Transfer learning in large-scale ocean bottom seismic wavefield reconstruction

Mi Zhang¹, ², Ali Siahkoohi¹, and Felix J. Herrmann¹

¹ Georgia Institute of Technology
² China University of Petroleum - Beijing
Problem setup

Ocean bottom node (OBN) geometry:

► desirable source sampling
► on the grid sparse receivers on Ocean bottom
Problem setup

Ocean bottom node (OBN) geometry:

► desirable source sampling
► on the grid sparse receivers on Ocean bottom

Objective:

► Recover missing receivers from severe subsampling
► Improve the reconstruction quality & computational efficiency
Previous works

Deep-learning-based wavefield reconstruction:

- training only relies on acquired data
- implicit deep “factorization” via a nonlinear neural net
- recover randomly or periodically from missing receivers
- high missing rates (~90%)

Main contribution

Transfer learning to accelerate the wavefield reconstruction.
Seismic data in a 3D survey

Seismic data is 5D: \( (t, \text{Src} \ x, \text{Src} \ y, \text{Rec} \ x, \text{Rec} \ y) \)

Seismic data in a 3D survey

Seismic data is 5D:

\((t, \text{Src } x, \text{Src } y, \text{Rec } x, \text{Rec } y)\)

Taking Fourier transfer w.r.t. time:

\((\omega, \text{Src } x, \text{Src } y, \text{Rec } x, \text{Rec } y)\)
Seismic data in a 3D survey

Seismic data is 5D:

\[(t, \text{Src} x, \text{Src} y, \text{Rec} x, \text{Rec} y)\]

Taking Fourier transfer w.r.t. time:

\[(\omega, \text{Src} x, \text{Src} y, \text{Rec} x, \text{Rec} y)\]

Monochromatic seismic data is 4D:

\[(\text{Src} x, \text{Src} y, \text{Rec} x, \text{Rec} y)\]


Matricization of monochromatic seismic data

Our framework operates on *monochromatic* frequency slices

\((\text{Src } x, \text{ Src } y, \text{ Rec } x, \text{ Rec } y)\)
Matricization of monochromatic seismic data

Our framework operates on *monochromatic* frequency slices

\[(\text{Src } x, \text{Src } y, \text{Rec } x, \text{Rec } y)\]

Two choices for matricization:

- \((\text{Rec } x, \text{Rec } y) \times (\text{Src } x, \text{Src } y)\) domain (canonical)
- \((\text{Rec } y, \text{Src } y) \times (\text{Rec } x, \text{Src } x)\) domain (non-canonical)
Matricization of monochromatic seismic data

- (Rec x, Rec y) × (Src x, Src y) domain (canonical)
- (Rec y, Src y) × (Rec x, Src x) domain (non-canonical)

fully-sampled data
Matricization of monochromatic seismic data

- (Rec x, Rec y) × (Src x, Src y) domain (canonical)
- (Rec y, Src y) × (Rec x, Src x) domain (non-canonical)

observed data – sampling rate 10%
Matricization of monochromatic seismic data

- $(\text{Rec } x, \text{Rec } y) \times (\text{Src } x, \text{Src } y)$ domain (canonical)
- $(\text{Rec } y, \text{Src } y) \times (\text{Rec } x, \text{Src } x)$ domain (non-canonical)

Matricization of monochromatic seismic data

- (Rec x, Rec y) × (Src x, Src y) domain (canonical)
- (Rec y, Src y) × (Rec x, Src x) domain (non-canonical)

observed data – sampling rate 10%
Objective 1: Recover missing receivers

Objective 2: Speed up the reconstruction

Training framework: Generative Adversarial Network (GAN)

\[
\min_{\theta} \mathbb{E}_{X \sim p(X)} \left[ (1 - D_{\phi} (G_{\theta}(M \odot X)))^2 + \lambda \left\| G_{\theta}(M \odot X) - X \right\|_1 \right],
\]

\[
\min_{\phi} \mathbb{E}_{X \sim p(X)} \left[ (D_{\phi} (G_{\theta}(M \odot X)))^2 + (1 - D_{\phi}(X))^2 \right],
\]

\( X \) : input ground truth drawn from the probability distributions \( p(X) \)

\( G_{\theta} \) : generator \hspace{1cm} \( D_{\phi} \) : discriminator \hspace{1cm} \( M \) : training mask

\( \lambda \) : ensures that each realization of discriminator maps to a particular input, i.e., rather than solely fooling the discriminator.
Training framework: Generative Adversarial Network (GAN)

\[
\min_{\theta} \mathbb{E}_{X \sim p(X)} \left[ (1 - D_\phi (G_\theta (M \odot X)))^2 + \lambda \| G_\theta (M \odot X) - X \|_1 \right],
\]

\[
\min_{\phi} \mathbb{E}_{X \sim p(X)} \left[ (D_\phi (G_\theta (M \odot X)))^2 + (1 - D_\phi (X))^2 \right],
\]

Training pair: \((M \odot X, X)\)

\(X\): input ground truth drawn from the probability distributions \(p(X)\)

\(G_\theta\): generator \hspace{1cm} \(D_\phi\): discriminator \hspace{1cm} \(M\): training mask

\(\lambda\): ensures that each realization of discriminator maps to a particular input, i.e., rather than solely fooling the discriminator.
Extract training pairs for training

- \((\text{Rec x, Rec y}) \times (\text{Src x, Src y})\) domain
Extract training pairs for training

$\textbf{Extract training pairs for training}$

$(\text{Rec } x, \text{Rec } y) \times (\text{Src } x, \text{Src } y)$ domain

![Image of training pairs extraction process]

- Reshape
- $M \odot X$


Extract training pairs for training

- \((\text{Rec } x, \text{ Rec } y) \times (\text{Src } x, \text{ Src } y)\) domain

Training pair: \((M \odot X, X)\) $\rightarrow$ $G_\theta$ $\sqrt{\text{ }}$
Testing Stage: reconstruction

- \((\text{Rec } x, \text{ Rec } y) \times (\text{Src } x, \text{ Src } y)\) domain

source-receiver reciprocity

apply the trained neural network to all columns

\(G_\theta\)
The problem of the deep-learning-based method

Problem: each frequency slice is treated independently  ➔ high training costs
Transfer learning to speed up the reconstruction

**Problem:** each frequency slice is treated independently  

**Solution:** train from scratch

exploit frequency-to-frequency similarities
Transfer learning to speed up the reconstruction

**Problem:** each frequency slice is treated independently → high training costs

**Solution:**
- train from scratch
  - exploit frequency-to-frequency similarities
  - train from transferred weights from neighboring frequencies (do recursively for all frequencies from low to high)


Transfer learning to speed up the reconstruction

**Problem:** each frequency slice is treated independently → high training costs

**Solution:**
- train from scratch
- exploit frequency-to-frequency similarities
- train from transferred weights from neighboring frequencies (do recursively for all frequencies from low to high) → lower training costs


Numerical experiments

Numerically simulated data on 3D BG Compass model

- $172 \times 172$ 2D periodic source grid
- $172 \times 172$ 2D periodic receiver grid
- spatial subsampling in both horizontal directions
- complex velocity model w/ strong vertical & lateral variations
Numerical experiments

We know the missing pattern of receivers

Training mask:

► previous: training mask equal to the missing receiver sampling mask
Numerical experiments

We know the missing pattern of receivers

Training mask:

- **previous**: training mask equal to the missing receiver sampling mask
- **now**: change the training mask (with a same missing rate) at every epoch
Similarity: neighboring frequency slices > non-neighboring frequency slices
(14.33 Hz -> 14.66 Hz)                       (9.33 Hz -> 14.66 Hz)
Similarity: \[(14.33 \text{ Hz} \to 14.66 \text{ Hz}) > (9.33 \text{ Hz} \to 9.66 \text{ Hz})\]

\[\sim \text{six fold speedup}\]
Reconstruction performance

Frequency: 9.66 Hz
Neighboring: 9.33 Hz

Training from scratch: 300 epochs

Transfer learning from 9.33 Hz: 50 epochs
Reconstruction performance

Training from scratch: 300 epochs

Transfer learning from 9.33 Hz: 50 epochs

(Rec y, Src y) x (Rec x, Src x) domain (non-canonical)
Reconstruction performance

**Frequency:** 14.66 Hz  
**Neighboring:** 14.33 Hz

- **Transfer learning from 14.33 Hz:**
  - 50 epochs
  - SNR: 18.09 dB

- **Training from scratch:**
  - 300 epochs
  - SNR: 17.18 dB

**Transfer learning from 14.33 Hz:**
- 50 epochs
- SNR: 18.09 dB
Reconstruction performance

Training from scratch: **300 epochs**

Transfer learning from 14.33 Hz: **50 epochs**

- (Rec y, Src y) x (Rec x, Src x) domain (non-canonical)
Conclusions

The method does not need any external training data, assuming:

► desirable source sampling
► source-receiver reciprocity

Transfer learning (recursively for all frequencies) can significantly speed up in the training, specially at consecutive frequency slices with relatively high correlation.
Conclusions

The method does not need any external training data, assuming:

- desirable source sampling
- source-receiver reciprocity

Transfer learning (recursively for all frequencies) can significantly speed up in the training, specially at consecutive frequency slices with relatively high correlation.

Future work: improve the reconstruction accuracy of high-frequency slices
Thank you for your attention!

https://github.com/slimgroup/Software.SEG2020