

JOINT MASTER'S IN APPLIED GEOPHYSICS

IDEA League

DELFT UNIVERSITY OF TECHNOLOGY, THE NETHERLANDS

ETH ZÜRICH, SWITZERLAND

RWTH AACHEN, GERMANY

Research Module Report

Image Reconstruction using Generative Adversarial Networks

written by

Ying Ting (Rhea) Lau

External Supervisors
from SLB UK

Dr. Rajiv Kumar
Dr. Massimiliano Vassallo

Internal Supervisor
from ETH Zürich

Dr. Dirk-Jan van Manen

Academic year 2023/2024

February 1, 2024

Abstract

This study focuses on exploring the applicability of deep learning, particularly Generative Adversarial Networks (GANs), on the seismic trace interpolation problem. We first compare conventional seismic interpolation methods with the machine learning counterpart. Subsequently, we evaluate the performance of GANs in the context of image reconstruction, leveraging simplified datasets. Given the complexity and computational demands of seismic datasets, we opted to simulate the seismic problem using the more manageable 28x28 pixel MNIST and Fashion-MNIST datasets. These datasets were augmented by introducing masked columns, mimicking the missing trace scenario encountered in seismic data. We employ three GANs, namely Linear model, LeNet-5, and U-NET, to reconstruct MNIST and Fashion-MNIST images. Our evaluation involves qualitative assessments and examination of losses. Notably, U-NET emerges as the most effective, producing high-definition results closest to the original images, as evidenced by MSE losses. Through this approach, we gained insights into the effectiveness of GANs in handling image reconstruction and their potential in addressing seismic trace interpolation. However, challenges like training instability and feature loss due to missing information highlight the need for further model and hyper-parameters refinement in handling larger and more complex datasets.

Contents

1	Introduction	1
2	GAN Theory	3
2.1	Generative adversarial network	3
2.2	Neural network architecture	4
2.2.1	Linear layer model	4
2.2.2	LeNeT-5	5
2.2.3	U-NET	5
3	Experimental results and discussion	6
3.1	MNIST data experiment	6
3.2	F-MNIST data experiment	7
3.3	Discussion	8
4	Conclusion	10
A	Data management plan	11
B	Appendix	12
	Bibliography	15
	References	15

CHAPTER 1

Introduction

In signal processing, we follow the Nyquist rate $f_{Nyq} = \frac{1}{2\Delta t}$ in sampling signals in order to avoid aliasing. However, sampling at least twice the maximum signal frequency could lead to high surveying and time costs. There are two approaches to address this challenge: compressive sampling and data interpolation. Compressive sampling (CS) leverages the principles of sparsity, which relates to the signals of interest, and incoherence, which relates to the sensing modality (Candès & Wakin, 2008). CS theory allows accurate recovery of signals from a significantly reduced set of samples, compared to the traditional Nyquist rate, by sampling only $O(S \log(n/S))$ projections. (Candès & Wakin, 2008) In this study, our focus is on interpolation methods. These methods spatially transform irregularly sampled traces to any desired grid (Kaur, Pham, & Fomel, 2021). There are two major categories of seismic interpolation methods: wave equation methods and signal processing methods (Naghizadeh & Innanen, 2011; Kaur, Pham, & Fomel, 2019; Kaur et al., 2021).

Wave equation methods are based on seismic wave propagation, which requires prior knowledge of the direction of lateral coherence of the event. A velocity model is a prerequisite for interpreting traces (Ronen, 1987). Given an accurate velocity model, wave-equation-based approaches can interpolate waves passing through complex geological structures and subsurface heterogeneities. Parabolic and hyperbolic events can be recovered. However, subsurface velocity models might not be present for the surveying area. These methods also employ a straight-raypath approximation, which might collapse for low-frequency waves (Stolt, 2002).

Signal processing methods include domain transform methods or prediction filter methods. No subsurface information is required to perform traces interpolation. Multiple studies have outlined other benefits of using signal processing methods: F-X domain interpolation is insensitive to random noises (Spitz, 1991), F-K domain interpolation can preserve amplitude and perform artifact-free denoise (Naghizadeh & Innanen, 2011), fast generalized Fourier transform (FTFG) can handle non-stationary seismic data (Naghizadeh, 2012), and 3D sparse time-domain Radon interpolator (TDRI) can reconstruct events with high continuity (Schonewille, Dishberger, & Kapadia, 2014). The major drawback is the sparsity and linearity assumptions in the t-x domain, making these

methods intolerant to highly curved or dispersed events (Spitz, 1991; Naghizadeh & Innanen, 2011; Naghizadeh, 2012).

Deep learning methods come without the aforementioned assumptions, and they do not require any velocity model. Convolutional neural networks (CNNs) combine domain transforms and prediction filter methods but without the linearity assumptions or time/frequency transforms (Oliveira, Ferreira, Silva, & Brazil, 2018). Transforms and filters are learned and applied to the neural networks through an iterative training process. Figure 1.1 shows different algorithms that have been exploited for seismic traces interpolation, such as CNNs, generative adversarial networks (GANs), and auto-encoders (Khosro Anjom, Vaccarino, & Socco, 2023). These deep learning methods yield high reconstruction accuracy even on steep events and can be easily applied to 3D data, but they require a large amount of training data (Kaur et al., 2019, 2021; Khosro Anjom et al., 2023).

In this study, our primary focus is on GANs. The upcoming chapters will delve into the theory behind GANs and explore neural network architectures. Due to the complexity and computational demands of seismic datasets, we decided to simulate the nature of seismic problem using more manageable 28x28 pixel MNIST and Fashion-MNIST datasets. These datasets were augmented by introducing masked columns, mimicking the missing trace scenario encountered in seismic datasets. We employ three GANs, namely Linear model, LeNet-5, and U-NET, to reconstruct MNIST and Fashion-MNIST images. Our evaluation involves qualitative assessments and examination of validation losses.

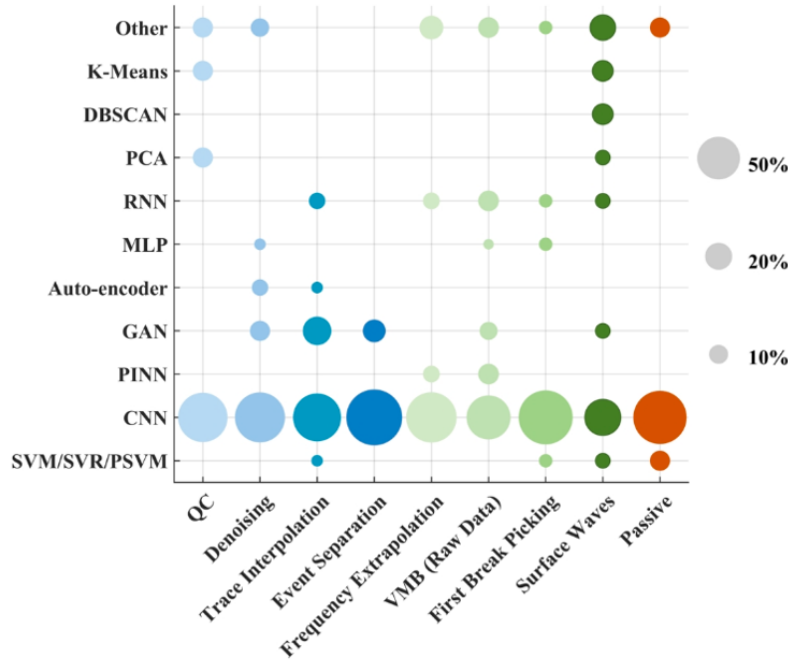


Figure 1.1: Machine learning algorithms used for most prominent seismic processing applications (Khosro Anjom et al., 2023)

2.1 Generative adversarial network

Generative Adversarial Networks (GANs) represent a powerful class of deep learning models comprising generator(s) (G) and discriminator(s) (D). They are fully connected layers which often consist of deep convolutional neural networks (CNNs). (Siahkoohi, Kumar, & Herrmann, 2018) G generates synthetic samples from the embedding space while D distinguishes between generated and real dataset samples. Through iterative training epochs, both G and D evolve by minimizing their respective losses, fostering the generation of more realistic samples by the generator.

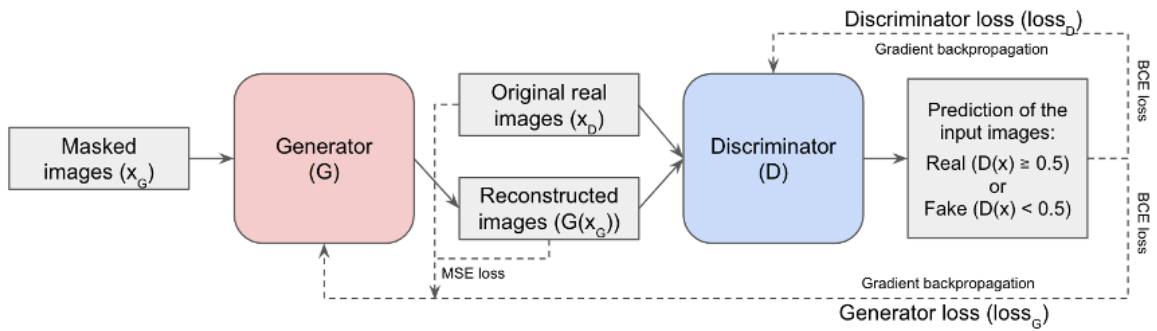


Figure 2.1: GAN architecture (modified from (Saxena & Cao, 2021))

In the context of image reconstruction, our application involves presenting G with images containing missing columns, denoted as x_G , and obtaining interpolated images as the output, i.e., $G(x_G)$. Subsequently, the discriminator D assesses whether the given sample is a reconstructed image ($G(x_G)$) or an authentic dataset image

(x_D), assigning a scalar value between 0 (indicating a reconstructed image) and 1 (indicating a real dataset image). The generator loss ($loss_G$) is computed through a combination of binary cross-entropy (BCE) loss between $D(G(x_G))$ and 1, as well as mean squared error (MSE) loss between $G(x_G)$ and x_D (Equation 2.1). In contrast, the discriminator loss ($loss_D$) is determined by the sum of binary cross-entropy (BCE) loss between $D(x_D)$ and 1, along with binary cross-entropy (BCE) loss between $D(G(x_G))$ and 0. (Equation 2.2) The gradients of these losses are backpropagated to update the parameters of G and D (Saxena & Cao, 2021).

$$loss_G = \underbrace{loss_{BCE}(D(G(x_G)), 1)}_{G \text{ performance on producing synthetic image}} + \underbrace{loss_{MSE}(G(x_G), x_D)}_{\text{Difference between synthetic and real image}} \quad (2.1)$$

$$loss_D = (\underbrace{loss_{BCE}(D(x_D), 1)}_{D \text{ performance on classifying real image}} + \underbrace{loss_{BCE}(D(G(x_G)), 0)}_{D \text{ performance on classifying synthetic image}}) / 2 \quad (2.2)$$

2.2 Neural network architecture

Neural networks (NN) comprise interconnected layers of neurons, with various layer types (e.g., convolution, linear, and pooling) and activation functions (e.g., Rectified Linear Unit (ReLU), sigmoid, and tanh). The architecture's complexity impacts image reconstruction performance and computation time. Three neural network architectures have been investigated on our generator and discriminator, incorporating refined hyperparameters.

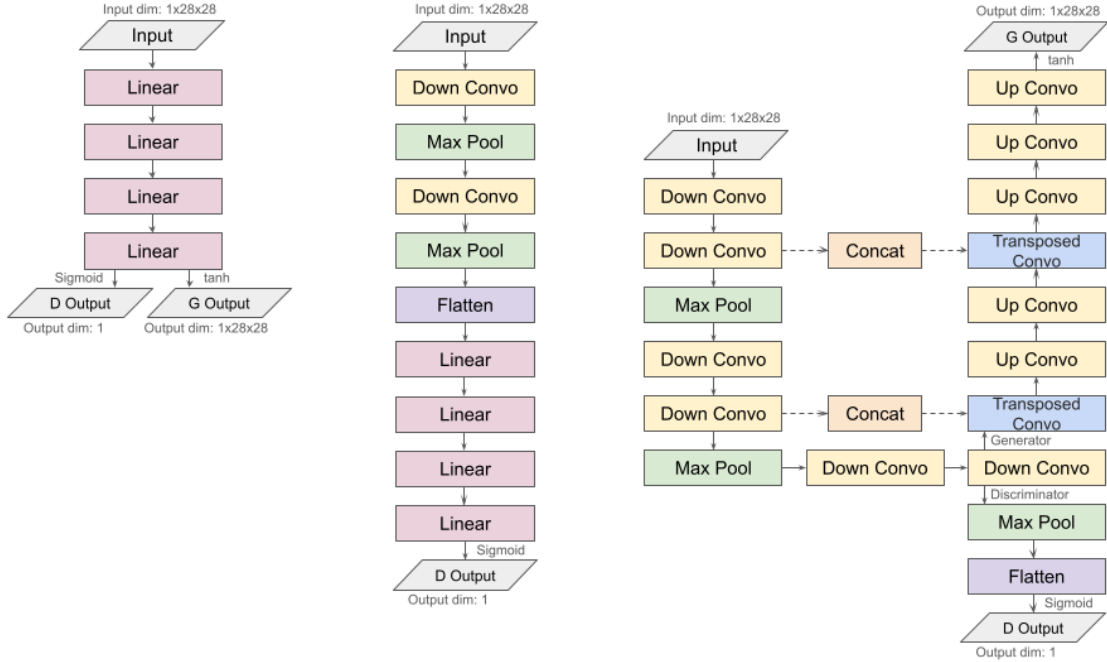


Figure 2.2: Neural network architecture of linear layer model (left), LeNet-5 (center) and U-NET (right)

2.2.1 Linear layer model

This is the simplest model we built for this study, which consists of four linear transformations $y = xA^T + b$ and three ReLU functions. (Figure 2.2 left) ReLU functions introduce non-linearity to NN, enabling them to

learn complex tasks and mitigating the vanishing gradient problem. The generator employs linear layers with an increased output dimension, while the discriminator utilizes linear layers with a decreased output dimension.

2.2.2 LeNet-5

Proposed in 1988, LeNet-5 stands as one of the earliest convolutional neural network (CNN) architectures (LeCun, Bottou, Bengio, & Haffner, 1998), notably designed for handwritten digit recognition. It is composed of two convolutional and pooling layers, a flattening function, four linear transformations, and three ReLU functions. (Figure 2.2 center) Convolutional layers apply a kernel to each image pixel, capturing local patterns, while pooling layers retain the most relevant "summary" of features by downsampling the information. As LeNet-5 downsamples the input dimension to 1, i.e. for classification purposes, we applied LeNet-5 only to our discriminator.

2.2.3 U-NET

Introduced in 2015 by Olaf Ronneberger, Philipp Fischer, and Thomas Brox, U-NET is a convolutional neural network (CNN) architecture specifically designed for image segmentation tasks. Its distinctive U-shaped architecture comprises two key segments: the contracting path and the expanding path. (Ronneberger, Fischer, & Brox, 2015; Long, Shelhamer, & Darrell, 2015)

In our generator model, inspired by U-NET, the contracting path is implemented with six convolutional and two pooling layers. This configuration aims to decrease the spatial resolution of the input image while simultaneously increasing the number of feature channels. (Figure 2.2 right) The contracting path is responsible for extracting essential features from the input image.

Conversely, the expanding path in our model incorporates two transposed convolutional, five convolutional layers, and two concatenation layers. (Figure 2.2 right) This part of the network is designed to elevate the spatial resolution of the feature maps. Additionally, it combines these features with information from the contracting path to generate the final segmentation mask, using the concatenate functions.

In our discriminator model, we leveraged the contracting path of the U-Net architecture, augmenting it with one pooling layer, one flattening function, and one linear layer. Our experimentation revealed that the U-Net architecture yielded the most refined interpolation results, which will be further discussed in chapter 3.

Experimental results and discussion

3.1 MNIST data experiment

The MNIST (Modified National Institute of Standards and Technology) database comprises a collection of 60,000 training images and 10,000 testing images. These images represent handwritten digits and are digitally captured in a grayscale format, contained within a 28x28 pixel bounding box. (Deng, 2012) To replicate the characteristics of the seismic traces interpolation problem, we adopted a strategy of masking 50% of the columns in the images randomly. Subsequently, these partially masked images were fed into the generator. The generator, in turn, produced reconstructed images that effectively addressed the missing traces.

Three GAN architectures were utilized in this experiment. The Linear model incorporates a generator and discriminator, both constructed with the linear layer model (section 2.2.1). The LeNeT-5 model is composed of a generator using the linear layer model and a discriminator implemented with LeNeT-5 architecture (section 2.2.2). Additionally, the U-NET model integrates a generator based on U-NET and a discriminator constructed with a modified U-NET architecture (section 2.2.3).

All three models underwent training using the Adam optimizer with a batch size of 64 and a learning rate of 0.0002. Dropout, with a probability of 0.5, was applied to both the generator and discriminator to enhance model generalization. The training ratio between the generator and discriminator was set to 2:1, emphasizing the importance of the generator's performance in image reconstruction. These parameter choices were meticulously chosen based on iterative experiments. Additional hyper-parameters for model training are detailed in the appendix table B.1.

In Figure 3.1, two generated samples are showcased. It is evident that the U-NET model produces results with the highest definition and bears the closest resemblance to the original images. The Linear model outputs a more generic representation of the digit along with a noisy background. The LeNeT-5 model performs between the

aforementioned two models. Notably, in the example of the digit '4', both the Linear model and LeNeT-5 struggle to discern whether the masked image represents a '4' or '9', leading to a smudged reconstruction in the upper part of the image."

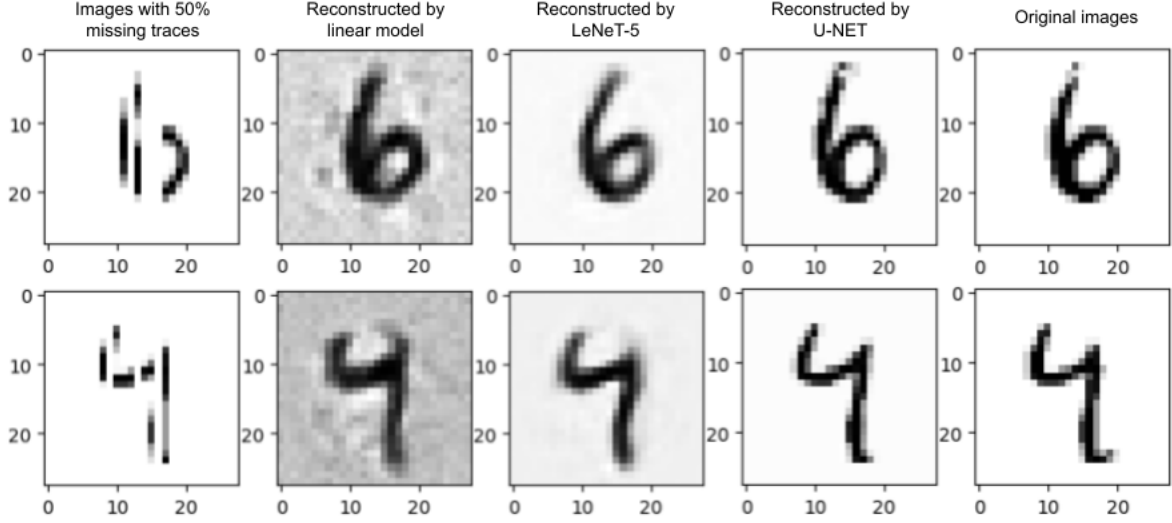


Figure 3.1: MNIST data reconstruction results comparison (from left to right: 50% masked image, image reconstructed by the linear layer model, image reconstructed by LeNeT-5, image reconstructed by U-NET, original image)

3.2 F-MNIST data experiment

Fashion-MNIST (F-MNIST) is a dataset comprised of Zalando’s fashion article images. Each instance in the dataset is represented by a 28x28 pixel grayscale image, associated with a specific label from a set of 10 classes. These classes cover various fashion items, including shirts and sandals. (Xiao, Rasul, & Vollgraf, 2017) The dataset’s size, GAN architectures, and hyper-parameters used in our experiment mirror those of the MNIST experiment. The only distinction lies in the training dataset, where Fashion-MNIST is employed.

In Figure 3.2, the U-NET model exhibits superior performance in reconstructing intricate details of the clothing items. While LeNeT-5 demonstrates more refined edges compared to the Linear model, both models struggle to reproduce patterns on the shirt and accurately locate the straps on the sandal. Additionally, the reconstructed background in both cases appears noisy.

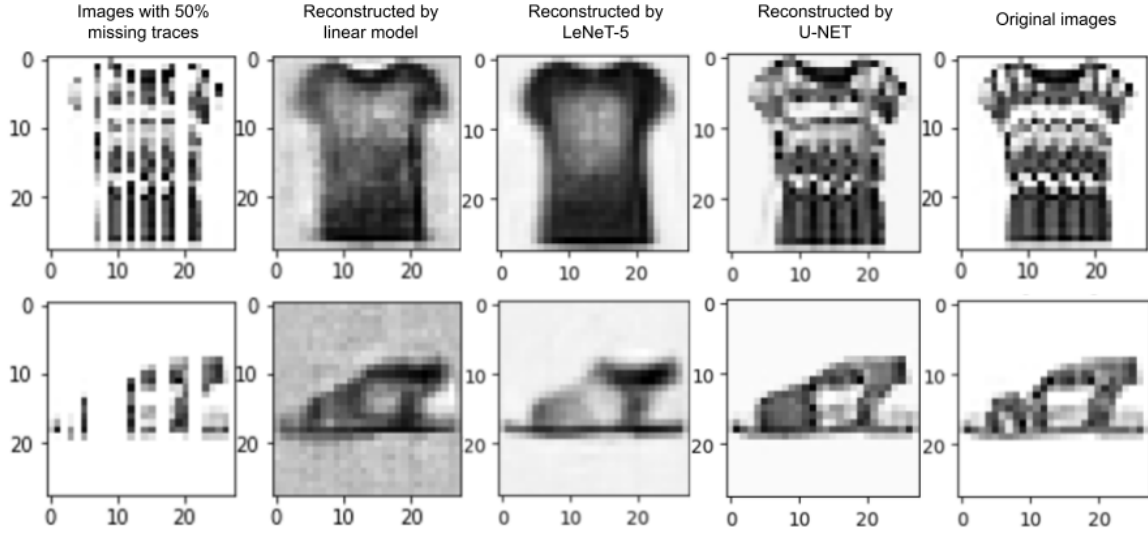


Figure 3.2: Fashion-MNIST data reconstruction results comparison (from left to right: 50% masked image, image reconstructed by the linear layer model, image reconstructed by LeNet-5, image reconstructed by U-NET, original image)

3.3 Discussion

In Figures 3.1 and 3.2, we qualitatively assess the image reconstruction results. For a quantitative analysis, we examine the average loss per epoch of the GANs. The average discriminator losses converge to a value of 0.693, indicating that the generator produces replicas close enough to the original dataset to confuse the discriminator, i.e., the discriminator gives a guess of 0.5 for the input images. Substituting $D(x_D) = 0.5$ and $D(x_G) = 0.5$ into Equation 2.2 yields a result of 0.693.

The discriminator loss incorporates two binary cross-entropy (BCE) losses (equation 2.2), while the generator loss combines BCE loss and mean squared error (MSE) loss (equation 2.1). BCE losses are interdependent on the generator and discriminator within a model. To quantitatively evaluate the reconstruction results across models, we focus on the average generator MSE loss between the reconstructed and original images.

The training loss data indicates the models are well trained by the third epoch. (Figure 3.3) In the final epoch of MNIST validation, the average MSE losses are 0.0064, 0.0150, and 0.0213 for U-NET, LeNet-5, and the Linear model, respectively. For F-MNIST training, the average MSE losses are 0.0065, 0.0162, and 0.0140. Excluding the Linear model, the average MSE loss is lower for the MNIST dataset, as handwritten digits contain fewer details than fashion items. U-NET outperforms the LeNet-5 model by approximately 2.5 times. It's noteworthy that the computation time of U-NET is five times longer than LeNet-5, with 7.18 seconds/iteration and 1.29 seconds/iteration, respectively, due to the increased complexity of the U-NET architecture.

The U-NET generator exhibits greater parameter efficiency, featuring 1,864,002 parameters compared to the Linear model generator's 14,631,184. Despite having an order of magnitude fewer parameters, the U-NET yields

significantly superior results. This underscores that the linear model contains a considerable number of unnecessary parameters.

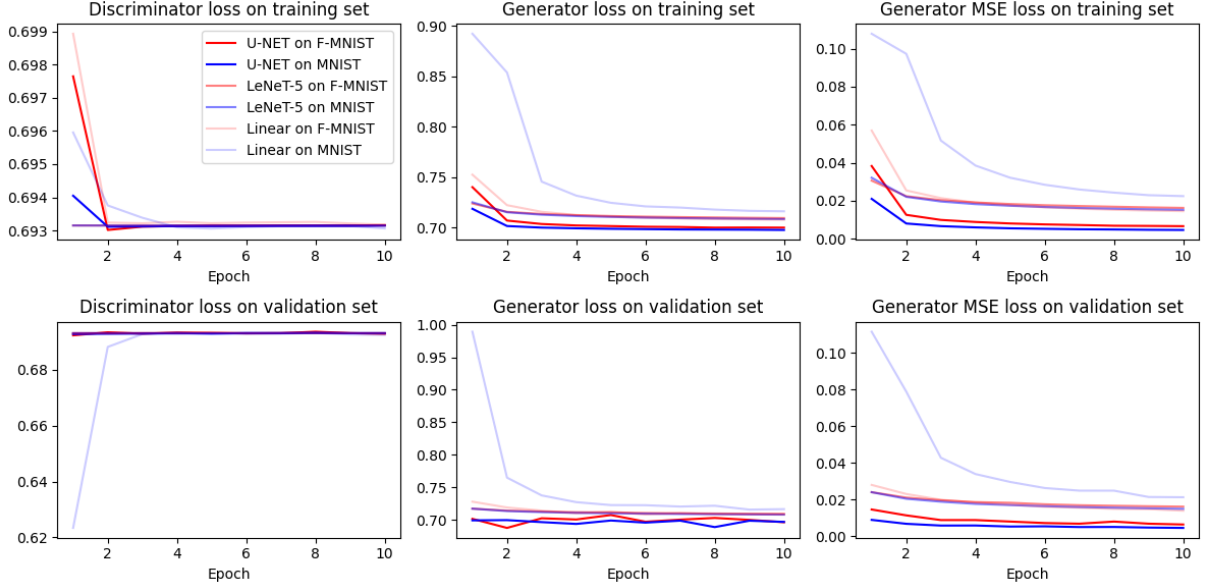


Figure 3.3: Average discriminator loss, generator loss, and generator MSE loss throughout 10 epochs for the three models on the MNIST and F-MNIST dataset

While GANs demonstrate satisfactory image reconstruction results, they are not without limitations. The first limitation is the challenge of training instability. Striking the right balance between the generator and discriminator poses difficulties, and GANs may experience oscillations or fail to converge during training. (Saxena & Cao, 2021) We raised the generator-discriminator training ratio to 2:1 to overcome this challenge. Second, there is the issue of feature loss due to missing information. The application of a masking function to random columns of the masked image can lead to areas with extended missing information, resulting in a blurred outcome (illustrated by the orange area in Figure 3.4). Third, GANs may encounter challenges in recovering high-frequency details. The struggle to capture intricate features, especially in small objects or complex patterns, can lead to potential smudging or loss of fine details during reconstruction (depicted by the red area in Figure 3.4).

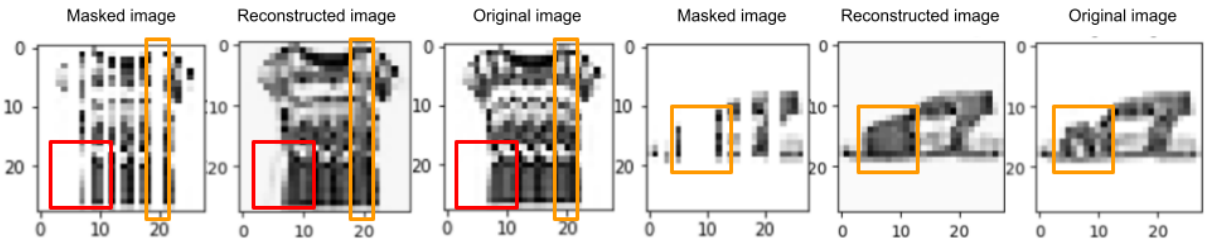


Figure 3.4: Feature loss in the reconstructed image due to missing information (orange) and high-frequency details (red)

CHAPTER 4

Conclusion

In this study, we adopt a deep learning approach for seismic data interpolation, employing three GANs: U-NET, LeNet-5, and Linear model. We assess their performance in image reconstruction using simplified MNIST and F-MNIST datasets. The generators and discriminators, built with U-NET, LeNet-5, or Linear model, iteratively learn features between masked datasets and the original dataset over 10 training epochs, generating reconstructed images at varying resemblance levels. We compare GANs' interpolation results through qualitative examination of output features and quantitative assessment using MSE losses on validation sets. U-NET consistently yields the most refined and accurate reconstruction, while LeNet-5 and Linear model capture the general shapes of hand-written digits and fashion items. Looking ahead to seismic data interpolation, we anticipate GANs to provide satisfactory results with more complex neural network architectures and fine-tuning of hyper-parameters. GANs offer flexibility by not requiring any velocity models or relying on linearity/sparsity assumptions. However, challenges such as training instability, feature loss due to missing information, and high-frequency details recovery need to be addressed.

In future research, we plan to extend our GANs interpolation approach to seismic datasets. An exploratory avenue would involve experimenting with CycleGANs, potentially incorporating U-NET with additional layers for enhanced performance. Additionally, exploring different masking schemes, such as determining the maximum consecutive columns to be masked while retaining features, could provide valuable insights into improving overall performance.

Data management plan

The datasets utilized in this study are sourced from the PyTorch dataset library, *torchvision.datasets.MNIST()* for MNIST datasets and *torchvision.datasets.FashionMNIST()* for Fashion-MNIST datasets. Training and validation sets are imported separately. The Jupyter notebooks for model training, namely **gan_interpolate_linear.ipynb**, **gan_interpolate_lenet.ipynb**, **gan_interpolate_unet.ipynb**, are available on GitHub, along with the plotting notebook **gan_plots.ipynb**. To reproduce the interpolation results using the same masked image, one can utilize the .pth files in the **sample_pics** directory. The PyTorch version used is v2.2.1, and the code was written in Python v3.9.2.

To access the code, you can visit the GitHub repository at: https://github.com/imrhealau/GAN_image_reconstruction

APPENDIX B

Appendix

Parameter	value
Batch size	64
Masking rate	0.4
Learning rate	0.0002
Number of epochs	10
Dropout	0.5
Optimiser	Adam
Training ratio	2 Generator / 1 Discriminator

Table B.1: *Hyper-parameters for model training*

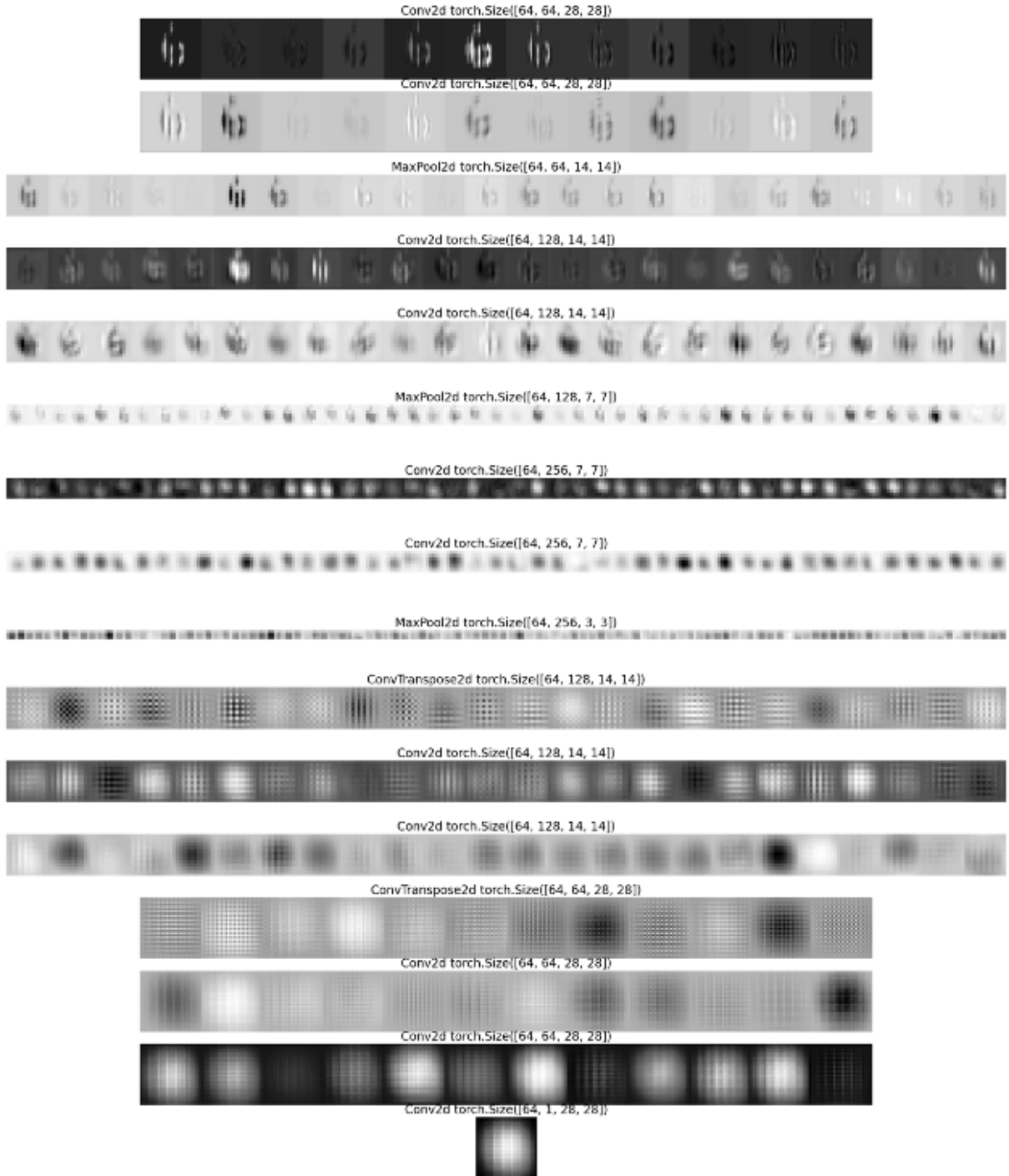


Figure B.1: *U-NET feature map for MNIST input*

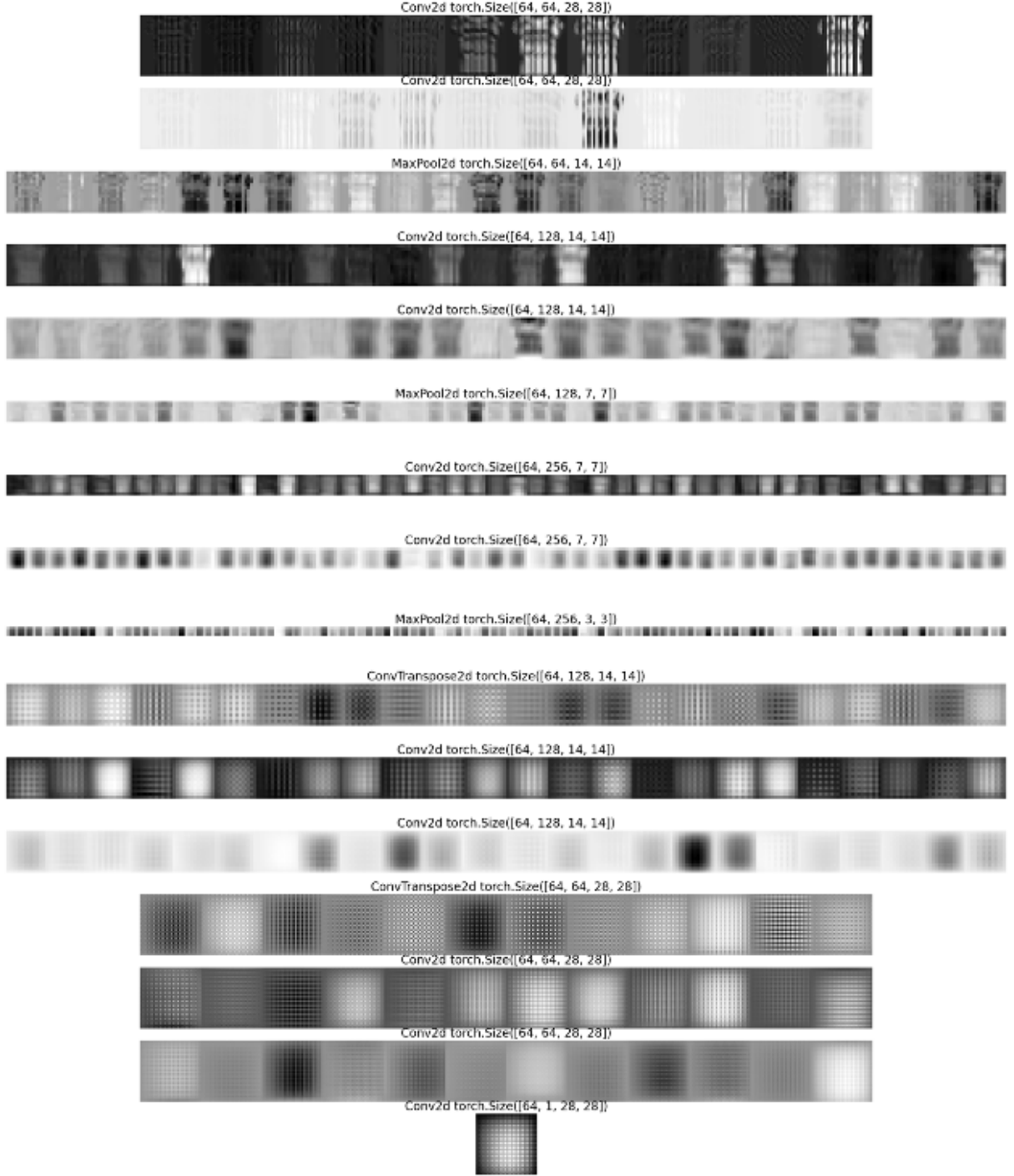


Figure B.2: *U-NET feature map for F-MNIST input*

References

- Candès, E. J., & Wakin, M. B. (2008). An introduction to compressive sampling. *IEEE signal processing magazine*, 25(2), 21–30.
- Deng, L. (2012). The mnist database of handwritten digit images for machine learning research [best of the web]. *IEEE signal processing magazine*, 29(6), 141–142.
- Kaur, H., Pham, N., & Fomel, S. (2019). Seismic data interpolation using cyclegan. In *Seg technical program expanded abstracts 2019* (pp. 2202–2206). Society of Exploration Geophysicists.
- Kaur, H., Pham, N., & Fomel, S. (2021). Seismic data interpolation using deep learning with generative adversarial networks. *Geophysical Prospecting*, 69(2), 307–326.
- Khosro Anjom, F., Vaccarino, F., & Socco, L. V. (2023). Machine learning for seismic exploration: Where are we and how far are we from the holy grail? *Geophysics*, 89(1), WA157–WA178.
- LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278–2324.
- Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3431–3440).
- Naghizadeh, M. (2012). Seismic data interpolation and denoising in the frequency-wavenumber domain. *Geophysics*, 77(2), V71–V80.
- Naghizadeh, M., & Innanen, K. A. (2011). Seismic data interpolation using a fast generalized fourier transform. *Geophysics*, 76(1), V1–V10.
- Oliveira, D. A., Ferreira, R. S., Silva, R., & Brazil, E. V. (2018). Interpolating seismic data with conditional generative adversarial networks. *IEEE Geoscience and Remote Sensing Letters*, 15(12), 1952–1956.
- Ronen, J. (1987). Wave-equation trace interpolation. *Geophysics*, 52(7), 973–984.
- Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention—miccai 2015: 18th international conference, munich, germany, october 5–9, 2015, proceedings, part iii 18* (pp. 234–241).
- Saxena, D., & Cao, J. (2021). Generative adversarial networks (gans) challenges, solutions, and future directions. *ACM Computing Surveys (CSUR)*, 54(3), 1–42.
- Schonewille, M., Dishberger, D., & Kapadia, D. (2014). Comparison of 3d time-domain radon and matching-pursuit fourier interpolation. In *76th eage conference and exhibition 2014* (Vol. 2014, pp. 1–5).
- Siahkoobi, A., Kumar, R., & Herrmann, F. (2018). Seismic data reconstruction with generative adversarial networks. In *80th eage conference and exhibition 2018* (Vol. 2018, pp. 1–5).
- Spitz, S. (1991). Seismic trace interpolation in the fx domain. *Geophysics*, 56(6), 785–794.
- Stolt, R. H. (2002). Seismic data mapping and reconstruction. *Geophysics*, 67(3), 890–908.
- Xiao, H., Rasul, K., & Vollgraf, R. (2017). Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*.