Generative Adversarial Network for Seismic Data Interpolation

Introduction

During seismic data acquisition, because of natural and economic reasons, the sampling spacing along the offset sometimes cannot satisfy the spatial sampling requirement, which will result in spatial aliasing. The existence of spatial aliasing will affect the subsequent processing. The seismic data with spatial aliasing can be seen as regular missing data. An effective interpolation method for regular missing seismic data can reconstruct dense seismic data without spatial aliasing.

For regular missing seismic data, an anti-aliasing strategy in the interpolation should be included. The $f$-$x$ prediction method (Spitz, 1991) transforms seismic data from the time-space domain to the frequency-space domain and reconstructs the seismic data based on the predictability of the linear events in the space-frequency domain. This method should be implemented under a chosen local window to guarantee linearity, which can affect the final interpolation performance. Based on the wave equation, the reconstruction method is proposed (Ronen, 1987). However, the method requires the velocity model as the pre-information. With the development of machine learning, some interpolation methods based on machine learning are proposed (Jia and Ma, 2017; Wang et al., 2019).

In this abstract, we design a conditional generative adversarial network (cGAN) with Wasserstein distance for the anti-aliasing interpolation of seismic data. To train the network, we create one geological model to synthetize seismic data. After training, we design a new geological model to test the cGAN. We prove the reliability of the trained cGAN using the synthetic data and field seismic data. For comparison, we also process the regular missing data using the $f$-$x$ prediction method. The test results indicate that the cGAN can interpolate for seismic data and remove the spatial aliasing effectively.

A brief description for cGAN

A cGAN (Mirza and Osindero, 2014) is a deep learning model composed of two networks: a generator $G$ and a discriminator $D$. Its goal is to train a generator that can generate images that have the same statistics with the training dataset based on the given condition. Pix2Pix GAN (Isola et al., 2017) is proposed based on cGAN to transform one image from one domain to another domain. Figure 1 shows the brief structure of cGAN. In Figure 1, $G$ is the generator and $D$ is the discriminator. The input is the image in one domain and the target is the image in the other domain that the input will be transformed into. On the left side, the generator $G$ transforms the input to the output which will be the input of the discriminator $D$, and the ideal $D$ will label the output of the discriminator as fake. On the right side, the input together with the target will also be sent to the discriminator. However, the output will be labeled as real by the ideal discriminator. But the generator aims at generating output that will fool the discriminator and being labeled as real. So this will cause a conflict between the generator and the discriminator. After training the generator and discriminator will reach the Nash Equilibrium, where the discriminator cannot tell the generated images from the real images.

The generator in cGANs is a decoder-encoder network with a U-Net (Ronneberger et al., 2015) as shown in Figure 2a. And the structure of the discriminator is shown in Figure 2b. We use the Wasserstein loss (Arjovsky et al., 2017) to train the cGAN:

$$\min_{G} \max_{D \in \mathcal{D}_{1}} \mathcal{L}_{cGAN} = \mathbb{E}_{x,y \sim \mathcal{P}_{r}(x,y)}[D(x,y)] - \mathbb{E}_{x \sim \mathcal{P}_{r}(x)}[D(x,G(x))]$$  \hspace{1cm} (1)$$

Here, $D(x,y)$is the output of the discriminator for the input $x$ and the target $y$, and $G(x)$ is the output of the generator with input $x$. $D(x,G(x))$ is the output of the discriminator for the input $x$ and the output
of the generator $G(x)$. Besides the loss of cGANs, we also add the L1 distance to the loss function following Pix2Pix GAN:

$$\mathcal{L}_{L1} = \mathbb{E}_{x,y \sim P_{r}(x,y)}[\|y - G(x)\|_1]$$

Finally, the final objective can be expressed as:

$$\min_G \max_D \mathcal{L} = \arg\min_G \max_D \mathcal{L}_{cGAN} + \lambda \mathcal{L}_{L1}$$

Here, $\lambda$ is the weight of the loss of $L1$.

cGAN training

CGANs need a large amount of data to make them workable. To train the cGAN for interpolation application, we create two geological models, one of which is used to provide a training dataset and one for a testing dataset. The synthetic data is obtained using a finite-difference method based on the designed geological models. The geological model used to synthesize training dataset is shown in Figure 3a, which is one 2D in-line. For this model using a 30Hz Ricker wavelet, we get 73 shots × 401 traces × 1501 samples, and the source interval is 200m with a receiver interval of 12.5m. Figure 3b shows one shot from the model. For each shot, the seismic profile can be divided into small patches whose sizes are 128 × 128 along the time and offset axis. After dividing the data, the small patches are regarded as the target data, and the regular missing data is the input data. Regular missing data means that, for 128 traces, half of the traces are missing and the value of every other trace is set to 0.

In training, we use a root mean square prop (RMSprop) optimizer with a learning rate of 0.0002. And in Equation 3 the value of $\lambda$ is 200. During training we store the recovered signal to noise ratio (SNR), which can be used to evaluate the performance of the interpolated seismic data. The recovered SNR is described as:

$$\text{SNR}(dB) = 20 \log_{10} \frac{\|y\|_2}{\|y - G(x)\|_2}$$

The recovered SNRs during training is shown in Figure 4, from which we can see that the SNRs remain stable after about 350000 steps. Thus, the tested network is trained with a training step of 400000.

cGAN testing

For this model shown in Figure 5 to synthesize the testing dataset the receiver spacing is 12.5m which is
shown by the black dots and the source is shown by the red triangle. Based on a finite-difference method we get 321 traces using a Ricker wavelet with dominant frequency 50Hz. The time interval is 0.002s and there are 2001 samples per trace. The synthetic data is shown in Figure 6a. Figure 6b is the regularly sampled seismic data with 25m as the trace interval (from 321 traces to 161 traces), from which we can see the visual serration effects. Then we add zero trace between every two traces and interpolate it using the trained cGAN. The interpolated seismic data is shown in Figure 6c. From the interpolated result we can see that the visual serration effects are weakened. Figure 6d is the residual between Figure 6a and Figure 6c. The recovered SNR of Figure 6c is 26.64dB.

We also process the regularly sampled data based on the \( f-x \) prediction method and the interpolated result is shown in Figure 6e. Figure 6f shows the residual between Figure 6a and Figure 6e. We can see that the residual using the cGAN is less than that of the \( f-x \) prediction interpolation data from near offset to far offset. The recovered SNR of the \( f-x \) prediction method is 21.53dB which is smaller than the cGAN interpolation profile.

For a better comparison, the frequency-wavenumber (\( f-k \)) spectra is provided in Figure 7, in which the vertical axis denotes the frequency and the horizontal axis denotes the normalized wavenumber. Figure 7a is the \( f-k \) spectrum of the complete data, and after regularly sampling, the \( f-k \) spectrum is shown in Figure 7b, in which we can see spatial aliasing. Figure 7c, 7d represent the \( f-k \) spectra of the interpolated data based on the trained cGAN, and the \( f-x \) prediction method, respectively. From the \( f-k \) spectra, we can see that the trained cGAN can recover the missing data with a higher quality than the \( f-x \) method, for which some artifacts still exist. This also demonstrates the validation of the cGAN interpolation method.

To further prove the reliability of the trained cGAN, we use a field dataset to assess their flexibility. Figure 8a shows one of the shot gathers obtained by towed streamers, which has 239 traces and 1248 sampling points per trace with a time interval of 0.002s. The trace interval is 12.5m. After regularly sampling from 239 traces to 120 traces, the data is shown in Figure 8b with a trace interval of 25m. We reconstruct the regular missing data using the trained cGAN and the result is shown in Figure 8c.
The difference between the complete data and the interpolated data using cGAN is shown in Figure 8d. Figure 8b shows visual serration effects clearly, and after interpolation, the visual serration effect is weakened. In Figure 8d we see the residual between the complete data and the interpolated data, and the major residual is in the missing trace of the field data. Using Equation 4, the calculated recovered SNR is $25.43 \text{dB}$. Figures 9a, 9b, and 9c are the $f-k$ spectrum of the complete data, regular sampled data and the reconstructed data using the trained cGAN, respectively. Figure 9b shows spatial aliasing indicated by the arrows, and after interpolation using the trained cGAN, the spatial aliasing disappears. This proves the reliability of trained cGAN for interpolation applications.

**Figure 8:** The field seismic data for testing. (a) Complete data, (b) the regularly sampled data, (c) interpolated data using the trained cGAN, and (d) the residual.

**Figure 9:** The $f$-$k$ spectrum of the field test data. (a) The complete data, (b) the regular missing data, and (c) the interpolated data.

**Conclusions**

When sampling at offset is too coarse, spatial aliasing will appear. Our work achieves spatial aliasing free data reconstruction based on cGAN. We synthesized training dataset using one designed geological model. After training, we proved the reliability of the trained cGAN using a test dataset synthesized by a re-designed geological model and field seismic data. The interpolation results and the calculated recovered SNR demonstrated the validity and flexibility of the trained cGAN. We analyzed the interpolation results using the provided $f$-$k$ spectrum, and further demonstrated the performance of the method. Compared to the conditional method, the cGAN interpolation method doesn’t need different parameter selections for different data sets. The cGAN strategy is an efficient way as the main cost comes from the training and after that the generation cost is neglectable.

**Acknowledgements**

This work is supported by the Edinburgh Anisotropy Project (EAP) of the British Geological Survey, and is published with the permission of the Executive Director of the British Geological Survey (NERC) and the EAP sponsors.

**References**