectives and scope. The second chapter gives background information to seismic, including acquisition and characteristics of data and noise. The third chapter scientifically introduces deep learning, followed by a short overview of this technology in the field of seismic signal separation. The fourth chapter gives a summary of each included paper. The four included papers are divided into two categories based on the processing tasks and presented in chronological order in each category. The first processing task is to attenuate seismic interference noise, while the second task is seismic deblending. At the end, a set of conclusions, discussion and outlook of future work are given.

Acknowledgements

Pursuing this Ph.D. is not an easy thing and I would not be able to achieve the work of this thesis without the support of my supervisors, colleagues, families and friends.

First, I would like to thank my main supervisor Dr. Vetle Vinje and my co-supervisor Prof. Valerie Maupin for their good guidance of my study and for always finding time to have interesting and open discussions. I would also like to thank my two supervisors at an early stage, Prof. Leiv Jacob Gelius and Dr. Thomas Elboth, for their kind help and contributions.

Further, I appreciate CGG for providing the equipment, software, computational resources, data, and such a nice office. I also would like to express my gratitude to everybody here working together, especially my lovely peers in the team, Volodya Hlebnikov, Saskia Tschache and Thomas de Jonge, and our smartest senior researcher, Dr. Peng Zhao.

At the same time, I would like to thank Dr. Song Hou at CGG UK for supervising and working closely with me on deep learning. I thank Mr. Gordon Poole and Ms. Ewa Kaszycka at CGG UK for always digitally meeting us Ph.D. students in Norway and give insightful comments.

I also want to thank my peers at the University of Oslo, Sigmund Slang and Thomas Larsen Greiner, and my previous colleague at CGG Norway Mr. Steven McDonald, the chapter of deep learning in my research would not be opened without meeting you in Oslo.

Finally, I want to thank my parents and friends for their trust, support and company across distances and time zone differences.

:Jing Sun

Oslo, October 2021

List of Papers

Paper I

Attenuation of marine seismic interference noise employing a customized U-Net,

Jing Sun, Sigmund Slang, Thomas Elboth, Thomas Larsen Greiner, Steven McDonald and Leiv Jacob Gelius,

Geophysical Prospecting, 2020, Vol. 68, no. 3, 845-871,

DOI: 10.1111/1365-2478.12893.

Paper II

An exploratory study toward demystifying deep learning in seismic signal separation,

Jing Sun and Song Hou,

In review.

Paper III

DNN-based workflow for attenuating seismic interference noise and its application to marine towed streamer data from the North Sea,

Jing Sun, Song Hou and Alaa Triki,

In review.

Paper IV

Deep learning-based shot-domain seismic deblending,

Jing Sun, Song Hou, Vetle Vinje, Gordon Poole and Leiv Jacob Gelius,

Geophysics, Accepted.

Related Publications

Paper I

A convolutional neural network approach to deblending seismic data,

Jing Sun, Sigmund Slang, Thomas Elboth, Thomas Larsen Greiner, Steven McDonald and Leiv Jacob Gelius,

Geophysics, 2020, vol. 85, no. 4, WA13-WA26,

DOI: 10.1190/geo2019-0173.1.

Paper II

Using convolutional neural networks for denoising and deblending of marine seismic data,

Sigmund Slang, Jing Sun, Thomas Elboth, Steven McDonald and Leiv Jacob Gelius,

81st EAGE Annual International Conference and Exhibition, 2019, Extended Abstract,

DOI: 10.3997/2214-4609.201900844.

Abbreviations

AdaGrad	Adaptive Gradient Descent
Adam	Adaptive Moment Estimation
AI	Artificial Intelligence
ANN	Artificial Neural Network
CNN	Convolutional Neural Network
CPU	Central Processing Unit
DL	Deep Learning
DnCNN	Denoising Convolutional Neural Network
DNN	Deep Neural Network
GD	Gradient Descent
GPU	Graphics Processing Unit
MAE	Mean Absolute Error
ML	Machine Learning
MSE	Mean Squared Error
OBC/OBN	Ocean Bottom Cable/Node
ReLU	Rectified Linear Unit
RMSprop	Root Mean Square Propagation
SGD	Stochastic Gradient Descent
SI	Seismic Interference
SNR	Signal-to-Noise Ratio
TWT	Two-Way Traveltime

Contents

Part I: Overview

Chapter 1

Introduction

1.1 Motivation

Throughout the energy industry, seismic exploration has been used extensively for decades, being essential in finding oil and natural gas. It consists of three main steps: (i) conducting seismic acquisition on-shore or off-shore by sending acoustic energy through subsurface layers, (ii) processing the recorded seismic data for imaging, and (iii) inferring geology from the above (the so-called seismic interpretation). Seismic data contain information on the subsurface structures through the changes in acoustic impedance. Its raw form from a field survey exhibits a mixture of various reflections and is always rather noisy. Separating the noise from the desired reflection signal is an important task of seismic data processing, being critical to the success of the following imaging and interpretation. To this intent, the seismic industry has invested a lot of manpower, material and financial resources in the past decades to develop corresponding scientific theories and methodologies, and has achieved a series of well-performed algorithms.

However, seismic data processing via algorithms for noise removal used in real industrial processing projects, is normally computationally intensive and manually time-consuming due to the extensive parameter tuning required. In the implementation of a conventional physics-based algorithm, to efficiently separate the noise whilst preserving the fidelity of the wanted signal, a workflow of multiple procedures is always employed. Moreover, even with a professionally designed workflow, the level of processing quality can still vary considerably from one project to another, being dependent on various factors of the survey. As a consequence, different processing approaches may be collected to build a tailored workflow for each project. In addition, while the processing algorithms are getting more complex and costly, survey designs have been made denser and larger. The increasing amount of seismic records in combination with the pressure to deliver processing results within a limited time frame represents a major challenge to the seismic processing teams. In this situation, the development of significantly cheaper and more time-efficient processing methods is a real demand.

Deep Learning (DL), the most popular subfield of Artificial Intelligence (AI), represents the data-driven technique that enables Deep Neural Networks (DNNs) to identify patterns in a wide range of data sets without the need to write specific code tailored for each problem. As such, this approach meets the demand of the seismic industry to process massive data. In recent years, the availability of powerful hardware (CPU and GPU) at relatively low cost, as well as easy access to open-source software (e.g. TensorFlow (Abadi et al., 2016), Keras (Chollet, 2015) and PyTorch (Paszke et al., 2017)), has enabled DNNs to be applied in a variety of scientific and engineering fields. Successful applications and breakthroughs of this technology in computer vision (Voulodimos et al., 2018) and natural image processing (Hemanth and Estrela, 2017) indicate a strong potential of DL-based processing in seismic. Before and during the period of this Ph.D. research, a large body of works has been conducted by the geophysical community. The level of its popularity is well illustrated by the number of DL-related papers published.

Among the various types of seismic noise, the coherent types are normally of more concern as these are more difficult to deal with compared to the incoherent noise types. In the suppression of coherent noise, a strategy of data resorting has been widely adopted in the conventional seismic processing methods, in order to break the coherency of the noise in the shot domain and thereby transform the problem to an easier alternative of suppressing the incoherent noise. Use of such a resorting strategy has been inherited to the DL-based methods in earlier studies (e.g. Baardman, 2018). Likewise, in my first investigation of DL in seismic data processing, I studied the deblending of marine seismic data by using a Convolutional Neural Network (CNN) while the blended data were first resorted into the common channel domain (Slang et al., 2019; Sun et al., 2020). It would be more efficient if the separation of seismic signal and coherent noise can be achieved directly in the shot domain, which can further manifest the advantage of DNNs' efficiency.

In addition to efficiency, a rather noteworthy issue is the processing quality of DNN in dealing with real field seismic data. DNN-based methods will probably only be deployed to real processing projects when their processing quality matches or ideally exceeds that of existing conventional tools. To this intent, we need to have an improved understanding of our DNNs instead of using this technology as a "black box". However, fundamental studies are rarely seen. In this context, the work of this thesis has been conducted, in hopes of enhancing a general understanding that can further lead to workflow generalization and automation, saving manual labor and computing time, and

even further improving the processing quality in real processing projects.

1.2 Objectives and scope

This thesis investigates DL in the scope of seismic data processing with a focus on DNNs with supervised learning. The processing tasks are Seismic Interference (SI) noise attenuation and deblending of marine seismic data. The overall objective is to develop DNN-based methods that can be implemented directly in the shot domain for real processing projects. To achieve the objective, the work of this thesis has demonstrated the feasibility of the DNN-based methods for shot-domain processing and the advantage of the DNN-based methods in processing efficiency over existing non-DL commercial algorithm. Furthermore, the work of this thesis explores how to better understand the DNN's behavior with a further aim to improve the DNN's performance. Since signal separation is the main theme of this thesis, the key goal here of improving the DNN's performance is to maintain the fidelity of the wanted signal while efficiently attenuating the coherent noise. To ensure that the needs of real processing projects are always considered, all the work of this thesis has been done on real field marine seismic data.

For SI noise attenuation, the thesis work includes:

Paper I that demonstrates the feasibility of attenuating different types of SI noise in the shot domain by using a DNN. Although the trained DNN model cannot outperform a commercial algorithm in processing quality, it is demonstrated to have a large advantage in terms of processing efficiency.

Paper II is an exploratory study toward demystifying DNN in seismic signal separation. We propose a novel quantitative analysis of the overall DNN model behavior based on synthetic data. The specific task studied is the separation of seismic signal and coherent noise in the shot domain. With a better understanding of the DNN's behavior, we further propose a method to improve the signal fidelity of the DNN and demonstrate its effectiveness on field SI attenuation.

Paper III proposes a DNN-based workflow for SI noise attenuation in the shot domain. We implement the DNN-based workflow on field data acquired from a marine towed streamer survey conducted in the North Sea and compare it with a commercial algorithm. The results show that the proposed DNN-based workflow outperforms the commercial algorithm in processing quality with less

signal leakage and more complete SI removal, validating its feasibility and value for real processing projects.

For seismic deblending, the thesis work includes:

Paper IV proposes a DNN-based shot-domain deblending approach. The complete approach includes a novel and practical strategy to generate training data of high quality and availability for the DNN, and a collection of data conditioning steps that can improve the deblending accuracy of the DNN but barely change the cost of application. The complete proposed approach is demonstrated on field marine blended-by-acquisition data. It performs comparable to a commercial algorithm in the shallow section and shows a large advantage in terms of efficiency. Even though in the deep section it performs slightly worse than the commercial algorithm, it is still able to remove the blending noise effectively.

Chapter 2

Seismic

This chapter introduces the concept of seismic, including acquisition and noise in the data, in order to provide background information on the processing tasks that the thesis work has been aiming to deal with.

2.1 Seismic acquisition

Seismic acquisition refers to the process of sending acoustic energy through subsurface layers and recording the reflections, whether on land or in the ocean. Marine seismic acquisition is conducted for offshore exploration of subsurface hydrocarbons, e.g., oil and gas, or other resources. The two most common acquisition types for marine seismic are towed streamer acquisition with sensors close to the water surface and Ocean Bottom Cable/Node (OBC/OBN) acquisition. Seismic data used in the work of this thesis were all acquired based on the former.

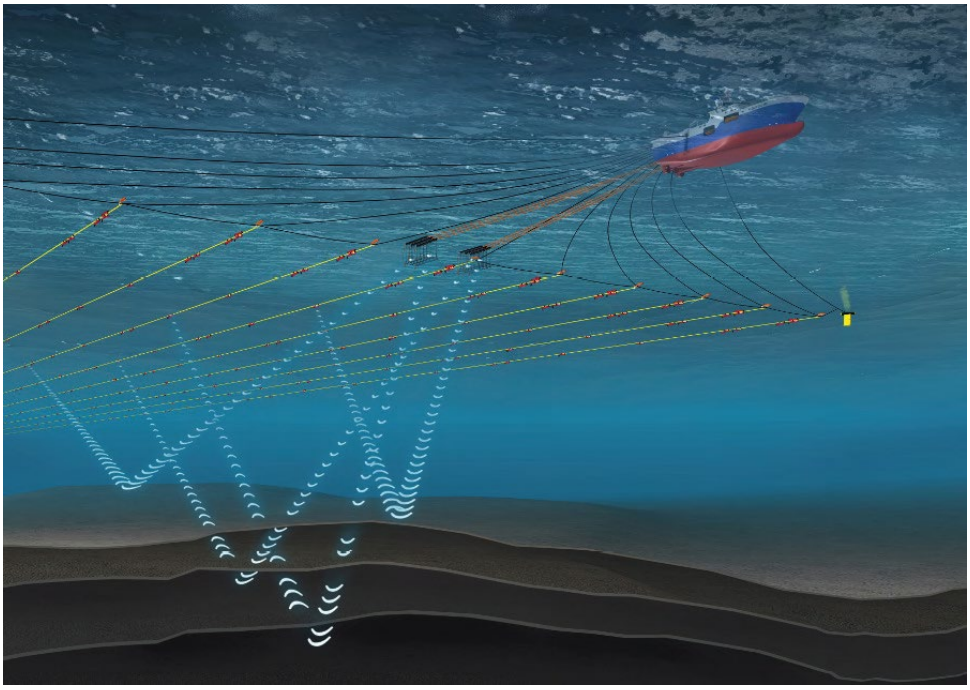

Figure 2.1: A 3D towed streamer marine seismic acquisition, showing seismic vessel, streamer and acoustics sources (Elboth, 2010).

In a 3D towed streamer acquisition, the streamer refers to the cables that detect and record underwater acoustic waves. The acoustic source is usually an air gun array which is also towed by the moving vessel. The air gun array is supplied with high-pressure air and is activated shot with a certain time interval. Following every shooting of the acoustic source, the acoustic energy propagates among the subsurface layers and is partly reflected. The reflections are recorded by the sensors (normally hydrophones) on the cables of the streamer and this is the so-called seismic data that can be used for subsurface imaging after proper processing. Figure 2.1 illustrates the schematic of a 3D towed streamer acquisition with one seismic vessel, two air gun arrays and multiple streamers.

2.2 Noise in seismic data

Seismic data can be generally regarded as a blend of the wanted signal and anything else, i.e., the noise. During a real-world field seismic acquisition, unwanted energy is inevitably recorded which negatively impacts our use of the seismic data. Developing the source and receiver arrays used in the acquisition can reduce some types of noise being recorded, but cannot eliminate external noise. Therefore, how to effectively and efficiently separate noise from wanted signal has become an essential task in seismic data processing.

In general, noise in seismic data can be divided into two categories: coherent noise and incoherent noise. Incoherent noise refers to noise having low correlation with neighboring channels, which appears more or less randomly. Swell noise and ambient disturbance are two types of incoherent noise commonly seen in marine seismic data. Swell noise is usually a high amplitude and low frequency (0~10Hz) noise caused by rough weather conditions. Ambient disturbance is usually lower in amplitude and often higher in frequency, and can be caused by for example machinery, tides, wind, rain, etc. (Slang, 2019; Hlebnikov et al., 2021).

Compared to incoherent noise, coherent noise is of more concern as typically being more difficult to suppress. As the name suggests, coherent noise refers to noise having coherency between channels. It can be linear or non-linear. An example of linear coherent noise is the direct wave which propagates in a straight path along the water surface directly from the source to the receivers. Examples of non-linear coherent noise are multiples and ghost. The two noise types I deal with in this thesis, SI noise and blending noise will be described in more detail in the next section. Being generated by a seismic source, they are

both broad banded, typically containing all seismic frequencies (2-250Hz), coherent and high amplitude. SI noise can be observed as linear or non-linear within one survey depending on the relative position and distance of its source origin to the receivers. Blending noise is non-linear as it is essentially another shot gather fired later than or simultaneously with the desired shot gather being overlapped. Note that in this context a shot gather represents data associated with one source and one streamer.

2.2.1 Seismic interference noise

SI noise is a type of coherent noise that occurs when several seismic acquisition activities are conducted in one area within the same time period. In this case, one option for the acquisition companies is to initiate timesharing, i.e. only one source vessel shoots at a time while the other source vessels keep on standby. This type of acquisition is very inefficient and often results in cost overrun due to substantial downtime. According to Elboth and Haouam (2015), vessels working in the North Sea in the period prior to 2015 could easily spend up to 30% of their available time on standby due to such a timesharing. Besides, timesharing was also found difficult to implement in locations where more than two vessels were operating at the same time (Elboth and Haouam, 2015). To reduce the costly time sharing, the industry has devised a more advanced strategy, i.e., to continue shooting as much as possible under the premise of proper control of the SI noise moveout and its arrival time on the seismic records (Dhelie et al., 2013; Elboth and Haouam, 2015; Laurain et al., 2016; Hlebnikov et al., 2021). The key point here is to avoid SI noise coming from broadside being continuously recorded in time, since such SI noise exhibits coherent from shot to shot and is very difficult to deal with in the processing stage. Except this type, the other types of SI noise can generally be attenuated by first randomizing the noise via data resorting, followed by a prediction filtering (Gülünay, 2008; Elboth et al., 2010; Zhang and Wang, 2015).

The characteristic appearance of SI noise can be coherent linear or non-linear (curved) events in the shot domain depending on the position of the external source generating such noise. Within one survey, the angles of incidence for SI noise may differ greatly from sail line to sail line depending on the relative placement of the external source to the receivers. Likewise, the amplitudes of SI noise within one survey also vary depending on the relative distance between the external source and the receivers. In general, the amplitude of SI noise tends to be high since this noise is generated by powerful

dedicated source for seismic exploration and it travels as guided waves in the water column. SI noise tends to be well preserved over large distances (Akbulut et al., 1984; Jansen et al., 2013) and may overlap with wanted signal reflected from sub-surface layers with much lower amplitude.

According to the relative direction of the external source to the receivers, SI noise can be divided into three classes: SI noise coming from ahead, SI noise coming from broadside and SI noise coming from astern of the recording vessel. Figure 2.2 illustrates shot gathers contaminated by SI noise with varying angle and distance from the external source to the recording vessel. When the external source is ahead of the recording vessel as shown in Figures 2.2a and 2.2b, the alignment of SI noise appears similar to the wanted seismic signal since they originate from the same direction. Compared to Figure 2.2b, SI noise in Figure 2.2a is almost linear as the distance from the external source to the recording vessel is longer. The further away the external source, the more linear the SI noise appears in the shot domain. It is hard to give a specific distance as this is a gradual shift. In general, the strictly linear structure of SI noise only yields when the origin of noise is further away than approximately 20km to 40km. For SI noise events originating from sources closer than approximately 20km, the events appear more curved. SI noise traveling in the water column may have a slightly changing velocity depending on the level of salinity and temperature in the water, but it can be approximated at 1500m/s (Sun et al., 2020). Figure 2.2c shows SI noise coming from astern with a large distance between the external source and the recording vessel. SI noise in this case appears similar to Figure 2.2b, but mirrored along the offset direction due to the opposite direction of the external source.

Figure 2.2d shows SI noise coming from the side of the recording vessel within close proximity. The amplitude of the SI noise is very high and can mask the underlying wanted signal. The moveout of the SI noise in this case has a significant curvature showing similarities to the wanted signal in kinematic and can therefore be difficult to remove. Figure 2.2e shows SI noise coming from the side of the recording vessel when the external source is further away. In this case, SI noise appears as nearly horizontal stripes. In real field acquisition, more than one external source may pass the recording vessel during the busy season, resulting in more than one type of SI noise recorded. An example is given in Figure 2.2f where SI noise coming from ahead and astern are recorded at the same time.

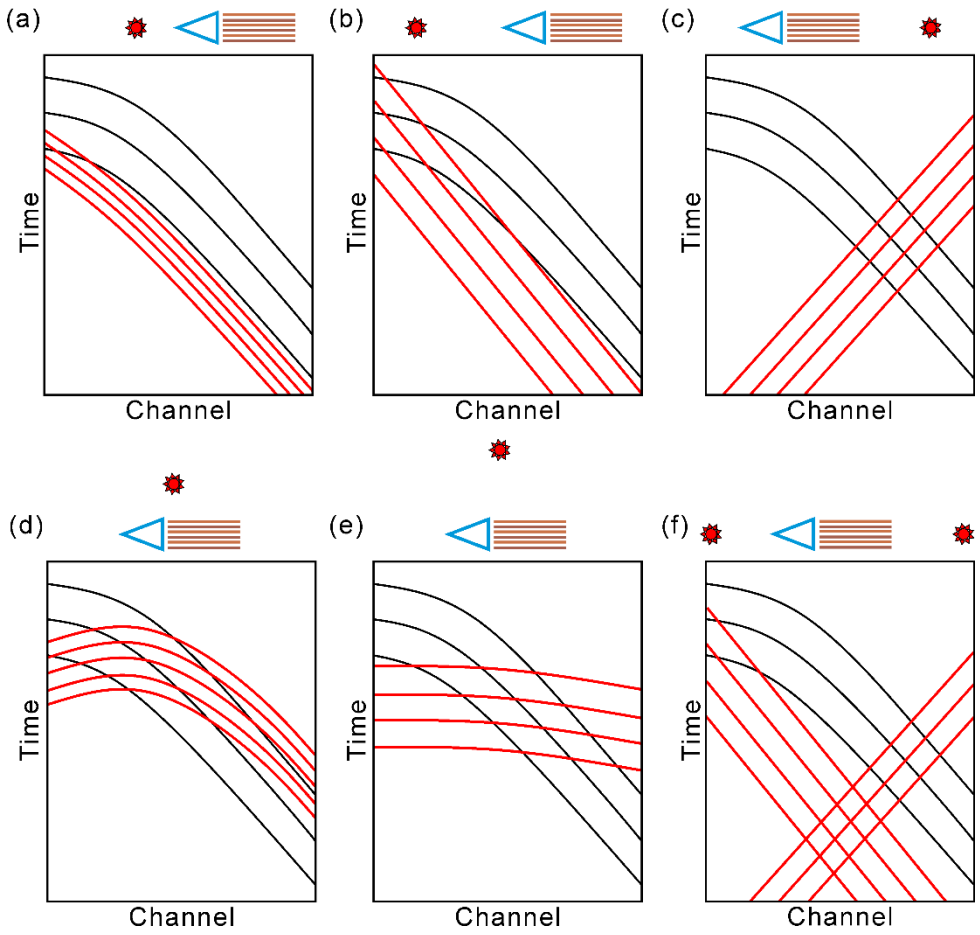

Figure 2.2: Schematics of shot gathers contaminated by SI noise from external source at different positions. In each case, the blue triangle represents the recording vessel with streamers towed and the red spot represents the external source. In each subfigure, seismic reflections from the linked source are drawn in black while SI noise from an external source is in red (adapted from Hlebnikov et al., 2021).

2.2.2 Blending noise

In conventional seismic acquisition, a sufficiently large time interval between successive shot records is typically chosen to avoid the overlap of useful reflection events from shot to shot. This implies that the source domain is often poorly sampled since the total number of shots needs to be kept at an acceptable minimum to reduce operational costs (Berkhout, 2008). To overcome such limitations in efficiency, the concept of blended acquisition has been introduced, where two or more shots are fired overlapping or almost simultaneously (Barbier, 1982; Timoshin and Chizhik, 1982; Vaage, 2005; Beasley, 2008; Berkhout et al., 2010).

The field blended survey I implement my DNN-based approach on in the work of this thesis employs a special source-over-streamer acquisition technology termed as “TopSeis” (Vinje et al., 2017). Therefore, I give a brief introduction to this acquisition technology in this section, but it worth noting that the DNN-based approach I developed can be applied to blended surveys using other acquisition methods/setups as well. TopSeis is a novel marine towed-streamer seismic solution where the location of the source is on top of the seismic spread. To clearly illustrate the configuration of a TopSeis blended acquisition, an example is given in Figure 2.3. This technology has recently been applied in several real acquisition surveys e.g. the Castberg field in the Barents Sea (Vinje and Elboth, 2019; Poole et al., 2020).

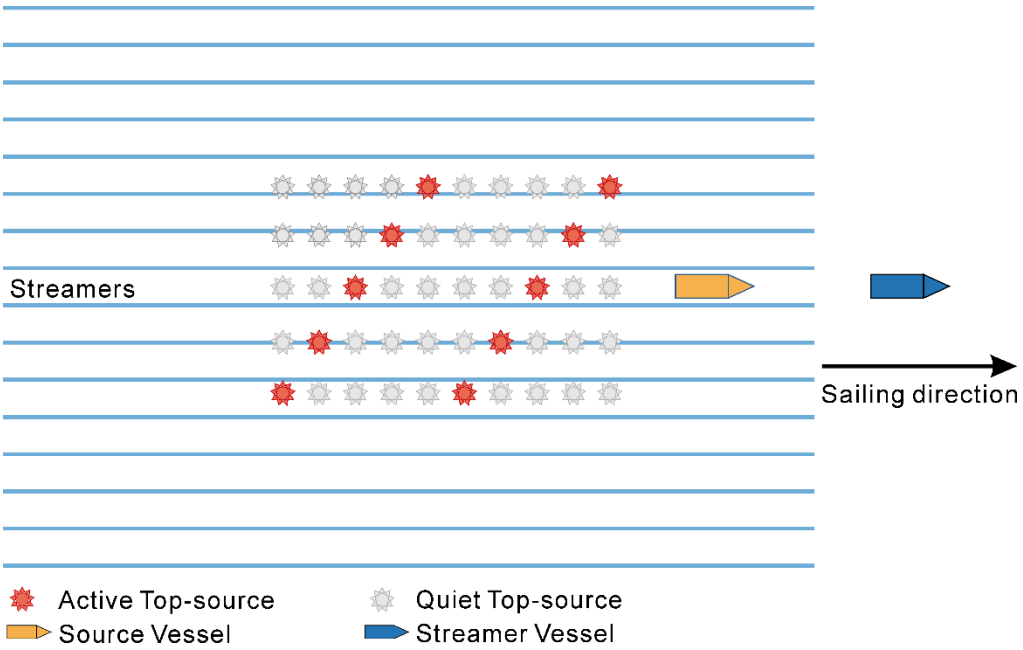

Figure 2.3: Lay-out of a TopSeis blended acquisition.

Examples of data acquired from a TopSeis blended acquisition in respectively the shot domain and the channel domain are schematically shown in Figure 2.4. Figure 2.4a represents the primary-source gather, and Figure 2.4b illustrates the blending noise that is the corresponding shot gather from the overlapping source. The purple box in Figure 2.4c highlights the blended section of a blended shot gather. The blue dotted curve in the blended section represents the primary-source events to be restored.

Due to shot point interval, blending noise comes into the shot records at larger two-way traveltimes (TWTs) than the primary-source events, resulting in a very poor Signal-to-Noise Ratio (SNR) in the blended section (the purple

boxes). To well illustrate how strong the blending noise is compared to the primary-source events to be recovered, Figure 2.5 shows a typical example of the SNR in the blended section, which is obtained based on real field seismic data acquired from a blended TopSeis acquisition in the Barents Sea. As we can see, the amplitude of the blending noise can be more than one order of magnitude higher than that of the primary-source events.

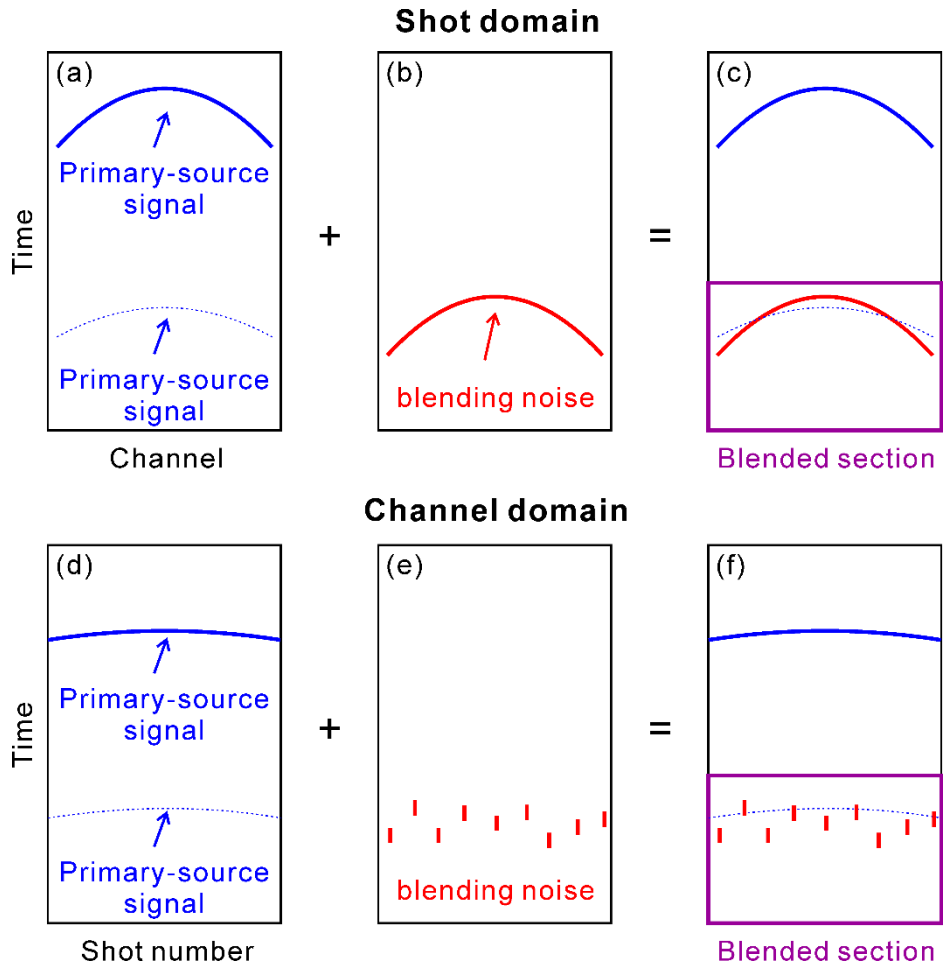

Figure 2.4: Schematics of blended source-over-streamer data in respectively the shot domain and the channel domain. The blue lines represent the primary-source events. The red lines represent the blending noise. The purple box highlights the blended section.

As displayed in Figure 2.4, in the shot domain, the coherent character of the blending noise closely resembles that of the primary-source events. This causes a rather challenging problem of shot-domain deblending. In order to reduce the difficulty of this processing task, a small random jitter between the firing of shots is introduced in a blended acquisition (Elboth and Vinje, 2020). Therefore, we can break the coherency of the blending noise when aligning the

consecutive blended shot gathers by the primary-source events through a strategy of data resorting. This is illustrated in Figures 2.4d, 2.4e and 2.4f. As we can see, after resorting into the channel domain, seismic signals from the primary-source preserve their coherent nature but the blending noise is transformed to incoherent distributions.

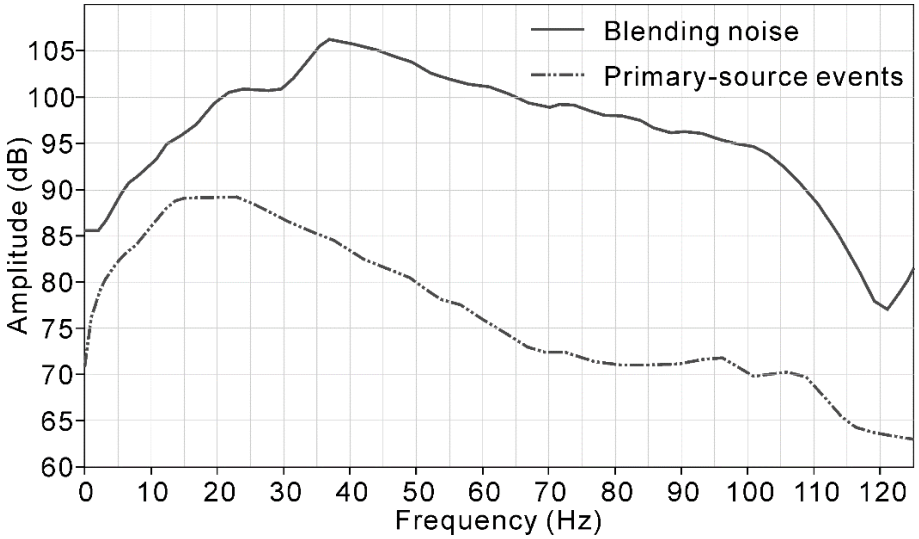

Figure 2.5: Example of SNR in the blended section of a typical blended-by-acquisition source-over-streamer shot gather from the Barents Sea.

Note on terminology: The word 'channel' is used as terminology in both fields of seismic processing and Machine Learning (ML) with different meanings. In this chapter, "channel" is used in the seismic sense, e.g., the channel domain. As defined in the SEG Dictionary entitled "Encyclopedic Dictionary of Applied Geophysics" (Sheriff, 2002), "channel" in seismic processing refers to "A single series of interconnected devices through which data can flow from source to recorder. Seismic systems may have thousands of channels allowing the simultaneous recording of energy from thousands of geophone groups."

In the next chapter, "channel" will be used in the ML lingo, which is transmitted from its definition in the field of conventional image processing. In general, natural images can be represented by third-order tensors, characterized by height, width and the number of channel(s). The height and width of an image relate to spatial information, whereas the concept of channels assigns a multi-dimensional representation to each pixel location. As an example, digital color images are represented by three standard channels (RGB channels) which reflect the amount of those three primary colors. Each of the three channels can be extracted separately and it will exhibit the same size (height and width) of the original color image. In this context, 2D seismic data can be regarded as grayscale images with a single channel only.

Chapter 3

Deep learning

This chapter introduces DL, the technology I studied in the work of this thesis for the processing of seismic data. DL, a branch of ML in AI, enables computational networks that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction (LeCun et al., 2015). In its ideal form, DL represents the science that allows computers to learn without explicit programming. The essence of DL is a DNN that attempts to mimic the human brain to learn from large amounts of data. In this context, we can bring in another concept: Artificial Neural Networks (ANNs). ANNs are models inspired by biological neural networks wherein neurons interact by sending signals in the form of mathematical functions between layers (Hjorth-Jensen, 2020). Neurons are the most basic components and computational units of an ANN. They contain adaptive weight and bias variables which can be tuned by a learning algorithm. An arbitrary number of neurons form a layer. An input layer, hidden layer(s) and an output layer form an ANN (Figure 3.1). A DNN refers to an ANN that has multiple hidden layers between the input and output layers.

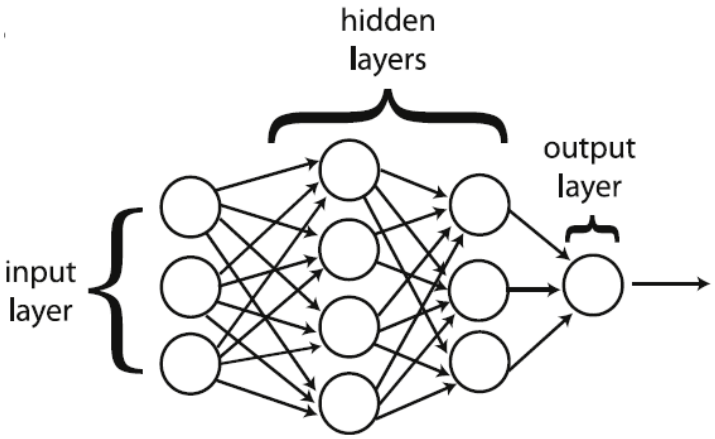

Figure 3.1: Example of an ANN with two hidden layers. Neurons are arranged into layers with the output of one layer serving as the input to the next layer (Mehta et al., 2019).

The history of DNNs is not short (Bishop, 1995), in many textbooks, DNNs are described as having re-emerged to prominence after being rebranded

as DL (Hinton et al., 2006; Hinton and Salakhutdinov, 2006; Mehta et al., 2019). The highlight moment of this technology for truly capturing people's attention was in 2012 when Krizhevsky et al. (2012) presented AlexNet, a GPU-based DNN, on the ImageNet Large Scale Visual Recognition Challenge. Since then, DNNs have been widely used in various fields of science and engineering. DNNs are capable of different types of learning, which are usually categorized as supervised, unsupervised, and reinforcement learning. The work of this thesis focuses on DNNs with supervised learning, in which a set of training pairs consisting of input and corresponding desired output (also known as learning target or the ground truth) is provided. The DNN can automatically learn a certain pattern from these given examples to map the input to the desired output. Once this mapping is learned, the trained DNN model can be applied to unseen data and provide a prediction according to these learned patterns.

The following subsections explain the building blocks of a DNN. To make the understanding of the coming concepts easier, I first briefly summarize how DL works here. In a DNN as shown in Figure 3.1, the neuros between layers are connected with linear transformations through weights and biases. To bring in non-linearity, a non-linear activation is usually employed in each layer. With a given input, the DNN first forward propagates from the first layer to the last layer. When the feed-forward is completed, a downstream-task-dependent cost function is utilized for measuring the discrepancies between the model predictions and the ground truth. Then, backpropagation performs a backward pass to update the models trainable parameters (weights and biases). Alongside back-propagation, an optimization algorithm is used to find out how to update the trainable parameters.

After introducing the above, I introduce the convolutional layer, pooling layer (downscaling layer), upscaling layer and skip connection. Based on them, the DNNs I used in this thesis work can be built. Since all my DNNs are of the encoder-decoder U-shape architecture, an introduction to the original U-Net is provided. At the end of this chapter, a short overview of the recent applications and developments of DL in seismic signal separation that are relevant to this thesis is given.

3.1 Feed-forward

Feed-forward means starting from the input layer and propagating to the output layer. The forward computation from one layer to the next is known as forward propagation. Assuming we have a set of labeled training examples represented as (x^i, y^i) where x^i represents the i^{th} input and y^i represents the corresponding desired output. The number of layers of the network is L. We define a notation system where the weight associated with the connection between the k^{th} neuron in the $(l-1)^{th}$ layer and the j^{th} neuron in the l^{th} layer is denoted as W_{jk}^l, and the bias of the j^{th} neuron in the l^{th} layer is denoted as b_j^l. Weights and biases are trainable parameters of a network. They are initially random before the training process of the networks starts.

For a network with fully-connected layers, the input vectors are fed forward to all the neurons in the first hidden layer, and the output of the first hidden layer serves as the input to all the neurons in the next layer. Let $f^l(\cdot)$ denote the so-called activation function for the l^{th} layer of the network which is always non-linear. In a network, different layers can have different activation functions. The concept of activation function is introduced in section 3.3. The non-linear transform of the feed-forward can be mathematically expressed as

$$a_j^l = f^l\left(z_j^l\right), \tag{3.1}$$

$$z_j^l = \sum_k W_{jk}^l a_k^{l-1} + b_j^l, \tag{3.2}$$

where a_j^l represents the activation for any neuron j in any layer l. This process of feed-forward can be further simplified in a matrix-vector form as

$$\mathbf{z}^l = \mathbf{W}^l \mathbf{a}^{l-1} + \mathbf{b}^l, \tag{3.3}$$

$$\mathbf{a}^l = f^l(\mathbf{z}^l). \tag{3.4}$$

3.2 Back-propagation

When the feed-forward is completed, the error between the network's estimate and the ground truth is calculated based on the cost function for updating the weights and biases of the network through back-propagation (Hjorth-Jensen, 2020) where we iterate backwards from the last layer of the network. The

concept of cost function is introduced in section 3.4. The goal of back-propagation is to compute the partial derivatives $\frac{\partial C}{\partial w_{jk}^l}$ and $\frac{\partial C}{\partial b_j^l}$ of the cost function C with respect to any weight w_{jk}^l or bias b_j^l in the network (Nielsen, 2019). We denote the cost function as C and the error in the j^{th} neuron in the l^{th} layer as δ_j^l. The error of the output layer can be expressed as

$$\delta_j^L = \frac{\partial C}{\partial a_j^L} \odot f^L(z_j^L), \tag{3.5}$$

where $\odot$ represents the Hadamard product (also known as Schur product). In equation 3.5, the term $f^L(z_j^L)$ measures how fast the activation function is changing at z_j^L. The term $\frac{\partial C}{\partial a_j^L}$ measures how fast the cost is changing as a function of the j^{th} output activation (Nielsen, 2019). This equation can be rewritten in a matrix-vector form, as

$$\boldsymbol{\delta}^L = \nabla_a C \odot f^L(\mathbf{z}^L), \tag{3.6}$$

where $\nabla_a C$ denotes a vector whose components are the partial derivatives $\frac{\partial C}{\partial a_j^L}$. The general equation representing moving the error backward through the network from any layer $(l+1)$ to layer l can be expressed as

$$\boldsymbol{\delta}^l = ((\mathbf{W}^{l+1})^T \boldsymbol{\delta}^{l+1}) \odot f^l(\mathbf{z}^l), \tag{3.7}$$

where $(\mathbf{W}^{l+1})^T$ is the transpose of the weights matrix $\mathbf{W}^{l+1}$ for the $(l+1)^{th}$ layer. The rate of change of the cost with respect to any bias b_j^l in the network can be computed by

$$\frac{\partial C}{\partial b_j^l} = \delta_j^l, \tag{3.8}$$

Similarly, the rate of change of the cost with respect to any weight w_{jk}^l in the network can be computed by

$$\frac{\partial C}{\partial w_{jk}^l} = a_k^{l-1} \delta_j^l. \tag{3.9}$$

The above introduced feed-forward and back-propagation can be summarized as Table 3.1.

Table 3.1: Summary of the feed-forward and back-propagation (Nielsen, 2019; Mehta et al., 2019).

1	**Input**: calculate the activation $\mathbf{a^1}$ for the input layer.
2	**Feed-forward**: start with the first layer, exploit the feed-forward architecture to compute $\mathbf{z}^l = \mathbf{W}^l\mathbf{a}^{l-1} + \mathbf{b}^l$ and $\mathbf{a}^l = f^l(\mathbf{z}^l)$ for each subsequent layer $l = 2, 3, \dots, L$.
3	**Error at the top layer**: compute the error of the top layer using $\boldsymbol{\delta}^L = \nabla_a C \odot f^L(\mathbf{z}^L)$.
4	**Back-propagate the error**: propagate the errors backwards and compute $\boldsymbol{\delta}^l = ((\mathbf{W}^{l+1})^T\boldsymbol{\delta}^{l+1}) \odot f^l(\mathbf{z}^l)$ for each layer $l = L-1, L-2, L-3, \dots, 2$.
5	**Calculate the gradient**: the gradient of the cost function is given by $\frac{\partial C}{\partial w^l_{jk}} = a^{l-1}_k \delta^l_j$ and $\frac{\partial C}{\partial b^l_j} = \delta^l_j$.

3.3 Activation function

Activation function has been widely adopted in various DNNs with the main purpose of bringing non-linearity into the network. It performs non-linear mathematical operation upon the linear operations of weights and bias (see equations 3.1 and 3.2) and therefore defines the output of a neuro given an input (Mehta et al., 2019). Activation function is adopted when deriving both forward and backward propagation algorithms. In a DNN, different layers can have different activation functions. Some layers can have no activation functions. In Figure 3.2, four commonly used activation functions are displayed.

The sigmoid function is defined as

$$f(z) = \frac{1}{1 + e^{-z}}. \tag{3.10}$$

The tanh function is defined as

$$f(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}. \tag{3.11}$$

The Rectified Linear Unit (ReLU) function is defined as

$$f(z) = \begin{cases} z, & z \geq 0 \\ 0, & \text{otherwise} \end{cases}. \tag{3.12}$$

The simplicity of the ReLU function means it is computationally efficient. However, it is non-differentiable at zero which in theory is a shortcoming. When employing the ReLU function, some neurons effectively die during the training process which means they stop outputting anything other than 0. This problem is known as the dying ReLUs (Hjorth-Jensen, 2020). To overcome this problem, the Leaky ReLU function which is a variant of the ReLU function is often adopted. It is defined as

$$f(z) = \begin{cases} z, & z \geq 0 \\ \alpha z, & \text{otherwise} \end{cases}, \tag{3.13}$$

where the parameter α is a customized constant that implements a small slope for negative arguments in the ReLU.

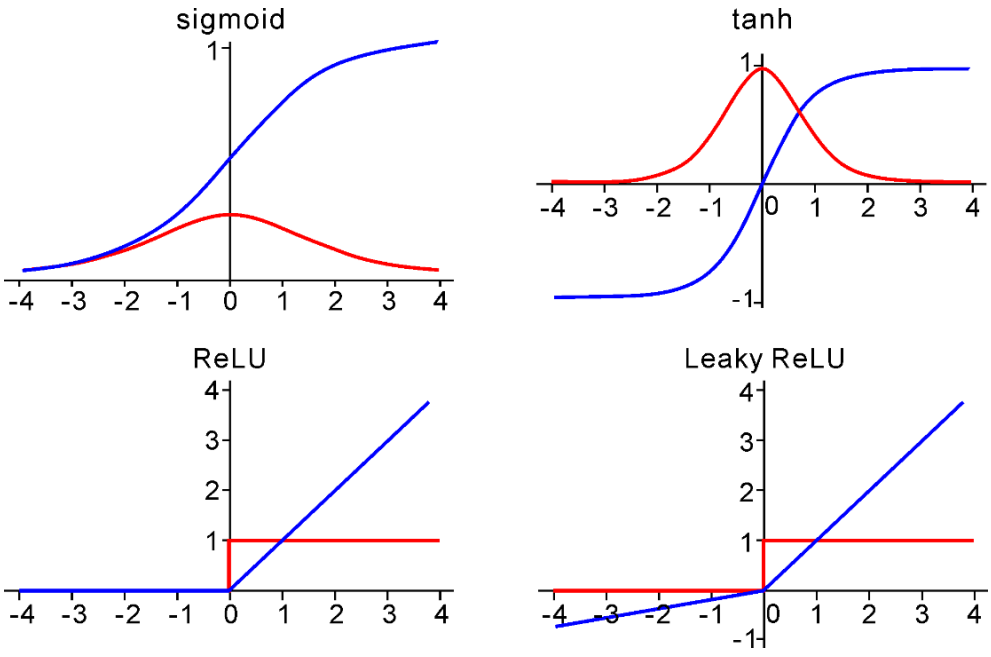

Figure 3.2: Commonly used activation functions (in blue): sigmoid (upper left), tanh (upper right), ReLU (bottom left) and Leaky ReLU (bottom right) and their derivatives (in red).

3.4 Cost function

Cost function is an essential component of DL that allows us to judge how well the DNN model performs. The DNN model is fit by finding the value of the parameters that minimizes the cost function. Each time a batch (a collection of training examples) is passed through the DNN, the loss (or error) is calculated and the parameters are updated accordingly. If the cost function is sub-optimal, the DNN model may break down, fluctuate or take longer than necessary to converge.

Mean Absolute Error (MAE) and Mean Squared Error (MSE) are two most commonly used cost functions. Let $\hat{\mathbf{y}}$ denote the DNN's output and $\mathbf{y}$ denote the ground truth. The MAE cost function can be expressed by an L1 norm as

$$C_1(\mathbf{W}, \mathbf{B}) = \|\mathbf{y} - \hat{\mathbf{y}}\|_1, \tag{3.14}$$

and the MSE cost function can be given by an L2 norm as

$$C_2(\mathbf{W}, \mathbf{B}) = \|\mathbf{y} - \hat{\mathbf{y}}\|_2, \tag{3.15}$$

where $\mathbf{W}$ and $\mathbf{B}$ represent the weights and biases of the DNN, respectively.

3.5 Optimization algorithm

As mentioned above, the point of fitting the DNN model is to find the values of the parameters that minimize the cost function. Therefore, the minimization problem is always a key issue in DL. And this is where optimization algorithms come in. An optimization algorithm is used alongside back-propagation, which helps to know how to change the parameters of the DNN model in order to reduce the losses. One of the most widely used classes of optimization algorithms is the Gradient Descent (GD) algorithm and its generalizations of which the basic idea is: iteratively adjust the parameters of the DNN model in the direction where the gradient of the cost function is large and negative (Mehta et al., 2019). A concept commonly seen when discussing optimization algorithms is the learning rate. The learning rate determines how big the steps are while moving toward a minimum point. Examples of commonly used optimization algorithms are Stochastic Gradient Descent (SGD) (Mehta et al., 2019), Adaptive Gradient Descent (AdaGrad) (Duchi et al., 2011), Root Mean Square Propagation (RMSprop) (Hinton et al., 2012) and Adaptive Moment Estimation

(Adam) (Kingma and Ba, 2014).

3.6 Convolutional layer

DNNs used in the work of this thesis all employ convolutional layers, therefore they can also be defined as CNNs according to Goodfellow et al. (2016) that a CNN has to have at least one convolutional layer. One of the key ideas behind CNN is the local receptive field (Nielsen, 2019). As mentioned in section 3.1, in the fully-connected layers, the inputs from one layer are connected to all the neurons in the next layer, which can be depicted as a vertical line of neurons for a vivid illustration. This is different from the connections within CNN. In a CNN, neurons in the first convolutional layer are not connected to every single pixel in the input image, but only to pixels in their receptive fields. In turn, each neuron in the second convolutional layer is connected only to neurons located within a small rectangle in its last layer (Géron, 2019).

Convolutional layers traditionally apply convolution operations on the input based on a bank of filter kernels (also called convolution matrix or mask) to create feature maps as suggested by the name. Each feature map captures different features from the same image. The number of kernels decides the number of feature maps will be generated. All the kernels of each convolutional layer have the same size. These kernels are normally square matrices consisting of decimal values with a chosen size. The convolution operation is the key step that allows CNNs to be spatial invariant. The standard convolution operation requires flipping of the kernel and placing the kernel over the right bottom section of the image. Then, it performs a dot product between the kernel parameters and the matching grid of the input. Next, the convolution operation slides the kernel from right to left and bottom to top across the input.

An interesting thing to know is that with decades of development, the so-called convolution operation of a convolutional layer has been changed to start from the left upper of the image without flipping of the kernel. The operation will slide the kernel from left to right and top to bottom across the input (Mechelli and Vieira, 2019). Even though in many articles the term convolution operation is still kept, this operation is actually a cross-correlation operation in a mathematical standard. Most present implementations, including TensorFlow (Abadi et al., 2016), Keras (Chollet, 2015) and Pytorch (Paszke et al., 2017), have replaced the above-described standard convolution with the alternative cross-correlation operation. In practice, these two operations can be regarded as

almost equivalent if training a CNN with kernel weights being initialized using the same random procedure. Thus, training the network based on a convolutional implementation will end up with the flipped version of kernels as compared to alternative training based on a cross-correlation implementation.

Figure 3.3 displays the cross-correlation operation in a modern convolutional layer (Anwar, 2020). The size of the kernel (the orange square) is selected as 3×3 in this example, and we denote it as k. Moreover, we need to define two other parameters, i.e., the padding (denoted as p) and the stride (denoted as s). The padding p represents the number of zeros padded around the original input which increase the size to $(i + 2 \times p) \times (i + 2 \times p)$ where i represents the size of the original input. The stride s represents how many pixels we move the kernel each time. The size of the output feature map o can be calculated by

$$o = \frac{i + 2p - k}{s} + 1. \tag{3.16}$$

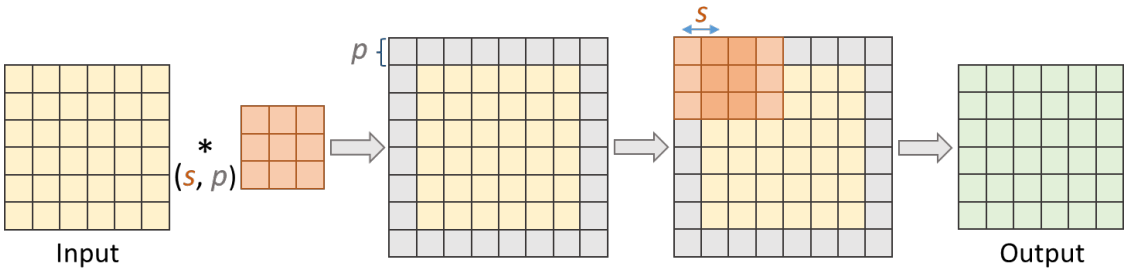

Figure 3.3: A plot showing the process of a convolutional layer (adapted from Anwar, 2020).

Another thing worth noting is that the input and output of a convolutional layer may have more than one channel (the definition of "channel" in the ML lingo can be found in "Note on terminology" at the end of Chapter "Marine seismic"). For example, in the work of this thesis, the input and output data used for the DNN training both have only one single channel in the study of **Paper I**, but both have multiple channels in the studies of **Paper II**, **Paper III** and **Paper IV**. To handle the multi-channel case, multiple kernels are employed where each kernel can produce a different feature of the data. When the input contains multiple channels, we need to construct kernels with the same number of channels as the input. Figure 3.4 shows how a convolutional layer maps an m-channel input to an n-channel output. As we can see, the number of kernels needed here is n, which is the same as the number of the desired output channels.

Each of the n kernels has m channels, which is the same as the number of the input channels.

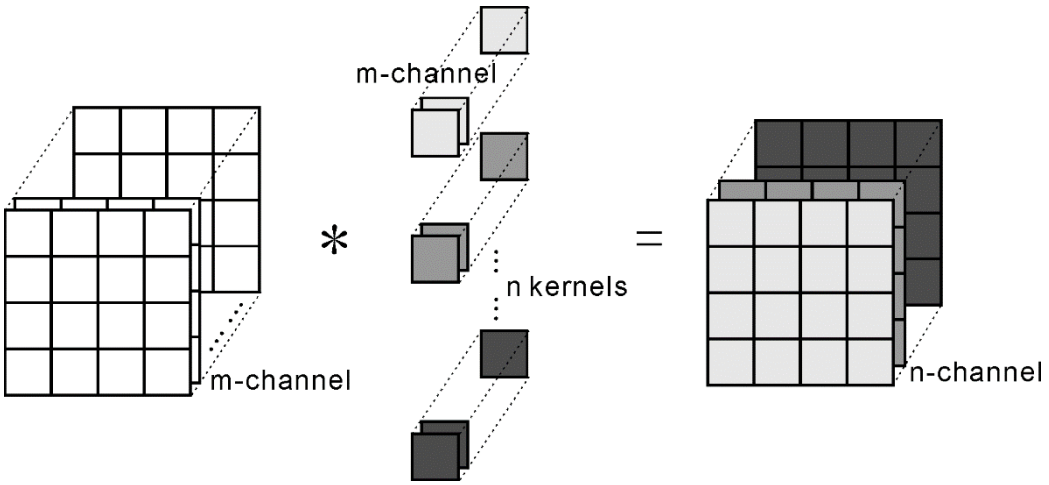

Figure 3.4: Schematic of mapping an m-channel input to an n-channel output through a convolutional layer.

3.7 Pooling layer

Pooling is a process of decimating the information in an image from a fine to coarse scale. As shown in Figure 3.5, pooling layers can be classified into different categories, including: max, average, and min pooling.

We assume the pooling has a filter size of 2×2. For each region, the layer does a specific operation based on the type of pooling. The pooling types will map the 4 values to one output value where: max pooling sets the output value to be the largest of the 4 values, min pooling sets the output value to be the smallest of the 4 values and average pooling computes the average of the 4 values. This process can be viewed in Figure 3.5, where different regions are color coded to easy understanding. Pooling layers are commonly employed in DNNs, especially max pooling. It is a way for the DNN to extract the features of the largest values, which tend to be more important when processing images (Slang, 2019). The use of pooling layers can reduce the size of the data volume and therefore increase the computational speed, but in the meantime, we should also notice that when downsampling is adopted, some information in the image gets lost.

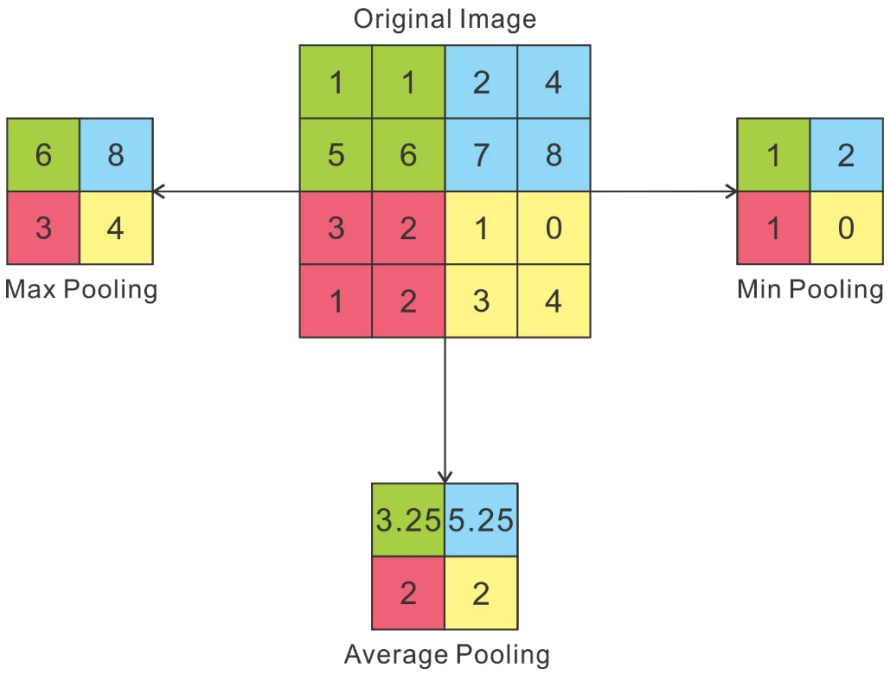

Figure 3.5: A sketch of different types of pooling (Slang, 2019; Sun et al., 2020).

3.8 Upscaling layer

The size of the input image is downscaled when using the pooling layer. In order to go back to the original size of the image, we need to use an upsampling layer or transposed convolutional layer. The upsampling layer adopts the algorithm of nearest neighbor which simply repeats the rows and columns of the input image to achieve the purpose of upscaling as shown in Figure 3.6. It is worth noting that the upsampling layer has no trainable parameters and this is the biggest difference with the transposed convolutional layer.

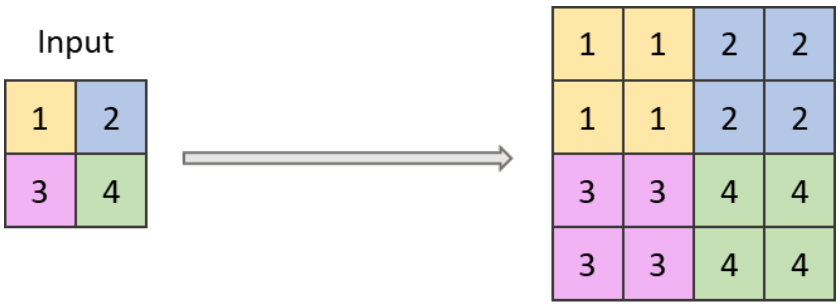

Figure 3.6: An example of upsampling through nearest neighbor.

The transposed convolutional layer operates in a similar manner to the convolutional layer with trainable parameters, but instead of scaling an image down, the image is scaled up. For a given size of the input i, kernel k, padding p and stride s, to calculate the size of the output feature map o, we need to first calculate two new parameters: $z = s - 1$, which represents how many row(s) and column(s) of zeros will be put between the rows and columns of the original input, $p' = k - p - 1$ which represents the number of zeros padded around the image after inserting zeros in between. In addition, there is one more parameter with a constant value is needed i.e. $s' = 1$ which indicates that after applying parameters z and p' to the original input image, the kernel will move on the modified image with a stride length always equals to 1 and perform operations as if in a conventional layer (Anwar, 2020). An example of input $i = 2$, kernel $k = 2$, padding $p = 0$ and stride $s = 0$ is given in Figure 3.7. The size of the output feature map o can be calculated by

$$o = (i - 1) \times s + k - 2p. \tag{3.17}$$

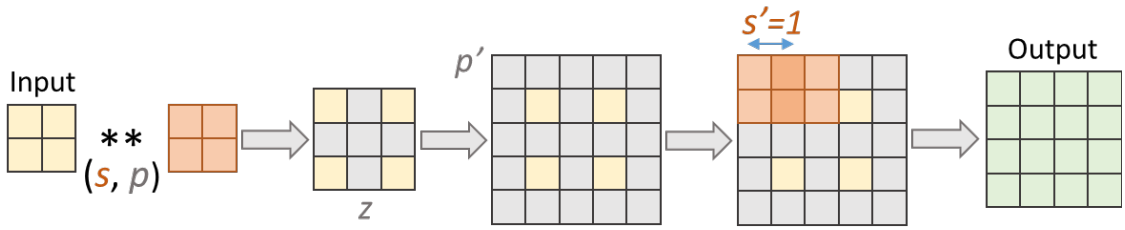

Figure 3.7: A plot showing the process of a transposed convolutional layer (adapted from Anwar, 2020).

3.9 Skip connection

A skip connection represents a connection between an early layer and a later layer in a DNN, thus jumping over all layers in between. Such a connection can be formed employing concatenation or summation. Figures 3.8a and 3.8b visualize the mathematical operation of using respectively summation and concatenation to connect a shallower layer in the DNN and a deeper layer. Summation requires that the size and number of feature maps are the same for the two layers. This is different from concatenation which only requires that the size of feature maps be the same for the two layers. Concatenation can be regarded as "copy and crop" where all feature maps are simply collected together so that the number of feature maps can be different for the two layers. From the perspective of computational efficiency, using concatenation as the way of skip

connection is slower than using summation since the number of feature maps increases.

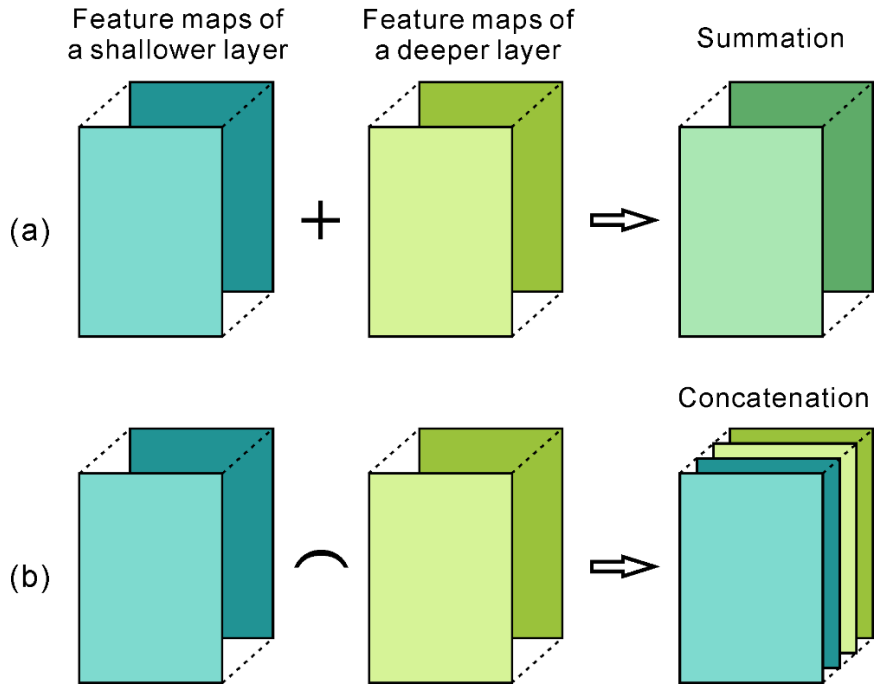

Figure 3.8: Schematics of the operations of (a) summation and (b) concatenation (adapted from Sun et al., 2020).

3.10 U-Net

Up to now, the core algorithm behind DL and the key building blocks of the DNNs used in this thesis work have been introduced. What my DNNs have in common is that they are all of an encoder-decoder structure with skip connections built in a U-shape. This is inspired by the original U-Net initially proposed by Ronneberger et al. (2015) to solve semantic segmentation tasks. The core idea behind U-Net architectures consists of three parts: an encoder transforms the input to feature maps that are essentially sparse representations of the input, a decoder reverse-transforms the feature maps back to the target (Hou and Hoeber, 2020), and skip connections that allow more information to be retained from previous layers of the DNN.

The architecture of the original U-Net is shown in Figure 3.9 (Ronneberger et al., 2015). The building block of the encoder consists of convolutional layers employing a typical filter size of 3×3 and a ReLU activation function. In addition, the introduction of max pooling layers with a pool size of

2×2 (i.e., data are downscaled to half size in both spatial dimensions) yields a multilevel, multi-resolution feature representation. The corresponding building block of the decoder then up-scales the low-resolution feature maps describing large-scale structures with a 2×2 convolution to full resolution feature maps. The skip connections between the encoding path and the decoding path employing a concatenation operation ensures information fusion.

The illustrations of my U-Nets can be found in the attached papers. Therefore, they are not repeatedly displayed in this section.

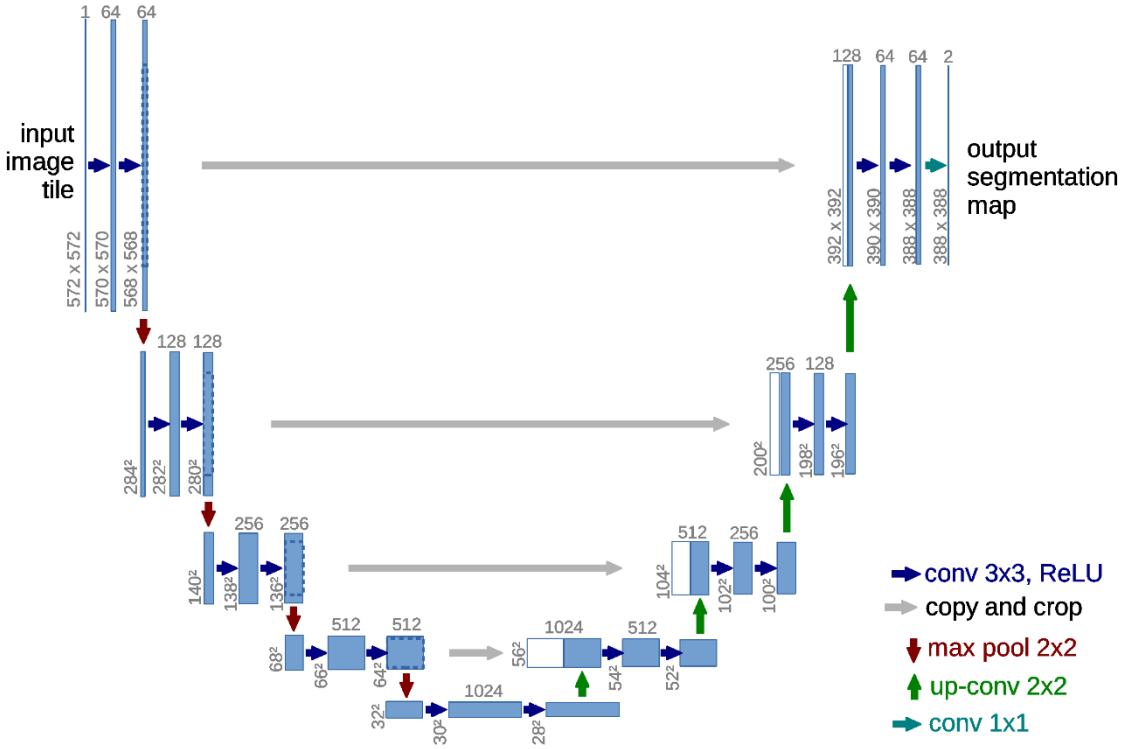

Figure 3.9: U-Net architecture originally proposed by Ronneberger et al. (2015). Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower left edge of the box. White boxes represent copied feature maps. The arrows denote the different operations.

3.11 Overview of deep learning in seismic signal separation

Before closing this chapter, I give a short overview of DL in seismic signal separation as this technology has drawn much attention in recent years and has been applied to many tasks relevant to this thesis. The included papers are all

published since 2018, a year in which we witnessed a rapid increase in the usage of the DL methods in seismic.

Within seismic noise attenuation, many applications of DL have been conducted. However, most of them worked on the removal of random noise, which is an easier task and not of much relevance to the work of this thesis. Except for those, the attenuation of SI noise has been studied by Slang (2019) in his Master's thesis where different DNN architectures were employed and compared. The attenuation of ground roll noise has also been studied by Kaur et al. (2020) who applied a CNN to land seismic data by using signal estimates as the training labels, while in another study, Li et al. (2019) trained their CNN with the ground-roll noise as labels. Klochikhina et al. (2020) proposed to use a CNN to remove noise formed by suboptimal destructive interference within the migration process and demonstrated its application on Brazil and North Sea field data sets. Yu et al. (2019) investigated the application of the Denoising CNN (DnCNN) (Zhang et al., 2017) to attenuate random noise, linear noise and multiples.

Seismic deblending is a major task in seismic signal separation and is also studied in the work of this thesis. In this area, quite a lot of studies of DL-based methods have been conducted. Since 2018, Baardman (Baardman, 2018; Baardman and Tsingas, 2019; Baardman and Hegge, 2020) proposed to classify data patches into "blended" and "non-blended" classes by using a CNN. Besides, they also proposed to deblend the blended data by using a second CNN which shared similarities with the first one in architecture. Both synthetic and field data were considered in their research. In another study of deblending the onshore distributed source array field data, Baardman et al. (2020) proposed to use the earlier acquired unblended data from the vicinity of the distributed source array acquisition to generate the training data for the DNN and demonstrated the feasibility of this strategy. Richardson and Feller (2019) used a U-Net to deblend seismic data in the common offset domain and proposed the use of adjacent offset gathers as additional channels of the input data to improve the performance of the network. Wang et al. (2020) proposed a workflow using a CNN to deblend seismic data in the common receiver domain. In this workflow, the synthetic training data are improved iteratively from velocity-model updates based on deblending predictions from the CNN employing the field acquisition parameters. Wang et al. (2021) used a U-Net first properly trained by synthetic data and then fine-tuned by a selected part of the field data based on transfer learning, to deblend the remaining part of the field data in the common receiver domain. For deblending in case of a two-source simultaneous shooting, Zu et al.

(2020) proposed an iterative CNN-based workflow. Two different field data tests, employing data sorting in respectively the common receiver domain and the common offset domain were presented.

The suppression of multiple and ghost is also of much concern in the field of seismic signal separation, and many interesting studies of DL have been carried out within these two areas as well. Siahkoohi et al. (2019) proposed to use a traditional method e.g. estimation of primaries by sparse inversion algorithm to obtain surface-related-multiple-free data of a subset of the entire field data set for the training of a CNN and then apply the trained model to the remaining part of the data. Qu et al. (2020) proposed a DL-based workflow for surface multiple removal in a shallow-water scenario and demonstrated its feasibility on 2D North Sea field data. The workflow uses a U-Net for gap reconstruction of the highly curved shallow reflections and parabolic Radon transform for the reconstruction of the more planar deep events, followed by the application of closed-loop surface-related multiple estimation to estimate the primaries. For deghosting, Vrolijk and Blacquiere (2021) proposed to use CNN for source deghosting in the coarse common receiver domain. With reciprocity, the training data is prepared by subsampling all shot records with and without receiver-ghost.

Chapter 4

Summary of papers

In this chapter, the results of my work are presented through four papers, including one published peer-reviewed journal paper (**Paper I**), one manuscript resubmitted to the journal after moderate revision (**Paper IV**), one manuscript submitted to the journal (**Paper II**) and one manuscript ready for submission (**Paper III**).

4.1 Paper I

Attenuation of marine seismic interference noise employing a customized U-Net,
Jing Sun, Sigmund Slang, Thomas Elboth, Thomas Larsen Greiner, Steven McDonald and Leiv Jacob Gelius, 2020, *Geophysical Prospecting*, Vol. 68, no. 3, 845-871.

When multiple marine seismic surveys are carried out simultaneously in the same area, SI noise becomes a problem. In this paper, we investigate the feasibility of employing a DNN to attenuate SI noise. The DNN we used is a U-Net. We train it to predict SI-free shot gathers from SI noise contaminated shot gathers which are manually blended from SI noise-free shot gathers (they are SI removal results produced from a commercial algorithm) acquired from the North Sea and records containing almost pure SI noise recorded from different directions. After being properly trained, the network is able to process various types of SI noise on a new data set. The network performs well, leaving only minor residuals, except for the case when SI comes from broadside. We further demonstrate that such noise can be treated by increasing the depth of the network.

To further test the performance of the trained network in a strict way, we apply it to another SI-contaminated data set acquired from an area around 300km away from where we get our training data set. The result is compared with a commercial SI noise attenuation algorithm (Zhang and Wang, 2015) in the CMP stacked domain. Even though the results of the network can still not compete completely with the commercial algorithm, it is demonstrated to have

great advantages in computational efficiency. Another important issue is that we demonstrate that DNN can attenuate SI noise directly in the shot domain without data resorting. This is a further advantage over commercial algorithms, which typically require a multi-shot input to break the coherency of the SI noise.

4.2 Paper II

An exploratory study toward demystifying deep learning in seismic signal separation,
Jing Sun and Song Hou, In review.

In recent years, DNNs have been applied to seismic data on various processing tasks. However, among the related publications, fundamental studies are rarely seen. Furthermore, while DNNs have proven to have considerable potential to improve processing efficiency and reduce processing costs, they usually cannot outperform or match conventional physics-based algorithms in terms of processing quality. This is the primary reason why DNNs have not been employed extensively to real processing projects. For seismic signal separation tasks, the critical metric of processing quality always refers to signal fidelity, which denotes the accuracy of a selected algorithm in preserving the true seismic signal. To improve the DNN's signal fidelity to meet the industrial standard, we first attempt to better understand what our DNN learns from the given training data.

In this paper, we propose to investigate the overall DNN model behavior in a signal separation task through quantitative analysis of synthetic experiments. Specifically, we simulate three types of seismic signals, i.e. primary-source signal, blending noise and linear noise. Their mixture can be regarded as seismic data from a blended acquisition using a short shot point interval contaminated by linear SI noise. The core task is to train a DNN to separate the three data components from their mixture into three output channels in the shot domain where they are all coherent. Furthermore, we study the impact of injecting random noise into the DNN's training data. In DL, data features are divided into low- and high-levels. High-level features are more globally representative, e.g., the overall curvature of the seismic events. To distinguish the performance of our trained DNN model on the low-level versus high-level features of the seismic events, we artificially create jittered gathers by employing random dither on each trace of the test input data. Such local jitters on the events of the test input provide visible low-level features. In total, three

different DNN models sharing the same architecture (a U-Net), but trained individually, are obtained. Their distinctive training is based on selectively injecting random noise into the training inputs and/or the targets. For comparison and analysis, we apply the three DNN models to the same jittered test data.

From the synthetic experiments, we find that manipulating training data through additional random noise can be a useful way to steer the DNN's focus on learning low- and high-level features of the data, and thereby drive the DNN's behavior. Based on the lessons learnt from the synthetic study, we proposed a new method to improve the DNN's signal fidelity for general seismic signal separation tasks i.e. injecting random noise into both the input data and the target channel of wanted signal when training the DNN. Finally, we demonstrated the effectiveness of this method via attenuating SI noise of field marine seismic data.

4.3 Paper III

DNN-based workflow for attenuating seismic interference noise and its application to marine towed streamer data from the North Sea, **Jing Sun**, Song Hou and Alaa Triki, In review.

In this paper, we propose a DNN-based workflow for separating SI noise in the shot domain. The basis of the proposed workflow is the employment of a DNN with supervised learning. To obtain training and validation data for the DNN with control of the ground truth, we manually blend SI-free shots and an estimate of the SI noise (from now on called SI model). The SI model has to be produced by first processing a small subset of the entire to-be-processed data set via a conventional algorithm. The SI-free shots can be either shots that are real free of SI noise during the acquisition or SI-attenuated shots produced in parallel with the SI model from the conventional algorithm, depending on the specific project.

A list of techniques developed for enhancing the DNN's signal fidelity are used in the proposed workflow. Once we obtain the SI model and SI-free shots, we randomly shuffle the SI model and blend it with SI-free shots and a set of simulated random noise. Instead of using a single channel input, we use adjacent shots on both sides of the to-be-processed shot as additional channels of the input. In this way, although the data slices feed to DNN is in shot domain, the

DNN can also make use of information in the channel domain i.e. seismic events from consecutive shots are continuous whereas the SI noise tends to be discontinuous. During the training and validation processes, the DNN learns to output the SI noise of the to-be-processed shot into one channel and the SI-free shot of the to-be-processed shot along with the intact random noise into another. Once trained, the DNN can be applied to the field SI-contaminated data straightaway, producing results efficiently. At this application stage, the step of injecting random noise into the SI-contaminated DNN input can be skipped.

We demonstrate the proposed DNN-based workflow on field marine seismic data acquired from a 3D towed streamer survey conducted in the North Sea. This survey represents a rather challenging and realistic case where all the sail lines are contaminated by various SI of different characteristics during the acquisition. Therefore, both the SI-free shots and SI model used for DNN training are produced through a non-DL commercial algorithm. The DNN employed in the workflow is a U-Net. After being properly trained, we compared the SI removal result of the proposed DNN-based workflow with the commercial algorithm. The results show that the proposed workflow outperforms the commercial algorithm in both processing quality and efficiency. The proposed DNN-based workflow produces SI removal results with reduced signal leakages, and its advantage in time can further scale up as the size of the processing area increases. This validates the value of the proposed DNN-based workflow in real-world industrial scale seismic processing projects.

4.4 Paper IV

Deep learning-based shot-domain seismic deblending,
Jing Sun, Song Hou, Vetle Vinje, Gordon Poole and Leiv Jacob Gelius, *Geophysics*, Accepted.

In this paper, we develop a DNN-based approach to deblend seismic data directly in the shot domain without any resorting. For real-world seismic deblending through DNN with supervised learning, the first challenge is how to obtain high-quality training pairs. This is a critical issue regarding the effectiveness of the training process and the accuracy of the trained predictions. For this purpose, we present a novel and practical strategy suitable for any type of blended acquisition in which the training data are manually blended by using the unblended shot gathers acquired at the end of each sail line. Accessing such unblended data requires no additional time or labor costs beyond the blended

acquisition. By manually blending these data, we obtain training input with good control of the ground truth and fully adapted to the given survey. To save computational memory and reduce the training time, only the blended sections (defined in Figure 2.4) are fed into the DNN.

In addition, we suggest three data conditioning techniques to further improve the performance of the data-driven DNN in deblending which can be selectively used. First, instead of using single-channel input, we can add adjacent blended shot gathers in the input as additional channels. Second, instead of training the DNN to predict only the primary-source events, we can train the DNN to predict both primary-source events and the down-scaled blending noise. This allows the DNN to make a more comprehensive use of the training data. Third, although in the applications of conventional deblending algorithms, blended shot gathers are always aligned by the primary-source events, in this DNN-based approach, we can align the to-be-deblended shot gathers by blending noise instead of by primary-source events. This is because the starting time of the blending noise varies among the samples due to the employed random jitters of the shooting intervals (can be found in section 2.2.2). Aligning the to-be-deblended shot gathers by primary-source events can therefore lead to a training data set having a relatively high variance.

To test the performance of the proposed DNN-based approach, we apply it to real TopSeis blended-by-acquisition data acquired from the Castberg field in the Barents Sea (Vinje and Elboth, 2019; Poole et al., 2020). The employed DNN is a U-Net. As a benchmark method, we apply a non-DL commercial algorithm. In order to verify the effectiveness of the above suggested data conditioning steps, we also introduce a so-called reference case for the same DNN architecture with no data conditioning applied. All comparisons are carried out in the stacked domain. Implementation on field blended-by-acquisition data demonstrates that introducing the suggested data conditioning steps can considerably reduce the leakage of primary-source events in the deep part of the blended section. The complete proposed approach performs almost as well as a commercial algorithm in the shallow section and shows significant advantages in efficiency. It performs slightly worse for larger TWTs, but still removes the blending noise efficiently.

Chapter 5

Conclusions, discussion and outlook

In this thesis, I have presented DL-based methods for attenuation of SI noise and seismic deblending in the shot domain through four papers. In this chapter, I give conclusions of these papers and present some future research directions based on the discussions.

5.1 Conclusions

To handle the processing task of signal separation, the seismic industry typically implements a physics-based processing flow consisting of multiple procedures which is effective in quality but costly in terms of manpower, computers and time. For the needs of the seismic industry of a cheaper and more automated processing solution, this thesis work has focused on the data-driven technique, DL. On top of that, I have studied and developed DL approaches for directly separating coherent seismic noise from wanted signal in the shot domain. This represents a further advantage over the commercial algorithms, which typically require a data resorting method to break the coherency of the noise, but also indicates a more difficult problem to deal with. Based on consideration of realistic field marine seismic surveys, two real challenging processing tasks which are SI noise attenuation and seismic deblending have been studied.

In **Paper I**, we investigated the attenuation of different types of SI noise in the shot domain through a DNN and demonstrated its feasibility. Even though the results obtained from the DNN did not match the quality of an advanced commercial algorithm, the DNN was verified to have a great advantage in computational efficiency. Moreover, the DNN's capability to process field SI-contaminated data shot by shot showed that it can be suitable for fast-track processing of large data volumes as well as for real-time processing and quality control onboard seismic vessels during the acquisition. The DNN's inferior performance to the commercial algorithm in terms of processing quality also specified our direction for the next research.

In order to improve the DNN's signal fidelity for application to real data

of commercial processing projects, we conducted an exploratory study in **Paper II** to gain a better understanding of our DNN's performance in seismic signal separation. We concluded from a quantitative analysis of the overall DNN model behavior using synthetic data that manipulating the training data by selectively injecting random noise into the input and/or the target can influence the DNN's learning of the low- and high-level features of the data. As a result of our observations, we proposed injecting random noise into both the input data and the target channel of wanted signal during the training process of the DNN as a method for enhancing the DNN's signal fidelity. This method was then demonstrated to be effective on field data examples of a SI noise attenuation task.

In **Paper III**, we proposed a DNN-based workflow of value in real processing projects for shot-domain SI noise attenuation. The proposed DNN-based workflow was demonstrated to be effective and efficient on field marine seismic data contaminated by various types of SI noise. The studied survey was very challenging as it has no sail lines free from SI noise during the acquisition that could be used for manually blending the training pairs for the DNN. Even so, the proposed DNN-based workflow outperformed a non-DL commercial algorithm in processing quality with reduced signal leakage and more complete SI noise removal. The success of the proposed DNN-based workflow in such a project validates the value of DL technology to be deployed in actual seismic processing projects. In addition, this workflow has the potential to be adapted to applications of removing other types of coherent noise.

Paper IV proposed and demonstrated the feasibility of directly deblending seismic data in the shot domain using a DL approach. A practical strategy that generates high-quality training data for the DNN and has no limitation to the configuration of the blended acquisition in use was presented, where the input data were manually blended by the unblended shot gathers acquired at the end of each sail line. The implantation in a blended TopSeis survey showed that, based on this strategy, the DNN after a proper training process efficiently removed blending noise from real blended-by-acquisition data, which also demonstrated how unblended shot gathers can be used to train a DNN in the acquisition domain in a generic sense. In that implantation, the trained DNN performed especially well in the shallower part of the blended section but showed more leakage of primary-source events for larger traveltimes. To address this leakage, a set of data conditioning techniques that can considerably improve the DNN's deblending accuracy while barely increasing the application cost have been developed. With all the data conditioning steps applied, we further compared our results with an advanced commercial

algorithm. The proposed approach was significantly faster and labor-efficient but gave slightly weaker primary-events and loss of high frequencies for the deep part of the same blended section. Despite the present shortcomings, this study has demonstrated the data-driven nature and learning potential of a modern DNN and demonstrated the potential of DL to change the way seismic deblending has been carried out.

5.2 Discussion and outlook

DNN architecture and data quality

Testing and tuning of DNN hyper-parameters are always necessary in the application of DNN to a real processing task. If we further consider how to develop DNN beyond making selection among the existing building blocks, the development of the loss function may be most worthy of attention from my perspective. For example, taking local seismic attributes into consideration when calculating the error between the DNN's output and ground truth should be investigated. In addition, since DNN is a data-driven technique, the quality of the training data is very important for the trained DNN model's accuracy. Although the above-mentioned testing of DNN hyper-parameters is work worth doing, my tests of different hyper-parameters when conducting the study of **Paper IV** found that they were not that determinant to the DNN model accuracy compared to the training data sets. This reflects a major challenge for using DNNs in real processing projects where the ideal ground truth (e.g. noise-free data) is always missed.

For this reason, we proposed a training data generation strategy for shot-domain seismic deblending in **Paper IV**. Our strategy makes use of shot gathers acquired at the end of each sail line that are real unblended from the blended acquisition. However, there can be some more serious cases in real seismic processing where no useful data can be found from the field acquisition. For example, in the SI noise attenuation task we studied in **Paper III**, all the sail lines were unfortunately contaminated by SI noise during the acquisition where no real SI-free shots can be used as the ground truth for DNN training. Two more extreme cases can be demultiple and deghosting, where real multiple-free and ghost-free data never exist in field acquisition. For such cases, running a conventional workflow or forward modeling may be the only practical solution to provide training pairs for the DNN, which makes the study of them worthwhile.

From "black box" to "glass box"

As a data-driven technique, DL does not rely on prior knowledge of the physical system but can directly identify classification or regression mappings from given examples. This is very different from the traditional seismic data processing methods, which are developed in accordance with known physical laws or causalities. The data-driven nature of DL is two-sided. It renders DL a better potential to adapt to problems of big data or complex systems lacking proven physical conclusions, but it also makes DL, to some extent, a "black box" which is easily out of our control.

For example, the convergence of a DNN's loss function is usually uncertain and this uncertainty cannot be easily measured. Two researchers using the same training data or equally informative data and DNNs of the exact same architecture may end up with two very different trained models which extract very different data features. Even in this case, there is still a chance for these two trained models to perform similarly when making predictions. Unlike with physics-based algorithms whose parameters can be manually fine-tuned, the weights and biases of a DNN are automatically updated during the training process. In many cases, even being aware of the DNN's architecture and able to print out the feature maps of its hidden layers, it is still difficult to give an exact answer as to what the DNN indeed learns from the training data. Living with such technology has dragged me into two completely different feelings in the preliminary and late stages of my Ph.D. research. At the early stage, I was eager to verify my DNN's capability of learning to do the selected task. Then, after a very short celebration, I got into another more serious struggle that was to figure out what my DNN was doing and how to put its learning on a more reasonable track.

In this context, an idea that appears naturally is that it would be great if we could pre-insert a certain prior knowledge, e.g. known physical laws, into the DNN before it starts the automatic learning process. A representative example here is the proposal of physics-informed neural networks (PINNs), which is a general framework developed by Raissi et al. (2019) for solving differential equations. The core idea behind PINNs is to introduce the underlying equation into the DNN's loss function as a physical constraint. PINNs have been applied to solve some seismic-related equations, e.g. wave equations (Moseley et al., 2020; Waheed et al., 2020; Song et al., 2021) and have been demonstrated to effectively improve the model accuracy in making predictions on unseen data.

For seismic data processing, up to now, DL-based seismic data processing (e.g. denoising and deblending studied in this thesis) still looks more like conventional image processing on seismic data. We have not discovered solid proof showing that DNNs are capable of automatically extracting the physical relationships between different waves from the massive seismic data. Based on the above, I suggest that in future research within the seismic community, instead of making attempts at different applications, we should pay more attention to understand DNN and then combine a physics-based algorithm with DNN or pre-insert some physical information into DNN to guide its learning. One sentence that can be borrowed from geologists to geophysicists is that there is often no unique interpretation of an observation. When our DNN learns to map an input to a desired output based on the given examples, we should also learn to interpret the DNN's performance based on observations and analysis. Taking this one step further, we may really be able to transfer this "black box" to a "glass box".

References

Abadi, M., P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. A. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zhang, 2016, TensorFlow: A system for large-scale machine learning: 12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16), 265-283, [Online]. Available: https://www.usenix.org/conference/osdi16/technical-sessions/presentation/abadi.

Akbulut, K., O. Saeland, P. Farmer, and T. Curtis, 1984, Suppression of seismic interference noise on Gulf of Mexico data: 54th Annual Meeting, SEG, Expanded Abstracts, 527-529, doi: 10.1190/1.1894083.

Anwar, A., 2020, What is Transposed Convolutional Layer?, [Online]. Available: https://towardsdatascience.com/what-is-transposed-convolutional-layer-40e5e6e31c11

Baardman, R. H., 2018, Classification and suppression of blending noise using CNN: First EAGE/PESGB Workshop on Velocities, EAGE, Extended Abstracts, cp-579-00011, doi: 10.3997/2214-4609.201803017.

Baardman, R. H., and R. F. Hegge, 2020, Machine learning approaches for use in deblending: The Leading Edge, vol. 39, no. 3, 188-194, doi: 10.1190/tle39030188.1.

Baardman, R. H., R. F. Hegge, and P. M. Zwartjes, 2020, Deblending via supervised transfer learning - DSA field data example: 82nd Annual International Conference and Exhibition, EAGE, Extended Abstracts, no. 1, 1-5, doi: 10.3997/2214-4609.202010935.

Baardman, R. H., and C. Tsingas, 2019, Classification and suppression of blending noise using convolutional neural networks: Middle East Oil and Gas Show and Conference, SPE, Conference Papers, SPE-194731-MS, doi: 10.2118/194731-MS.

Barbier, M. G., 1982, Pulse coding in seismology: Springer.

Beasley, C. J., 2008, A new look at marine simultaneous sources: The Leading

Edge, vol. 27, no. 7, 914-917, doi: 10.1190/1.2954033.

Berkhout, A. J., 2008, Changing the mindset in seismic data acquisition: The Leading Edge, vol. 27, no. 7, 924-938, doi: 10.1190/1.2954035.

Berkhout, A. J., G. Blacquière, and D. J. Verschuur, 2010, Multi-scattering illumination in blended acquisition design: 80th Annual Meeting, SEG, Expanded Abstracts, 1251-1255, doi: 10.1190/1.3513071.

Bishop, C. M., 1995, Neural networks for pattern recognition: Oxford University Press.

Chollet, F., 2015, Keras, [Online]. Available: https://github.com/fchollet/keras.

Dhelie, P. E., D. Harrison-Fox, and M. F. Abbasi, 2013, Increasing the efficiency of acquisition in a busy North Sea season — Dealing with seismic interference: 75th Annual International Conference and Exhibition, EAGE, Extended Abstracts, We 14 14, doi: 10.3997/ 2214-4609.20130885.

Duchi, J., E. Hazan, and Y. Singer, 2011, Adaptive subgradient methods for online learning and stochastic optimization: Journal of machine learning research, vol. 12, no. 7, doi: 10.5555/1953048.2021068.

Elboth, T., 2010, Noise in marine seismic data: Ph.D. thesis.

Elboth, T., I. Presterud, and D. Hermansen, 2010, Time-frequency seismic data denoising: Geophysical Prospecting, 58, 441–453, doi: 10.1111/j.1365-2478.2009.00846.x.

Elboth, T., and F. Haouam, 2015, A seismic interference noise experiment in the Central North Sea: 77th Annual International Conference and Exhibition, EAGE, Extended Abstracts, Th N116 03, doi: 10.3997/2214-4609.201413314.

Elboth, T., and V. Vinje, 2020, System and method for generating dithering sequences for seismic exploration: U.S. Patent 10,649,108.

Géron, A., 2019, Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems: OReilly Media.

Goodfellow, I., Y. Bengio, and A. Courville, 2016, Deep learning: MIT press, [Online]. Available: http://www.deeplearningbook.org.

Guo, B., L. Li, and Y. Luo, 2018, A new method for automatic seismic fault detection using convolutional neural network: 88th Annual Meeting,

SEG, Expanded Abstracts, 1951-1955, doi: 10.1190/segam2018-2995894.1.

Gülünay, N., 2008, Two different algorithms for seismic interference noise attenuation: The Leading Edge, 27, 176–181, doi: 10.1190/1.2840364.

He, K., X. Zhang, S. Ren, and J. Sun, 2016, Deep residual learning for image recognition: Proceedings of the 29th IEEE Conference on Computer Vision and Pattern Recognition, 770-778, [Online]. Available: https://openaccess.thecvf.com/content_cvpr_2016/papers/He_Deep_Residual_Learning_CVPR_2016_paper.pdf.

Hemanth, D. J., and V. V. Estrela, 2017, Deep learning for image processing applications, vol. 31, IOS Press.

Hinton, G. E., S. Osindero, and Y. W. Teh, 2006, A fast learning algorithm for deep belief nets: Neural Computation, vol. 18, no. 7, 1527-1554, doi: 10.1162/neco.2006.18.7.1527.

Hinton, G. E., and R. R. Salakhutdinov, 2006, Reducing the dimensionality of data with neural networks: Science, vol. 313, no. 5786, 504-507, doi: 10.1126/science.1127647.

Hinton, G. E., N. Srivastava, and K. Swersky, 2012, Neural networks for machine learning, Lecture 6a, Overview of mini-batch gradient descent, [Online]. Available: http://www.cs.toronto.edu/~hinton/coursera/lecture6/lec6.pdf.

Hlebnikov, V., T. Elboth, V. Vinje, and L. J. Gelius, 2021, Noise types and their attenuation in towed marine seismic: A tutorial: Geophysics, vol. 86, no. 2, W1-W19, doi: 10.1190/geo2019-0808.1.

Hjorth-Jensen, M., 2020, Lectures notes in FYS-STK4155, Data analysis and machine learning: Linear regression and more advanced regression analysis, [Online]. Available: https://compphysics.github.io/MachineLearning/doc/pub/Regression/html/Regression.html.

Hou, S., and H. Hoeber, 2020, Seismic processing with deep convolutional neural networks: opportunities and challenges: 82nd Annual International Conference and Exhibition, EAGE, Extended Abstracts, vol. 2020, no. 1, 1-5, doi: 10.3997/2214-4609.202010647.

Jansen, S., T. Elboth, and C. Sanchis, 2013, Two seismic interference attenuation methods based on automatic detection of seismic interference moveout:

75th International Conference and Exhibition incorporating SPE EUROPEC 2013, EAGE, Extended Abstracts, cp-348-00362, doi: 10.3997/2214-4609.20130886.

Kaur, H., S. Fomel, and N. Pham, 2020, Seismic ground-roll noise attenuation using deep learning: Geophysical Prospecting, vol. 68, no. 7, 2064-2077, doi: 10.1111/1365-2478.12985.

Kingma, D. P., and J. Ba, 2014, Adam: A method for stochastic optimization: arXiv, 1412.6980.

Klochikhina, E., S. Crawley, S. Frolov, N. Chemingui, and T Martin, 2020, Leveraging deep learning for seismic image denoising: First Break, vol. 38, no. 7, 41-48, doi: 10.3997/1365-2397.fb2020048.

Krizhevsky, A., I. Sutskever, and G. E. Hinton, 2012, Imagenet classification with deep convolutional neural networks: Advances in Neural Information Processing Systems, 25, 1097-1105, [Online]. Available: https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

Laurain, R., G. Pattison, T. Elboth, and J. Pollatos, 2016, Cooperating for optimizing seismic acquisition — A case study in the Horda Tampen area: 78th Annual International Conference and Exhibition, EAGE, Extended Abstracts, Tu STZ 03, doi: 10.3997/2214-4609.201600857.

Li, H., D. Chen, and D. Chang, 2019, Ground-roll noise attenuation based on convolutional neural network: Workshop: Fractured Reservoir and Unconventional Resources Forum: Prospects and Challenges in the Era of Big Data, SEG, 69-73, doi: 10.1190/frur2019_18.1.

Mechelli, A., and S. Vieira, 2019, Machine learning: methods and applications to brain disorders: Academic Press, doi: 10.1016/C2017-0-03724-2.

Mehta, P., M. Bukov, C. H. Wang, A. G. Day, C. Richardson, C. K. Fisher, and D. J. Schwab, 2019, A high-bias, low-variance introduction to machine learning for physicists: Physics reports, vol. 810, 1-124, doi: 10.1016/j.physrep.2019.03.001.

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

Nielsen, M. A., 2019, Neural networks and deep learning, [Online]. Available: http://neuralnetworksanddeeplearning.com/index.html.

Paszke, A., S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A.

Desmaison, L. Antiga, and A. Lerer, 2017, Automatic differentiation in PyTorch: 31st Conference on Neural Information Processing Systems, [Online]. Available: https://openreview.net/pdf?id=BJJsrmfCZ.

Poole, G., V. Vinje, E. Kaszycka, K. Cichy, and N. Salaun, 2020, Assessing the Value of Source-Over-Streamer Acquisition in the Barents Sea: 2nd Marine Acquisition Workshop, EAGE, Extended Abstracts, 1-3, doi: 10.3997/2214-4609.202034024.

Qu, S., E. Verschuur, D. Zhang, and Y. Chen, 2021, Training deep networks with only synthetic data: Deep-learning-based near-offset reconstruction for (closed-loop) surface-related multiple estimation on shallow-water field data: Geophysics, vol. 86, no. 3, A39-A43, doi: 10.1190/geo2020-0723.1.

Raissi, M., P. Perdikaris, and G. E. Karniadakis, 2019, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations: Journal of Computational Physics, vol. 378, 686-707, doi: 10.1016/j.jcp.2018.10.045.

Richardson, A., and C. Feller, 2019, Seismic data denoising and deblending using deep learning: arXiv, 1907.01497.

Ronneberger, O., P. Fischer, and T. Brox, 2015, U-Net: Convolutional networks for biomedical image segmentation: International Conference on Medical image computing and computer-assisted intervention, Springer, Conference Papers, 234-241, doi: 10.1007/978-3-319-24574-4_28.

Sheriff, R. E., 2002, Encyclopedic dictionary of applied geophysics, SEG.

Siahkoohi, A., D. J. Verschuur, and F. J. Herrmann, 2019, Surface-related multiple elimination with deep learning: 89th Annual Meeting, SEG, Expanded Abstracts, 4629-4634, doi: 10.1190/segam2019-3216723.1

Slang, S., 2019, Attenuation of seismic interference noise with convolutional neural networks: Master's thesis.

Slang, S., J. Sun, T. Elboth, S. McDonald, and L. J. Gelius, 2019, Using convolutional neural networks for denoising and deblending of marine seismic data: 81st Annual International Conference and Exhibition, EAGE, Extended Abstracts, vol. 2019, no. 1, 1-5, doi: 10.3997/2214-4609.201900844.

Sun, J., S. Slang, T. Elboth, T. Larsen Greiner, S. McDonald, and L. J. Gelius, 2020, A convolutional neural network approach to deblending seismic data: Geophysics, 85, no. 4, WA13-WA26, doi: 10.1190/geo2019-0173.1.

Sun, J., S. Slang, T. Elboth, T. Larsen Greiner, S. McDonald, and L. J. Gelius, 2020, Attenuation of marine seismic interference noise employing a customized U-Net: Geophysical Prospecting, vol. 68, no. 3, 845-871, doi: 10.1111/1365-2478.12893.

Timoshin, Y. V., and A. I. Chizhik, 1982, Method for spatial seismic exploration by the multiple overlapping technique: Russia Patent SU1543357A1.

Vaage, S. T., 2005, Method and system for acquiring marine seismic data using multiple seismic sources: U.S. Patent 6,906,981.

Vinje, V., J. E. Lie, V. Danielsen, P. E. Dhelie, R. Siliqi, C. I. Nilsen, E. Hicks, and A. Camerer, 2017, Shooting over the seismic spread: First Break, vol. 35, no. 6, doi: 10.3997/1365-2397.35.6.89461.

Vinje, V., and T. Elboth, 2019, Hunting high and low in marine seismic acquisition; combining wide-tow top sources with front sources: 81st Annual International Conference and Exhibition, EAGE, Extended Abstracts, 1-5, doi: 10.3997/2214-4609.201900899.

Voulodimos, A., N. Doulamis, A. Doulamis, and E. Protopapadakis, 2018, Deep learning for computer vision: A brief review: Computational Intelligence and Neuroscience.

Vrolijk, J. W., and G. Blacquiere, 2021, Source deghosting of coarsely-sampled common-receiver data using a convolutional neural network: Geophysics, vol. 86, no. 3, v185-v196, doi: 10.1190/geo2020-0186.1.

Waheed, U. B., E. Haghighat, T. Alkhalifah, C. Song, and Q. Hao, 2020, Eikonal solution using physics-informed neural networks: 82nd Annual International Conference and Exhibition, EAGE, Extended Abstracts, no.1, 1-5, doi: 10.3997/2214-4609.202011041.

Wang, B., J. Li, J. Luo, Y. Wang, and J. Geng, 2021, Intelligent deblending of seismic data based on U-Net and transfer learning: IEEE Transactions on Geoscience and Remote Sensing (early access), 1-10, doi: 10.1109/TGRS.2020.3048746.

Wang, S., W. Hu, Y. Hu, X. Wu, and J. Chen, 2020, A physics-augmented deep learning method for seismic data deblending: 90th Annual Meeting, SEG, Expanded Abstracts, 3877-3881, doi: 10.1190/segam2020-w13-07.1.

Yu, S., J. Ma, and W. Wang, 2019, Deep learning for denoising: Geophysics, vol. 84, no. 6, V333-V350, doi: 10.1190/geo2018-0668.1.

Zhang, K., W. Zuo, Y. Chen, D. Meng, and L. Zhang, 2017, Beyond a Gaussian

denoiser: Residual learning of deep CNN for image denoising: IEEE Transactions on Image Processing, vol. 26, no. 7, 3142-3155, doi: 10.1109/TIP.2017.2662206.

Zhang, Z., and P. Wang, 2015, Seismic interference noise attenuation based on sparse inversion: 85th Annual Meeting, SEG, Expanded Abstracts, 4662-4666, doi: 10.1190/segam2015-5877663.1.

Zu, S., J. Cao, S. Qu, and Y. Chen, 2020, Iterative deblending for simultaneous source data using the deep neural network: Geophysics, vol. 85, no. 2, V131-V141, doi: 10.1190/geo2019-0319.1.