Convolutional neural network-based reverse time migration with multiple energy

Shang Huang and Daniel O. Trad

ABSTRACT

Reverse time migration (RTM) with multiples has the advantage that it can handle steeply dipping structures and offer high-resolution images of the complex subsurface. However, there are some limitations on the initial model chosen, aperture illumination and computation efficiency. Reverse time migration has the dependency on reliable initial velocity models. If the input background velocity model is smoothed or not accurate, the RTM result image will have poor performance. RTM with multiple energy (RTMM) can help improve the illumination but will generate crosstalks because of the interference between different orders of multiples. One solution is to apply least-squares reverse time migration, which updates the reflectivity and suppresses artifacts through iterations. However, the accuracy still depends heavily on the input and the computation time is costly. We proposed a method based on a convolutional neural network (CNN) that behaves like the filter in LSRTM but is more efficient. This approach can learn patterns or features from geological structures and predict reflectivity by some smoothed velocity models through a modified residual U-Net, trained to improve the quality of RTM images. Numerical experiments show that RTMM-CNN can recover major structures and thin layers with higher resolution and improved accuracy compared with RTM-CNN method.

INTRODUCTION

Reverse time migration (RTM) (Baysal et al., 1983; Whitmore, 1983; McMechan, 1983; Levin, 1984) can handle steep geologic structure flanks and lateral velocity variations. However, it suffers from the coherent or incoherent artifacts in diving waves and backscattered waves, and low resolution for deep structure when given fixed or insufficient source-receiver coordination. Lu et al. (2020) and Torres and Sacchi (2021) proposed to use convolutional neural networks (CNNs) (LeCun et al., 1998; Krizhevsky et al., 2012; LeCun et al., 2015) on dip-angle domain elastic reverse time migration and single-step least-squares reverse time migration respectively to solve that issue. The methods worked for mitigating some artifacts and improving the resolution. On the other hand, reverse time migration is aperture limited because the major extrapolated energy is from the primary reflection which reflects the subsurface boundaries once. Multiple migration used in the RTM (RTMM) (Liu et al., 2011; Li et al., 2017; Wang et al., 2017; Zhang et al., 2020) can help to broaden the subsurface illumination and improve the accuracy and resolution.

Another concern is that reflectivity prediction derived from the reverse time migration depends heavily on the good initial velocity model. With an accurate migration velocity, adequate data and regularization terms will compensate for the illumination, whereas a large bias in the initial velocity will lead to inaccurate RTM results. One solution to suppress the artifact is applying least-squares reverse time migration (LSRTM) (Dong et al., 2012) which uses the RTM as the forward modeling and inverse engine to minimize amplitude differences between observed data and predicted data, and updates reflectivity coefficient

iteratively. The other way is using a CNN machine learning model mentioned above to learn the geology structure pattern between RTM images, background velocities and true reflectivity coefficients.

Exploiting the two facts that convolutional neural network (CNN) can learn the lithologic structure from different feature maps, and multiple energy can enhance the imaging bandwidth, we propose a CNN-based RTM with multiple reflection energy method (RTMM-CNN). This approach uses a modified U-Net, which first came up by Ronneberger et al. (2015), acting as a filter to learn the reflection boundaries from the RTM results with multiple energy. The baseline model is the U-Net-based RTM image without adding multiple energy (RTM-CNN). Results show that the proposed method can obtain the reflectivity prediction with extended illumination, refined structural boundaries, high accuracy and improved resolution.

THEORY

In this section, a modified U-Net-based reverse time migration with multiple energy (RTMM-CNN) will be delineated in detail. The workflow is shown in Figure 1. For a simple introduction, the input includes RTMM initial image and background velocity. Training through the modified U-Net, we can obtain a predicted reflectivity model as the outcome.

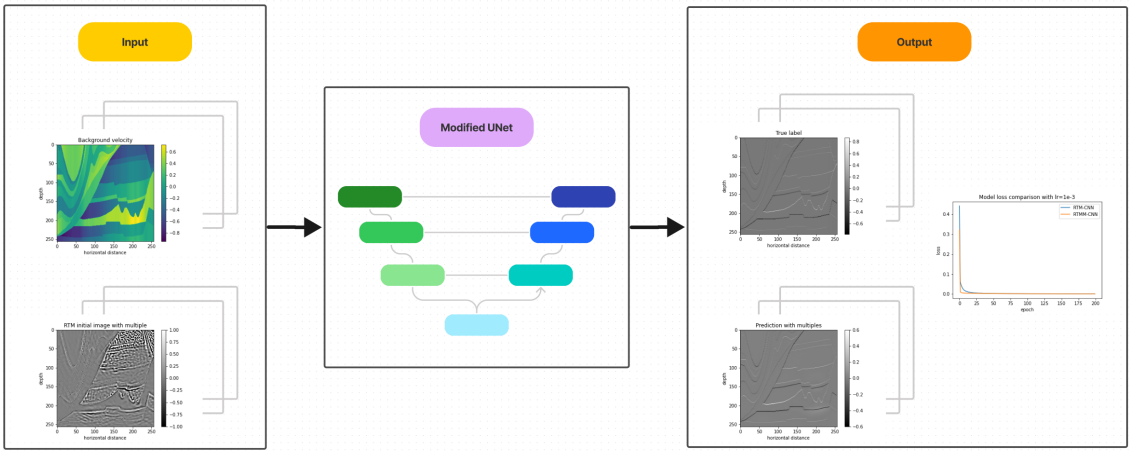


FIG. 1: Workflow for a modified U-Net based RTM with multiple reflections.

RTM with surface multiple

Based on the modified RTM scheme with multiple reflections (Liu et al., 2011), we only use the primary and first-order multiple reflections. The first step is to generate the primary wavefield and then use it as the virtual source to determine the first-order surface multiples, and apply the imaging condition between primaries and surface multiples.

The explosive sources located on the surface are simulated, and it generates wavefields downgoing to the subsurface. The subsurface structure response in the frequency domain can be expressed as:

$$\mathrm{U}(z_0, z_0) = \mathrm{XS}(z_0, z_0), \tag{1}$$

$$\mathrm{D}(z_0, z_0) = -\mathrm{U}(z_0, z_0), \tag{2}$$

where $\mathrm{S}(z_0, z_0)$ is the source, $\mathrm{U}(z_0, z_0)$ denotes the shot record or the observed data including primary and multiple reflections from the subsurface media, X represents the media response matrix, and $\mathrm{D}(z_0, z_0)$ means the observed data after free surface reflection.

Then, to generate the first-order multiples, the observed seismic data $\mathrm{U}(z_0, z_0)$ is downgoing forwarded as the virtual source, and we can obtain:

$$\mathrm{M}(z_0, z_0) = -\mathrm{XD}(z_0, z_0), \tag{3}$$

where $\mathrm{M}(z_0, z_0)$ represents the first-order multiple. This method is data-driven because recorded data $\mathrm{U}(z_0, z_0)$ is used as the virtual source to determine the first-order surface multiple, which means there is no need to estimate source wavelet again in this process.

For applying the imaging condition of RTM with multiples, traditionally, the source and receiver wavefields are cross-correlated to migrate the subsurface structure. In this method, the source wavefield can be replaced by the forward extrapolated observed data $\mathrm{D}(z_0, z_0)$, and the receiver wavefield can be given by the reversely extrapolated multiples $\mathrm{M}(z_0, z_0)$. Thus, the multiple imaging condition can be expressed as the cross-correlation between $\mathrm{D}(z_0, z_0)$ and $\mathrm{M}(z_0, z_0)$:

$$\mathrm{I}(x, z) = \sum_{t=1}^{t_{max}} \mathrm{D}(x, y, z) * M_B(x, z, t), \tag{4}$$

$$\mathrm{D}(x, y, z) = \mathrm{D}_P(x, y, z) + \mathrm{D}_M(x, y, z), \tag{5}$$

where the forward extrapolated total observed data $\mathrm{D}(z_0, z_0)$ is composed of primaries $\mathrm{D}_P(z_0, z_0)$ and multiples $\mathrm{D}_M(z_0, z_0)$, $\mathrm{M}_B(z_0, z_0)$ denotes the reversely downward first-order surface multiples, and t_{max} means the total recording time.

A modified U-Net based RTM with surface multiple

U-Net (Ronneberger et al., 2015) has efficient segmentation performance replying to data augmentation with annotated samples. In this proposed method, we developed a modified U-Net (Figure 2) with more multilayer convolutional blocks and skip connections to learn from residuals and patterns for the regression task. For the encoder part, the network down-sampling our sample data into small sizes for learning key features of different reflectors from RTMM images, background velocities and true reflectivity labels. Then, the subsurface structure key features are up-sampled to the original dimensions by transpose convolutions. Additional skip connections help to strengthen the training result with weak constraints.

The network operator acts similar as the least-squares reverse time migration. For LSRTM, the solution is derived from the minimum difference between true and migrated images,

$$\mathbf{m}^* = \arg\min_m \{\frac{1}{2}||\mathbf{\Gamma m} - \mathbf{m}_{mig}||_2^2\}. \tag{6}$$

A formal solution to equation 6 is

$$\mathbf{m}^* = \mathbf{\Gamma}^{-1}\mathbf{m}_{mig} = \mathbf{\Gamma}^{-1}(\mathbf{L}^{\mathrm{T}}\mathbf{d}), \tag{7}$$

where $\mathbf{\Gamma}^{-1}$ is the inverse Hessian, $\mathbf{L}^{\mathrm{T}}$ is the adjoint operator and $\mathbf{d}$ represents the observed seismic data.

Similarly, this modified U-Net can be used as an alternative way of inverse Hessian to determine the imaging result. The benefit is that there is no need to compute the expensive inverse Hessian operator or process the shot records. The forward modeling or feedforward procedure in our proposed method for a multilayer CNN is $\mathbf{\Gamma}_{unet}$, and the solution can be determined as:

$$\mathbf{m}_{pred} = \mathbf{\Gamma}_{unet}(\mathbf{m}_{rtmm}, \mathbf{m}_{vel}), \tag{8}$$

where $\mathbf{m}_{rtmm}$ is the RTMM initial image, $\mathbf{m}_{vel}$ denotes the background velocity model and $\mathbf{m}_{pred}$ represents the output reflectivity coefficient prediction.

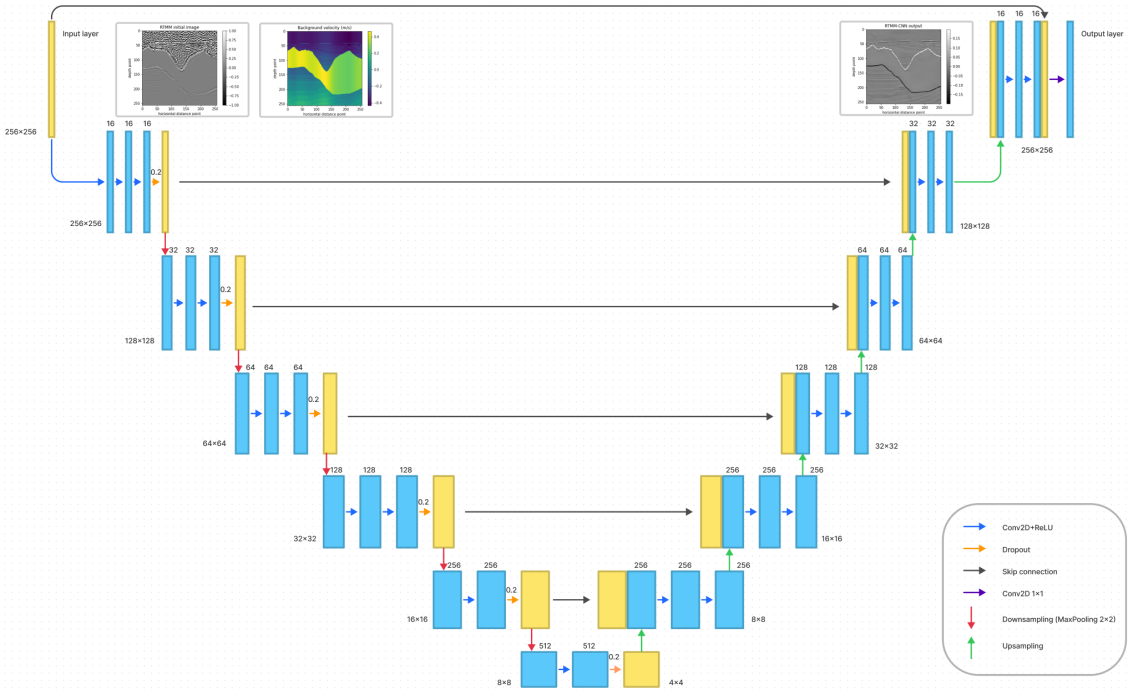


FIG. 2: Detailed workflow of the modified U-net architecture. Each blue box represents a multi-channel feature. The yellow boxes stand for the concatenated copied feature from the encoder part. The arrows between boxes correspond to the different operations as shown in the right legend. The number of channels is located on top of the box and the image dimensionality is denoted on the left edge.

The mean squared error (MSE) loss is applied to evaluate the model performance and penalize the large prediction errors:

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{m}^i_{pred} - \mathbf{m}^i_{true})^2, \tag{9}$$

where $\mathbf{m}_{pred}$ is derived from equation 8 U-Net using RTMM initial images, and $\mathbf{m}_{true}$ denotes the true reflectivity models.

NUMERICAL EXAMPLES

In this section, some numerical examples will be shown to illustrate the approach and the refined performance on reflectivity prediction with this proposed method.

Train and test set

We've used Sigsbee2b, Amoco, Agbami, Pluto, BP2004 and Marmousi as the origin input set. Eight meters is used as the spatial interval for each grid point. The total record time is 7.2 seconds with 0.8 milliseconds temporal sampling. The shot and receiver intervals are 80 and 16 meters respectively. A fourth-order finite different method is used for the forward modeling. RTM-CNN is chosen as our baseline model. Before training, the whole RTM and RTMM images are partly chosen and divided randomly into 2100 spatial windows with 256x256 points. The ratio of train and test set is 0.8: 0.2. Note that we have not used all the pixels in the migration images, and the new windowing images will be treated as the validation set. All the output predictions have normalized scaling.

Example 1: Pluto model

The example shown in Figure 4 is extracted from the Pluto model (Figure 3). The original size is 1234x401 points. 122 shots and 615 receivers are located at the same depth layer 40 meters below the surface. True velocity model and ground truth label are shown in Figure 4a and b respectively. As for the second channel network input, RTMM initial image (Figure 4d) has a higher amplitude and resolution of the salt baseline as well as the beneath thin layer structures than those in the RTM image (Figure 4c). After using the proposed method, compared with RTM-CNN output (Figure 4e), RTMM-CNN (Figure 4f) can predict the reflectivity with improved quality.

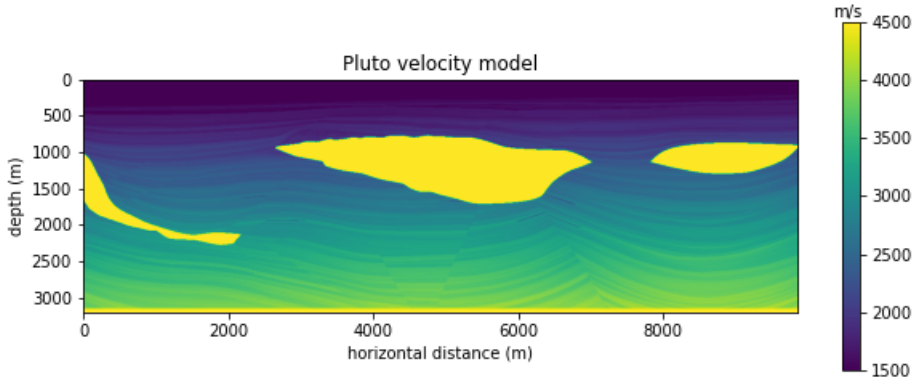
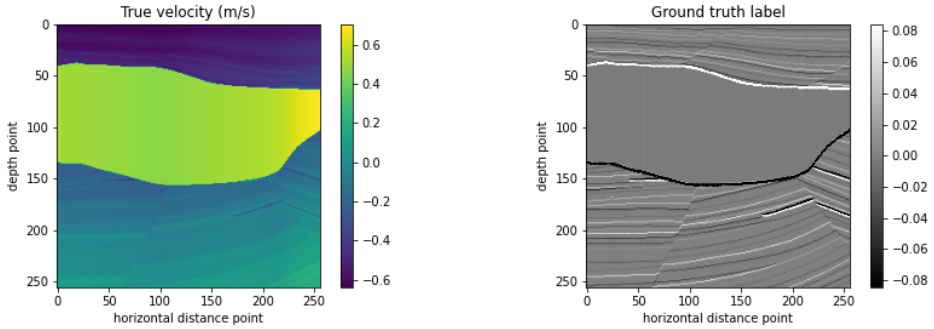
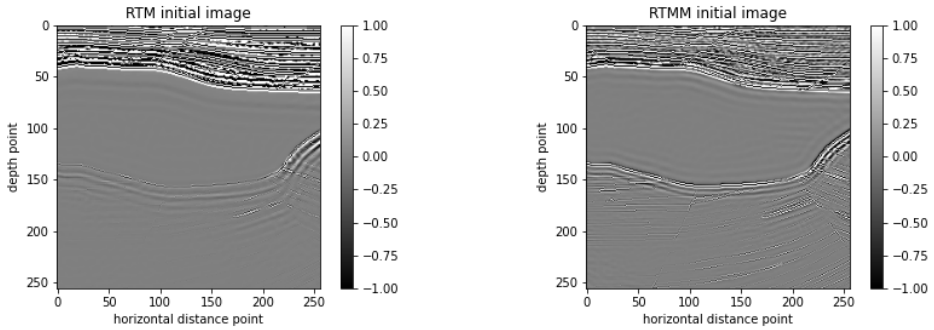


FIG. 3: Pluto velocity model.



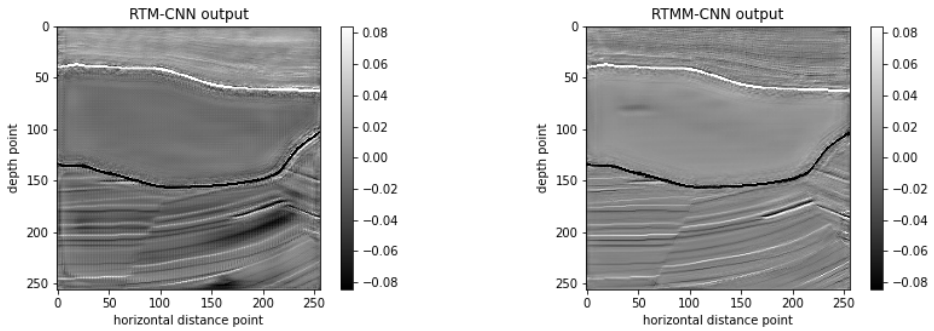
(a)

(b)



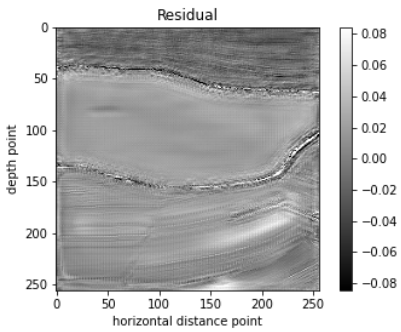
(c)

(d)



(e)

(f)



(g)

FIG. 4: (a)True velocity model (b) ground truth reflectivity coefficient label (c) RTM images (d) RTMM images (e) RTM-CNN prediction (f) RTMM-CNN prediction and (g) the residual between (f) and (e).

For example, through the middle and left parts extracted from the windowed structure (Figure 5), the small faults can be seen clearly in the RTMM-CNN outcomes (Figure 5c and f) with large amplitude and accurate location, whereas the corresponding structure in the RTM-CNN results (Figure 5b and e) have a low resolution with artifacts which are hard to recognize. The residual between RTMM-CNN and RTM-CNN results shown in Figure 4g can tell more differences in detail. For instance, the thin layers both above and below the salt body are recovered thoughtfully in RTMM-CNN.

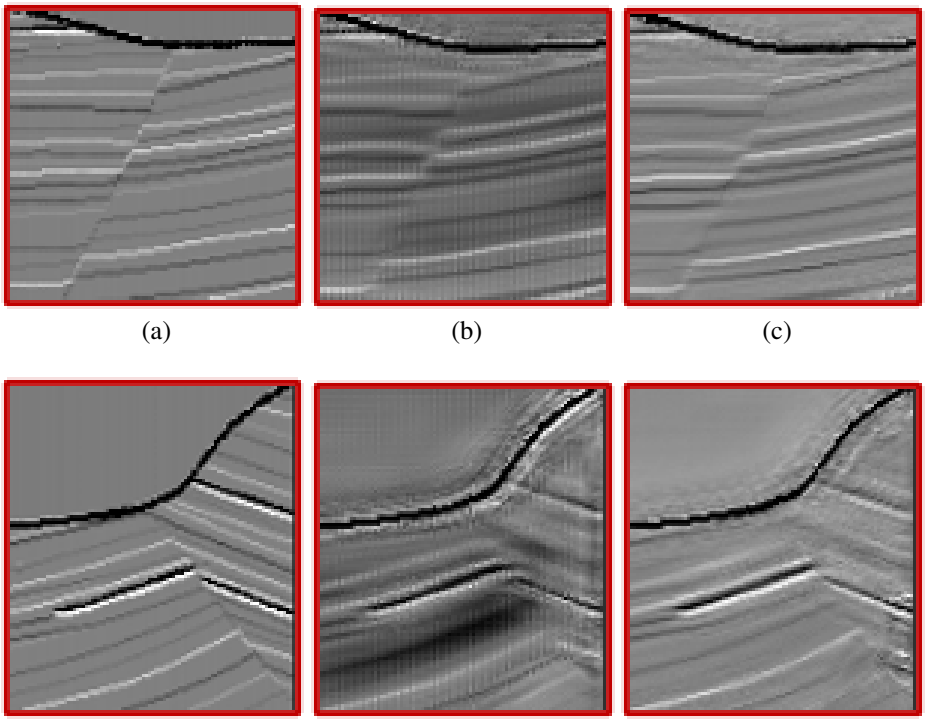


FIG. 5: Enlarged regions from the outputs in Figure 4. (a) and (d) give the true label; RTM-CNN outputs are shown in (b) and (e), and corresponded RTMM-CNN predictions are in (c) and (f).

Example 2: Marmousi model

This test set is part of the unused Marmousi model (Figure 6), with 1942x400 points with 193 shots and 970 receivers located 40 meters depth. True and smoothed velocity models are shown in Figure 7a and b separately.

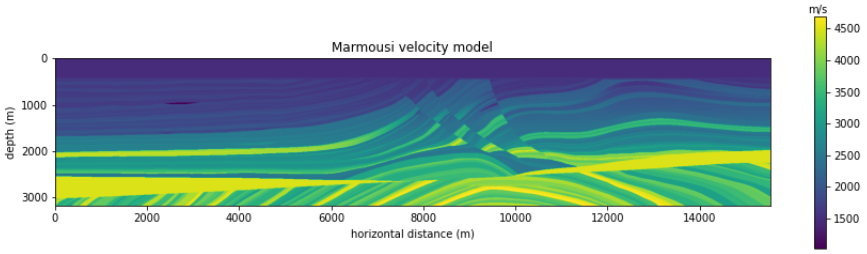


FIG. 6: Marmousi velocity model.

After using RTMM-CNN (Figure 7f), the result can recover more thin layer structures at the shallow part and migrate them to the accurate location compared with RTM-CNN output (Figure 7e). Precisely, RTMM-CNN can predict the thin layer beneath 100 depth points (Figure 8c), whereas the structure in RTM-CNN (Figure 8b) is unclear and vague. For the deep structure below the wedge reflector, Figure 8f can also output the prediction with high resolution and accuracy. The residual in Figure 7h proves that RTMM-CNN provides more information about thin layers, and dipping events below the wedge.

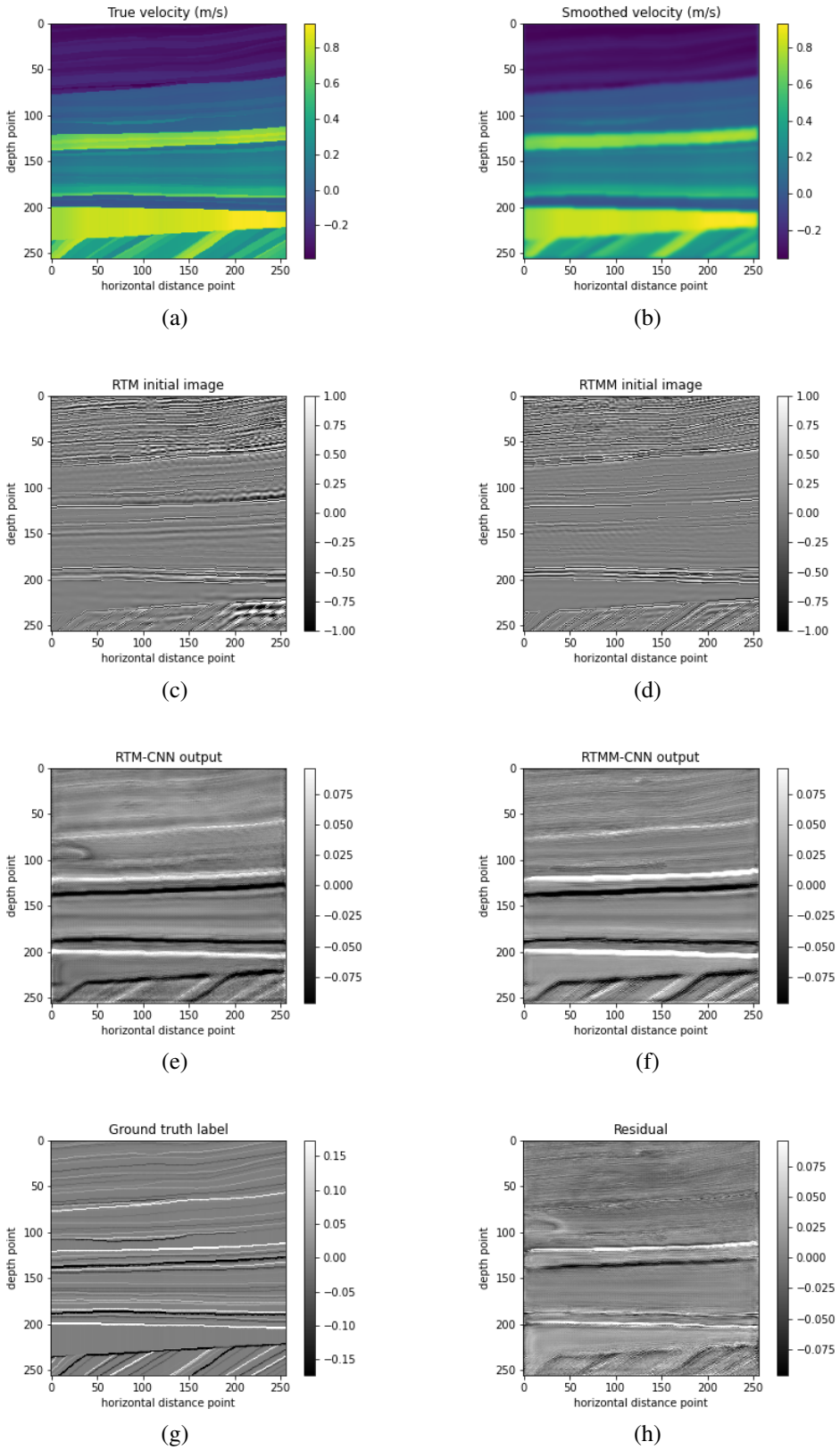


FIG. 7: (a)True velocity model (b) smoothed velocity model (c) RTM images (d) RTMM images (e) RTM-CNN prediction (f) RTMM-CNN prediction and (g) ground truth reflectivity coefficient label and (h) the residual between (f) and (e).

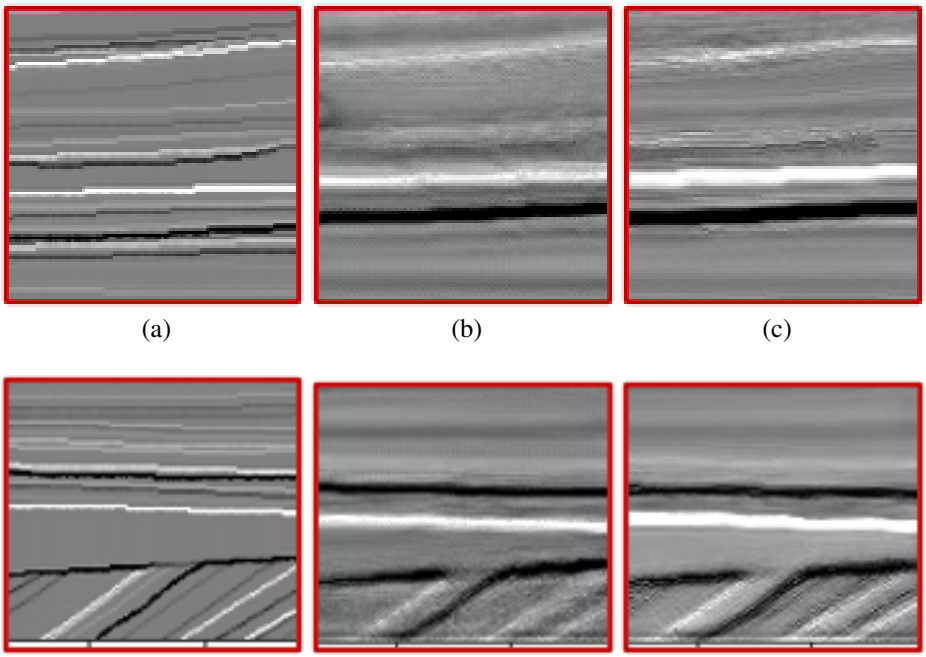


FIG. 8: Enlarged regions from the Marmousi model. (a) and (d) give the true label; RTM-CNN outputs are shown in (b) and (e), and corresponded RTMM-CNN predictions are in (c) and (f).

Example 3: Foothill model

To avoid the overfitting and fixed learning pattern issue, we decided to test our proposed model on a different model, the Foothill model, which has never participated in neither training nor testing process. The true velocity model is shown in Figure 9, and 300 windows are chosen to test our model performance. The total model size is 1600x1000, with 10 meters spatial interval. We have used 99 shots and 798 receivers which locating at the near surface. The total recorded time is the same as in previous examples. Two situations are tested in this section: one is using the accurate velocity model as the input, the other is applying a smoothed velocity model to test if our model can handle an inaccurate initial input.

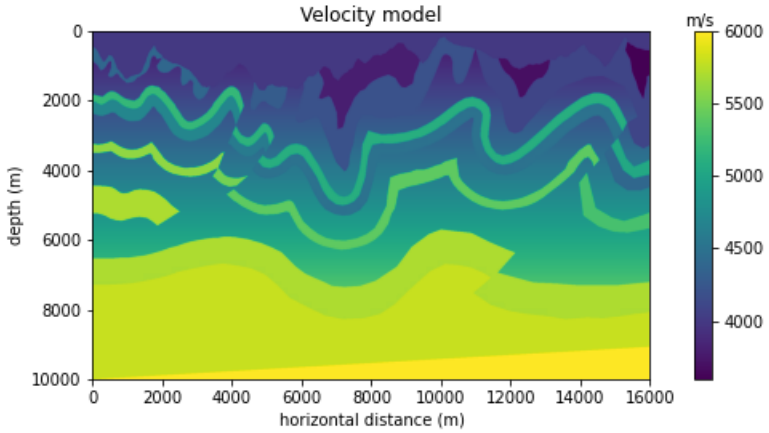


FIG. 9: Tested Foothill velocity model.

For the first situation, one windowed result is shown in Figure 10. RTMM initial image has higher amplitudes of curvature boundaries than the RTM image. Correspondingly, the RTMM-CNN result provides improved resolution with fewer artifacts compared with the RTM-CNN outcome. For instance, Figure 11 shows the detailed comparison extracted from the left and upper-right parts of Figure 10. RTMM-CNN outputs (Figure 11c and f) can predict either dipping events or curvature boundaries with sharpening resolution. Both Figure 10 and 11 can prove that RTMM-CNN results have a better fit and are closer to the true reflectivity labels compared with the RTM-CNN when given an unseen velocity input.

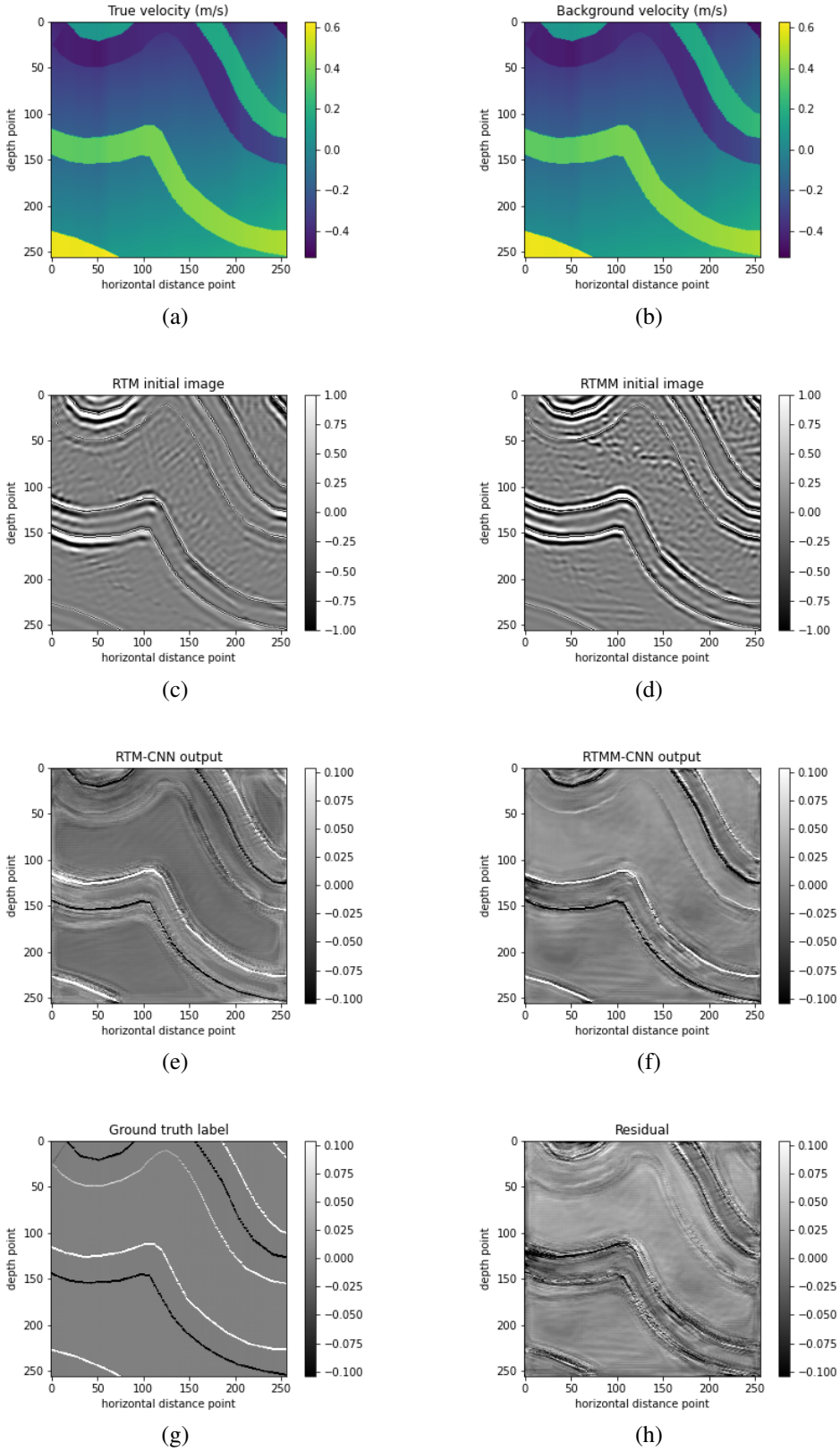


FIG. 10: (a)True velocity model (b) input smoothed velocity (c) RTM images (d) RTMM images (e) RTM-CNN prediction (f) RTMM-CNN prediction (g) ground truth reflectivity coefficient label and (h) the residual between (f) and (e).

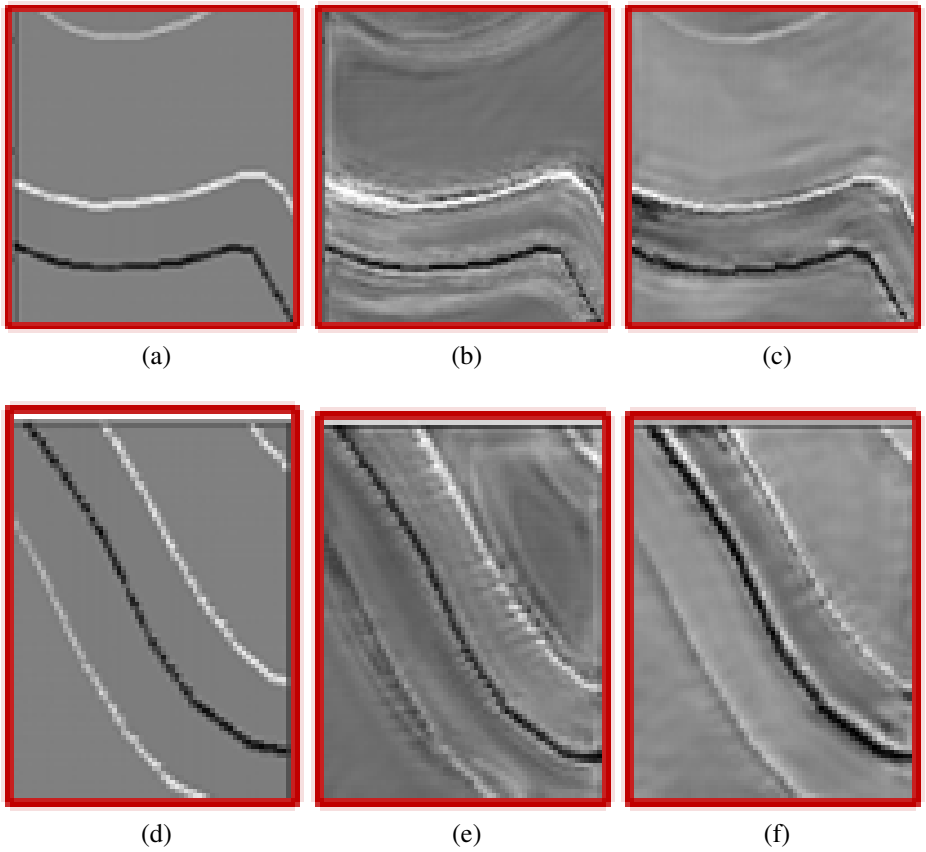
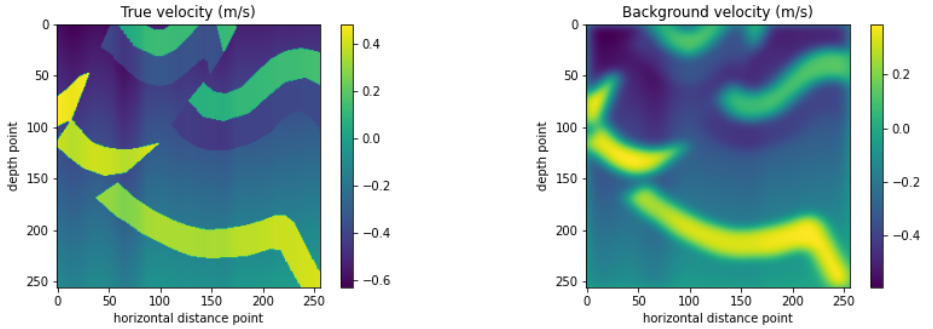


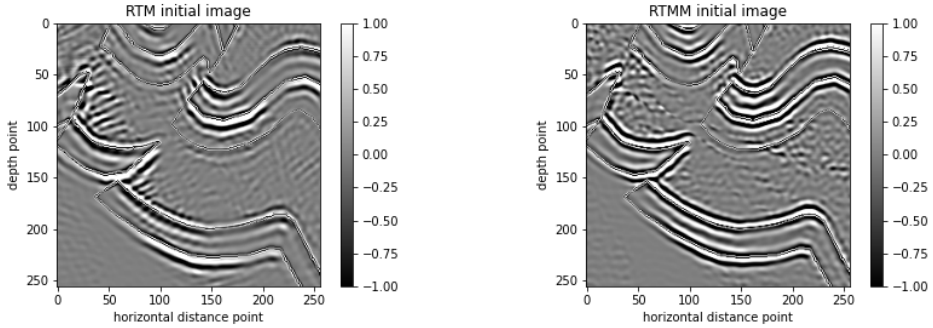
FIG. 11: Enlarged regions from the Foothill model given the accurate velocity model as the input. (a) and (d) give the true label; RTM-CNN outputs are shown in (b) and (e), and corresponded RTMM-CNN predictions are in (c) and (f).

The second situation is described in Figure 12, where the referred true velocity and actual input smoothed velocity are shown in Figure 12a and b respectively. Except for applying the smoothed velocity model, the other input channel in the RTM-CNN is the RTM initial image (Figure 12c). On the other hand, the proposed method RTMM-CNN has corresponded RTMM initial image (Figure 12d) as the input. The latter has significant amplitudes for some curvatures. For example, upper-top curvature boundaries can be seen more clearly with higher resolution in Figure 13c rather than Figure 13b. As for the shadow region (Figure 13d), RTMM-CNN (Figure 13f) would give the faded prediction but still with a high resolution for the above structure compared with Figure 13e. Furthermore, the tilted fault cross this windowed model can be recovered properly in the RTMM-CNN output as well. From the residual (Figure 12h), we can see RTMM-CNN provides more details on each reflector boundary which is more reliable on subsurface structure interpretation compared with the RTM-CNN result.



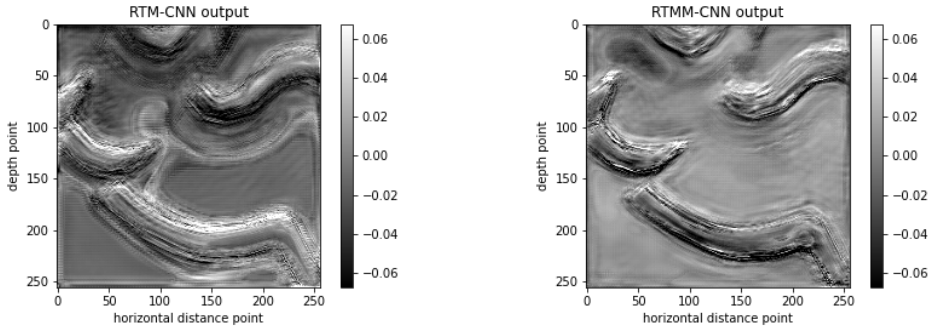
(a)

(b)



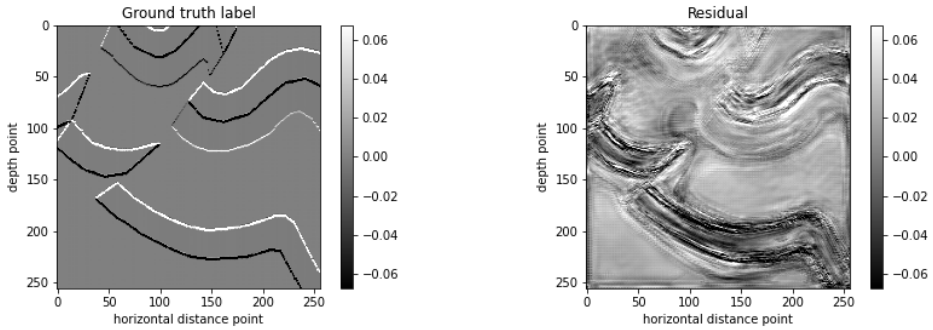
(c)

(d)



(e)

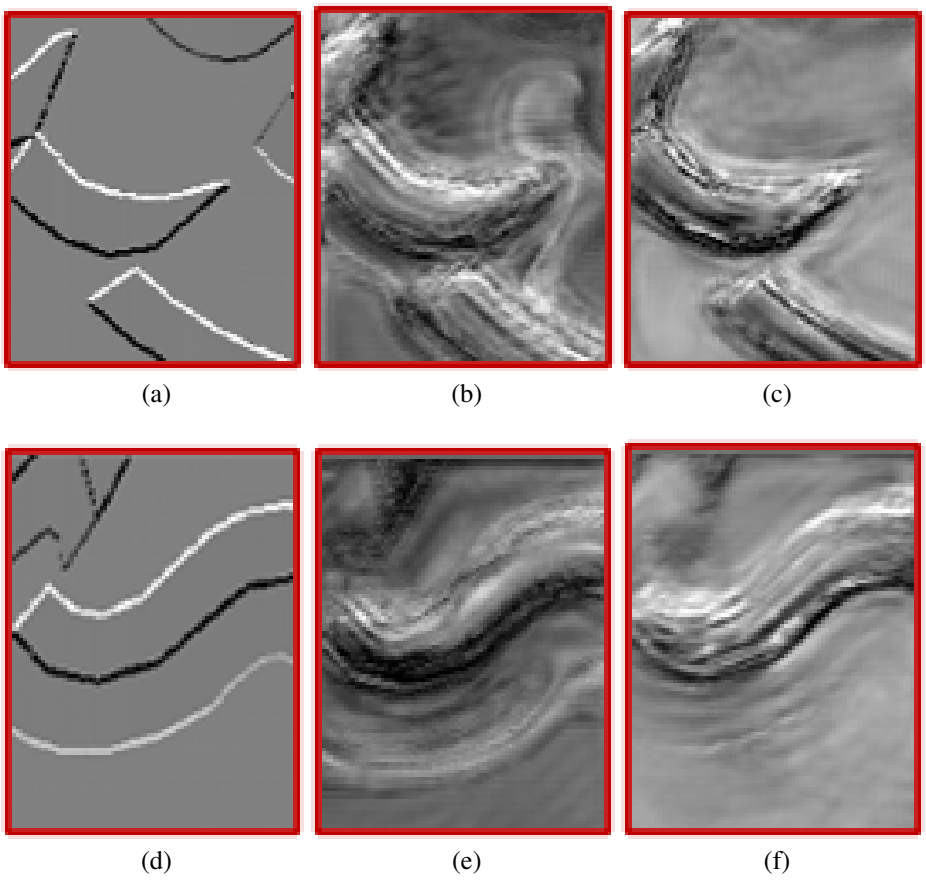
(f)



(g)

(h)

FIG. 12: (a)True velocity model (b) input smoothed velocity (c) RTM images (d) RTMM images (e) RTM-CNN prediction (f) RTMM-CNN prediction (g) ground truth reflectivity coefficient label and (h) the residual between (f) and (e).



(a) (b) (c)

(d) (e) (f)

FIG. 13: Enlarged regions from the Foothill model given the smoothed velocity model as the input. (a) and (d) give the true label; RTM-CNN outputs are shown in (b) and (e), and corresponded RTMM-CNN predictions are in (c) and (f).

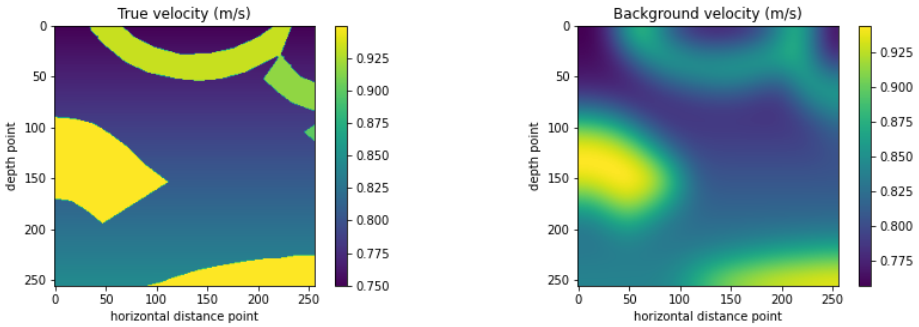
Consideration - model dependence on the input background velocity model

After analyzing the numerical examples shown above, another situation is considered if this proposed model depends heavily on the input background velocity model. In other words, the outcome has not been decided if it is a velocity-driven result so far. To find out the fact, we deploy two more tests in this section. One is to apply a larger gaussian smooth filter with $\sigma_x = 10$ and $\sigma_y = 15$ than the previous examples ($\sigma_x = 5$ and $\sigma_y = 7$); the other is to remove the input velocity completely.

As the amplitude might be difficult to recognize by eye, the peak signal-to-noise ratio (PSNR) is used to evaluate the model performance.

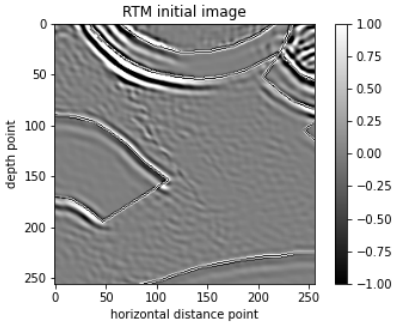
$$\text{PSNR} = 20 * \log_{10}(\frac{\text{MAX}_I}{\sqrt{\text{MSE}}}), \tag{10}$$

where MAX_I denotes the maximum possible pixel value of the image, and MSE is the mean squared error based on the equation 9.

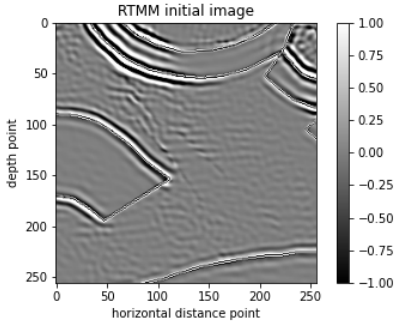


(a)

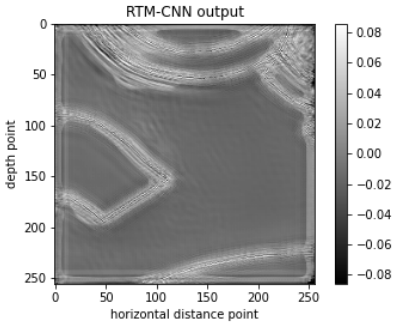
(b)



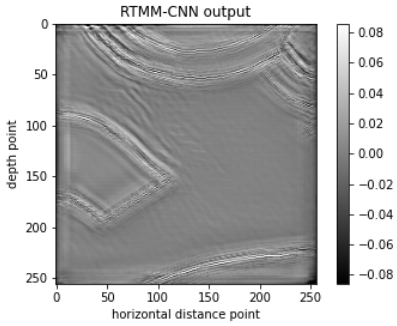
(c)



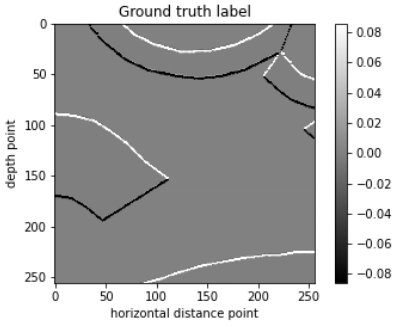
(d)



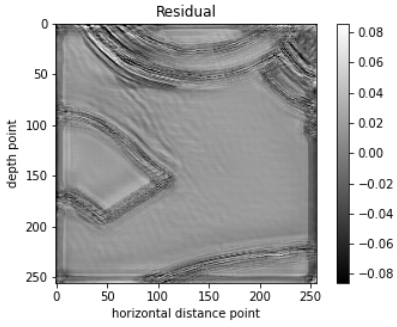
(e)



(f)



(g)



(h)

FIG. 14: Test 1 on the smoothed velocity input. (a) True velocity (b) Input smoothed velocity (c) RTM images (d) RTMM images (e) RTM-CNN prediction (f) RTMM-CNN prediction (g) ground truth reflectivity coefficient label and (h) the residual between (f) and (e).

Figure 14 shows the results using a large smooth filter on the background velocity input. Compared with the RTM-CNN result (Figure 14e), RTMM-CNN (Figure 14f) can sharpen the syncline structure boundaries on the top and anomalies at the bottom as well. For the second test without the velocity input, the RTMM-CNN (Figure 15b) can also behave better than the RTM-CNN (Figure 15a) in the image resolution improvement and artifact suppression.

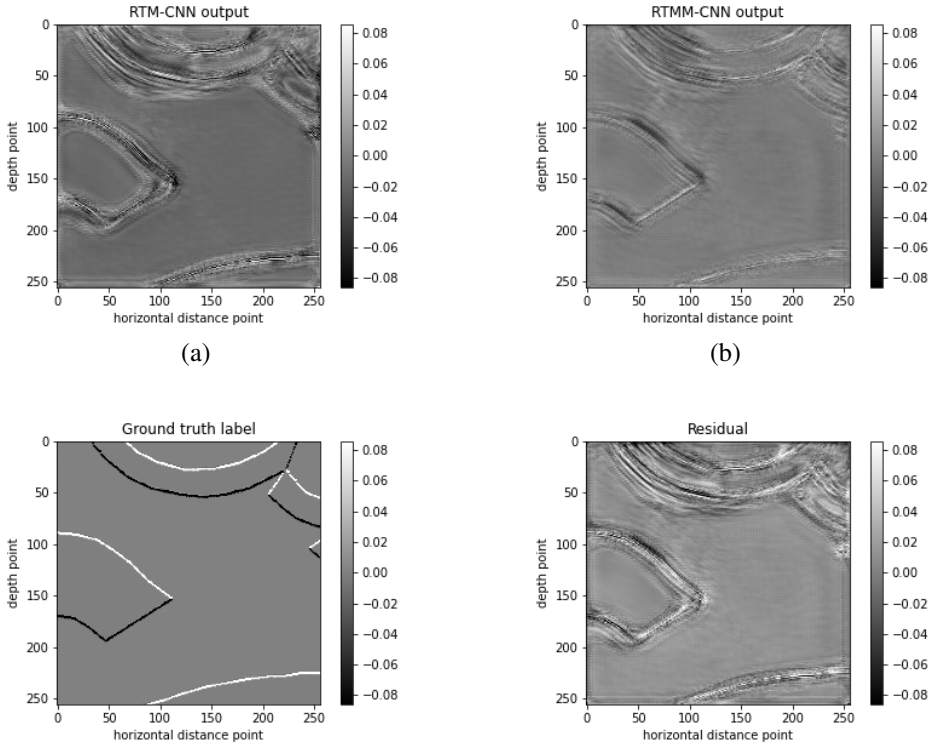


FIG. 15: Test 2 when removing the background velocity input. (a) RTM-CNN prediction (b) RTMM-CNN prediction (c) ground truth reflectivity coefficient label and (d) the residual between (b) and (a).

Precisely, the enlarged region comparison (Figure 16) and corresponding table (Table 1) can give a detailed illustration of the results. The right column (Figure 16b and d) derived from RTMM-CNN shows the predictions with less noise and more accurate location. Comparing the two tests in Table 1, predictions with background velocity have higher PSNR than removing it, where RTMM-CNN has a 0.2% increase and RTM-CNN gain a 0.6% rise. Relatively, the image resolution is better as well when using a velocity input.

Table 1: PSNR comparison in the two tests.

	PSNR (RTM-CNN)	PSNR (RTMM-CNN)
Using a smoothed velocity input	24.68	25.05
Removing the velocity input	24.54	25

The tests help us understand the background velocity can provide more information

about the subsurface structure, but either RTM-CNN or RTMM-CNN does not heavily depend on the velocity input. One reason is that the input velocity has a large gaussian smooth filter applied before training, which is not necessarily called a "good initial model". This proves our assumption that our proposed model can handle input velocities with bias. The other reason is that even though the velocity has been removed from inputs, the results can still learn features from the RTM and RTMM initial images. Though the amplitude is not as high as the example with velocity input, we can derive another step for future work: RTM or RTMM initial images can be improved by adding sources or receivers. In that case, the trained model will be better to handle true boundaries and artifact amplitudes.

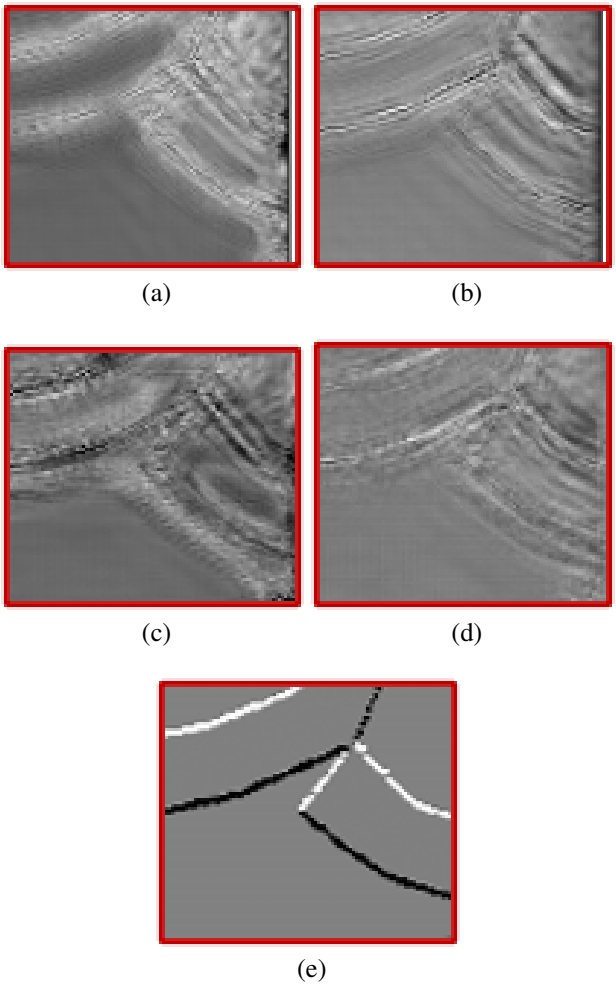


FIG. 16: Enlarged region comparison. (a) and (b) are outcomes from RTM-CNN and RTMM-cnn when given a smoothed velocity input. (c) and (d) are RTM-CNN and RTMM-CNN results when no background velocity model is provided. (e) denotes the true label.

Model performance and computation time

Figure 17 illustrates the model loss comparison between RTM-CNN and RTMM-CNN after 20 iterations when the learning rate is 0.001. RTMM-CNN (in orange line) can converge to a smaller loss value, e.g 0.0005, compared with the RTM-CNN baseline model (in

blue line). Additionally, RTMM-CNN has a more stable convergence rate along with the iterations than the baseline model, whereas the latter shows a larger fluctuation change.

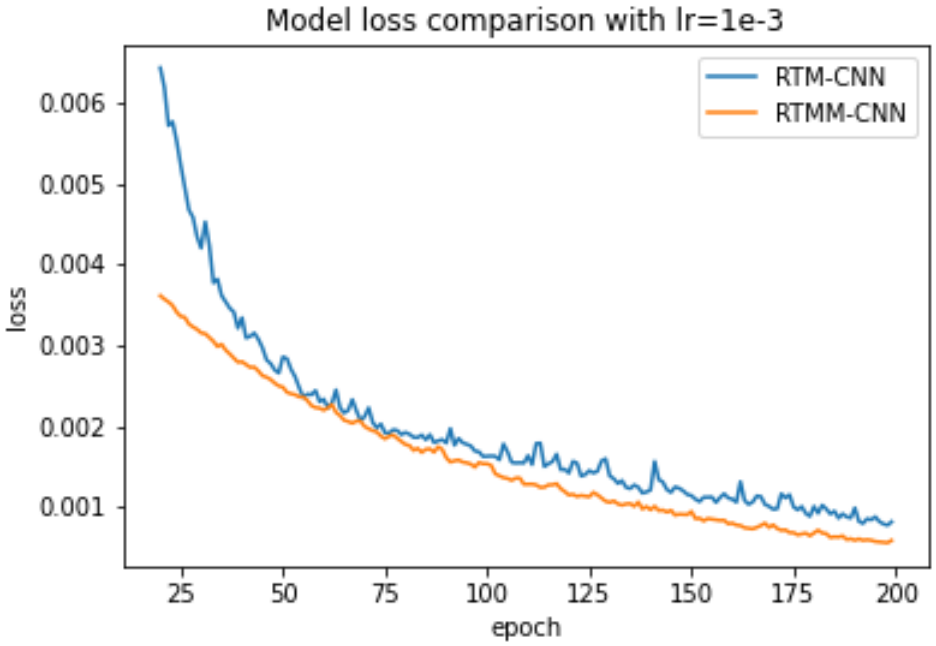


FIG. 17: Loss comparison between RTM-CNN and RTMM-CNN after 20 iterations.

When training the network, each iteration will take around 53 seconds, and the total 200 iterations take approximately 10600 seconds with Nvidia K80 / T4 16GB GPU and 25.46GB RAM. Since the method works similarly to the least-squares reverse time migration, we tested the total time running for one iteration least-squares reverse time migration (without GPU) on one model. The LSRTM needs 298 seconds for each shot to process the forward and reversely time backward wavefields imaging condition. Thus, the total computation time is around 29601 seconds for 99 shots.

CONCLUSIONS

Both RTM-CNN and RTMM-CNN can have some tolerance on the initial background velocity model. The proposed method RTMM-CNN can provide improved reflectivity coefficient prediction with a smoothed initial velocity input compared with the RTM-CNN output. It takes the benefit of multiple reflections that can widen the subsurface structure illumination. The neural network operator acts as the least-squares reverse time migration, which can suppress image artifacts and improve the reflector resolution. The next step is to let the model learn how to predict a steady reflectivity when given a more smoothed input and field data. Furthermore, we need to find a way to improve the model performance on the shadowed zone.

ACKNOWLEDGEMENTS

The authors would like to thank the sponsors of CREWES for continued support. This work was funded by CREWES industrial sponsors, China Scholarship Council (CSC) and

NSERC (Natural Science and Engineering Research Council of Canada) through the grants CRDPJ 543578-19.

REFERENCES

Baysal, E., Kosloff, D. D., and Sherwood, J. W., 1983, Reverse time migration: Geophysics, **48**, No. 11, 1514–1524.

Dong, S., Cai, J., Guo, M., Suh, S., Zhang, Z., Wang, B., and Li, e. Z., 2012, Least-squares reverse time migration: Towards true amplitude imaging and improving the resolution, *in* SEG technical program expanded abstracts 2012, Society of Exploration Geophysicists, 1–5.

Krizhevsky, A., Sutskever, I., and Hinton, G. E., 2012, Imagenet classification with deep convolutional neural networks: Advances in neural information processing systems, **25**, 1097–1105.

LeCun, Y., Bengio, Y., and Hinton, G., 2015, Deep learning: nature, **521**, No. 7553, 436–444.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P., 1998, Gradient-based learning applied to document recognition: Proceedings of the IEEE, **86**, No. 11, 2278–2324.

Levin, S. A., 1984, Principle of reverse-time migration: Geophysics, **49**, No. 5, 581–583.

Li, Z., Li, Z., Wang, P., and Zhang, M., 2017, Reverse time migration of multiples based on different-order multiple separation: Geophysics, **82**, No. 1, S19–S29.

Liu, Y., Chang, X., Jin, D., He, R., Sun, H., and Zheng, Y., 2011, Reverse time migration of multiples for subsalt imaging: Geophysics, **76**, No. 5, WB209–WB216.

Lu, Y., Sun, H., Wang, X., Liu, Q., and Zhang, H., 2020, Improving the image quality of elastic reverse-time migration in the dip-angle domain using deep learning: Geophysics, **85**, No. 5, S269–S283.

McMechan, G. A., 1983, Migration by extrapolation of time-dependent boundary values: Geophysical prospecting, **31**, No. 3, 413–420.

Ronneberger, O., Fischer, P., and Brox, T., 2015, U-net: Convolutional networks for biomedical image segmentation, *in* International Conference on Medical image computing and computer-assisted intervention, Springer, 234–241.

Torres, K., and Sacchi, M., 2021, Deep learning based least-squares reverse-time migration, *in* First International Meeting for Applied Geoscience & Energy, Society of Exploration Geophysicists, 2709–2713.

Wang, Y., Zheng, Y., Xue, Q., Chang, X., Fei, T. W., and Luo, Y., 2017, Reverse time migration of multiples: Reducing migration artifacts using the wavefield decomposition imaging condition: Geophysics, **82**, No. 4, S307–S314.

Whitmore, N. D., 1983, Iterative depth migration by backward time propagation, *in* SEG Technical Program Expanded Abstracts 1983, Society of Exploration Geophysicists, 382–385.

Zhang, Y., Liu, Y., Liu, X., and Zhou, X., 2020, Reverse time migration using water-bottom-related multiples: Geophysical Prospecting, **68**, No. 2, 446–465.