Deep learning for salt body detection applied to 3D Gulf of Mexico data

Benjamin Consolvo, Fairfield Geotechnologies, Ehsan Zabihi Naeini, Earth Science Analytics, Paul Docherty, Fairfield Geotechnologies*

Summary

Salt interpretation on seismic data has historically been a very manual process, requiring weeks or even months to complete on one 3D seismic survey. The accuracy of the interpreted salt boundary is critical for sub-salt imaging and subsequent drilling for oil and gas. The nature of the salt problem can be reduced to a binary classification problem that is well suited to modern machine learning (ML) algorithms: each location on an image either contains salt or sediment. Seismic surveys are collected and processed in different ways, which poses a challenge to traditional ML methods that rely on statistical similarity between training data and prediction data, especially where limited training data are available. We propose to use a supervised ML approach that treats each seismic survey independently. In particular, we show that an adaptive U-Net approach yields accurate salt bodies in minutes rather than weeks and requires minimal training data.

Introduction

Salt interpretation is important for velocity model building and seismic migration workflows (Wang et al., 2008). Many attempts to automate salt interpretation have been made because of the time-consuming nature of the task. Some of the recent attempts include attribute-based methods (Guillen et al., 2015; Wu, 2016; Shafiq et al., 2017; Wu et al., 2018), and convolutional neural networks or CNNs (Shi et al., 2018; Zeng et al., 2019; Sen et al., 2020).

CNNs have been widely used successfully for object classification and detection tasks for photographs (Russakovsky et al., 2015). However, seismic images are different from photographs in two fundamental aspects: 1) acquisition methods and 2) processing techniques applied to the data. The amplitude distribution, sampling intervals, frequency content, and pixel relationships are also fundamentally different between surveys, which presents a significant challenge to ML when attempting to use transfer learning. A common ML approach is to gather as much statistically varying data as possible for training to obtain a generalized ML model that would work on any survey. As noted by Sen et al. (2020), however, it is impossible to come up with an all-encompassing training set for all variations. Like many folks in the seismic industry, we had limited access to data, and so a different approach was required. We treated each survey as an independent ML project with its own training and prediction. The adaptive U-Net method we chose obtains high accuracy with a limited amount of manual picking.

We first explore CNN architectures and our adaptive U-Net approach, and second we show overall accuracy rates of above 98% as compared to manual picks on two 3D seismic surveys from the Gulf of Mexico. In this paper, all the work for training and prediction were completed on a single 16 GB Tesla V100 GPU.

CNN Architectures

CNNs are generally made up of convolutional layers, pooling layers, and dropout layers. The convolution step involves multiplying filters by small pixel regions in the image and summing up these multiplications. Initially, a CNN has no knowledge of the relationship between the pixels in the input image, and so the filters are random. However, during the learning process, the filters become representative of abstracted relationships between pixels. After a forward pass through the neural network, the filters are updated when the gradient is calculated in a backward pass. The pooling layer helps reduce computation time, abstracts features, and also helps prevent overfitting during training. The pooling function we used was ‘max’ pooling—it takes the maximum value of a pixel region and discards the other pixel values moving forward in the network. Dropout is a regularization technique used to avoid overfitting. Some number of layer outputs are ‘dropped out’ of the neural network during training and replaced with 0’s. The central idea of a dropout layer is that it introduces noise into the network such that the model is less susceptible to patterns unique to the training data that might not be present in the prediction data (Chollet, 2018).

Three main CNN architectures were tested to automatically predict the locations of salt in the subsurface: a 2D patch-based CNN, a 3D patch-based CNN, and an adaptive 2D U-Net. The first two approaches use a similar concept of feeding many small patches into the CNN. For example, a whole image might actually be 1465 × 1601 pixels, but many 128 × 128 overlapping patches could be extracted for training. Extending to 3D, overlapping mini cubes could be extracted for training and prediction. Figure 1 shows a representative 2D patch-based CNN architecture.

After Ronneberger et al. (2015) and their U-Net approach, we designed and tested an adaptive U-Net architecture (Figure 2). The *adaptive* term signifies that the network can receive any size of image as an input and will produce that same size as output. Unlike many patch-based networks, our adaptive U-Net takes a complete line of the image (an inline) of any size as input. We found that the adaptive U-Net

10.1190/segam2020-3417484.1

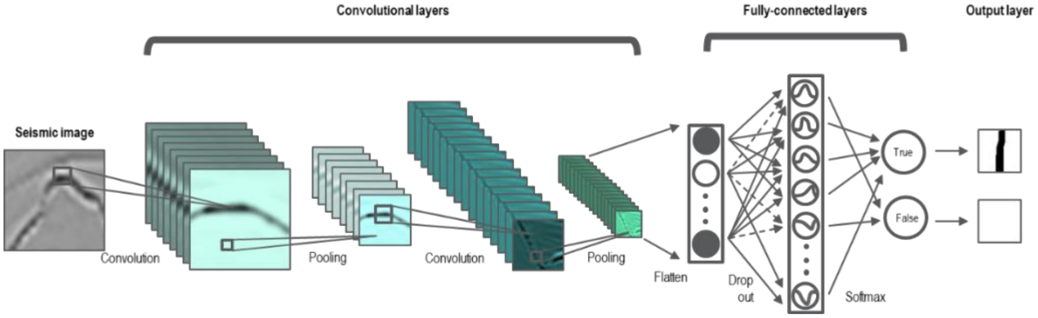

Figure 1: An example of a patch-based CNN architecture: a small patch of a seismic image is fed into the network, with a corresponding salt prediction as the output (Zabihi Naeini and Di, 2018).

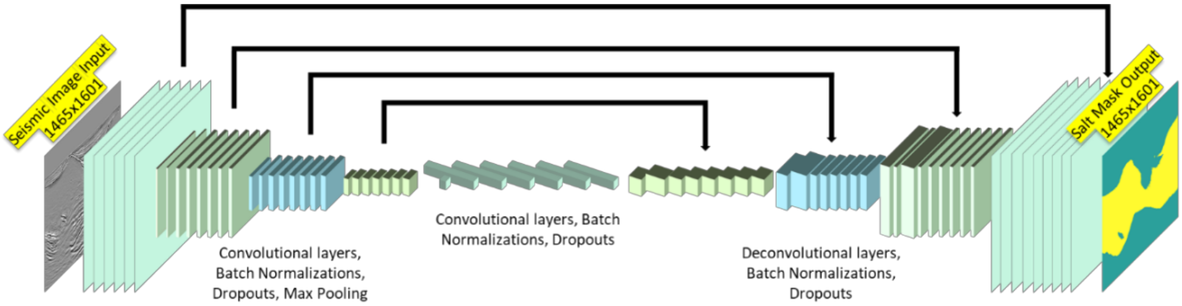
Seismic Image Input 1465x1601
Convolutional layers, Batch Normalizations, Dropouts, Max Pooling
Convolutional layers, Batch Normalizations, Dropouts
Deconvolutional layers, Batch Normalizations, Dropouts
Salt Mask Output 1465x1601

Figure 2: An adaptive U-Net architecture was used for training and predicting the locations of salt. The shapes of layers displayed are their true relative shapes to the input layer. The example input on the left is one complete inline from the offshore Julia data and the corresponding output is a mask of 0's (no salt as green) and 1's (salt as yellow).

architecture significantly outperformed the patch-based CNN approaches in both accuracy and speed.

Case Studies

We now show the application of adaptive U-Net on two case studies from the Gulf of Mexico. The first survey is 18 km × 18 km at the Julia field. Julia is situated about 425 km southwest of New Orleans, Louisiana at water depths of around 2200 m. The second survey, which we call Ogo, is 41 km × 47 km, situated about 225 km southwest of New Orleans at water depths of around 50 m.

Julia

The Julia survey contains massive salt bodies that are up to 5 km thick. The input images to the adaptive U-Net algorithm were inlines only. There were a total of 1405 inlines, 1465 crosslines, and 1601 depth samples. Thus, each training data point was 1465 × 1601 pixels. The complete 3D volume had already been hand-picked, making it an ideal candidate to test the accuracy of U-Net as compared to the manual picks. Five randomly selected inlines were used for training. Two additional unseen inlines were used as validation data to ensure that the model was not overfitting on the training data and to pick the best model. Manual picking took around 20 minutes on 5 inlines, and the U-Net training time was 11 minutes. Once the model was trained, the prediction time took only 3 minutes for all 1405 inlines. In total, the experiment took 34 minutes, whereas completing the manual picking took around 3 weeks on the full 3D volume. Only 5 out of 1405 inlines needed to be manually picked for the ML model to succeed, or 0.4% of the total data. 99.6% of the data were unseen to the neural network. Figure 3 shows an example inline and crossline from the Julia survey, comparing the manually picked salt to the predicted salt. The salt accuracy was 99.4%, the sediment accuracy was 98.6%, and the overall accuracy was 98.9%. Very few training samples were required to obtain such high accuracies, indicating a well-built architecture. A 50% distribution of salt to no salt also helped the network to learn quickly.

Ogo

The Ogo survey in comparison to the Julia survey contains much smaller, dispersed salt bodies with thicknesses closer to 1 km. There were a total of 1900 inlines, 1640 crosslines, and 1501 depth samples. Each training data point was thus 1640 × 1501 pixels. Similar to the Julia survey, the complete 3D volume was already hand-picked, and so it served as a

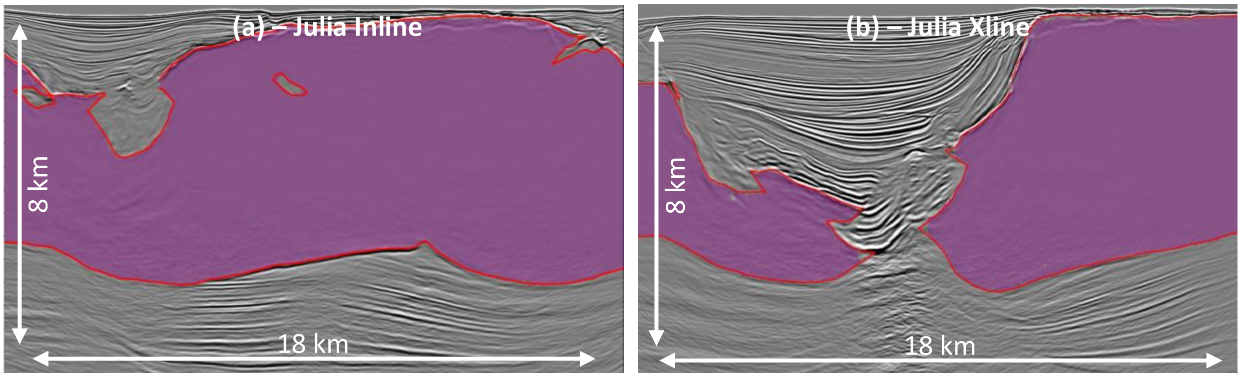

Figure 3: On the Gulf of Mexico Julia data, a comparison is shown between a manually picked salt boundary (red line) and the adaptive U-Net-predicted salt body (purple). An excellent fit between human and ML exists in the inline direction (a) and the crossline direction (b), even though the network only trained and predicted on inlines. The salt bodies in this survey frequently take up approximately 50% of the area and extend upward very close to the ocean bottom floor.

good second accuracy measure for the adaptive U-Net algorithm.

Initially, we used the same strategy for Ogo as for Julia: only 5 inlines were used for training. However, because the salt to sediment ratio was much smaller for Ogo at around 10%, more inlines were required during training to achieve high accuracies. Using 45 inlines for training resulted in a salt accuracy of 96.6%, a sediment accuracy of 99.9%, and an overall accuracy of 99.7%. The discrepancy in the salt accuracy and the sediment accuracy is likely due to the fact that the network had more sediment to learn from than salt, and thus performed better predicting the locations of sediment. Figure 4 shows an example inline and crossline of the Ogo data with salt picks overlaid. Though more inlines in this case would need to be hand-picked for training, only 2.3% of the total data were needed. The total time of hand-picking the 45 inlines was 3 hours, training was 1.25 hours, and prediction on 1900 inlines was 3 minutes, summing to a total of 4.3 hours for the end to end ML workflow. Manually picking the entire 3D Ogo volume took around 3 weeks. Figure 5 shows a 3D comparison between manual picks and ML picks.

Conclusions

The adaptive U-Net was able to distinguish between salt and sediment at an overall accuracy of 98.9% on the Julia data, and 99.7% on the Ogo data. The Julia data only required training on 0.4% of the data, and the Ogo data required training on 2.3% of the data. The prediction time for Julia's 1405 inlines was 3 minutes and Ogo's 1900 inlines was also 3 minutes. For these case studies, we effectively reduced the time to pick salt from 3 weeks to 3 minutes. In practice, for cases like Ogo, the ML workflow allows one to apply it in an "active learning" mode where the practitioner could start with five inlines and sequentially add more training data by predicting on other inlines and modifying the output for more labelled sections.

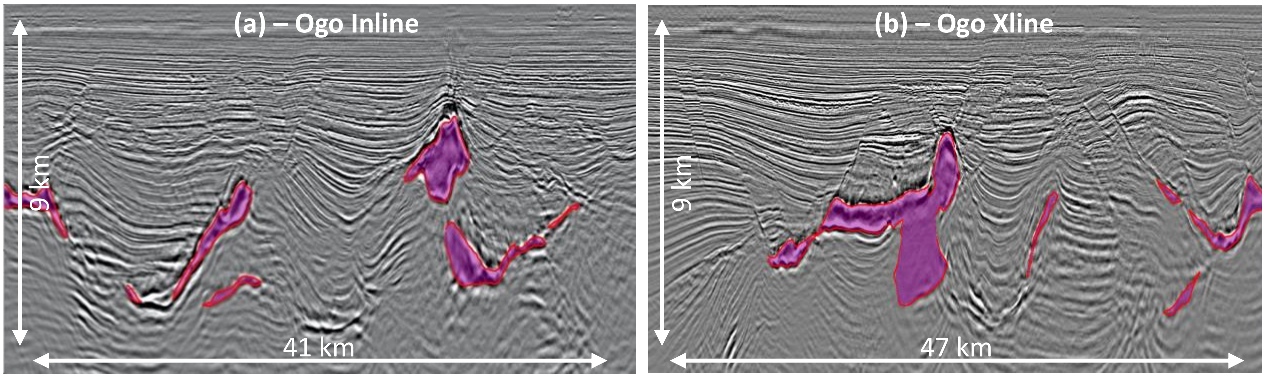

Figure 4: On the Gulf of Mexico Ogo data, a comparison is shown between a manually picked salt boundary (red line) and the adaptive U-Net-predicted salt body (purple). An excellent fit between human and ML exists in the inline direction (a) and the crossline direction (b). The salt bodies are much smaller and discrete compared to the Julia survey.

10.1190/segam2020-3417484.1

One limitation of this study is that only final migrated images were used. In a real-world scenario, salt model building occurs in stages, generally as follows: first a top of salt boundary is picked, and then the data are migrated; second, a bottom of salt is picked, and the data are migrated again. This process continues until the lowest body of salt is picked. Provided a small percentage of the overall data would be hand-picked for training at each stage, we are confident that adaptive U-Net can save significant time at each iteration of model building. Should more training data become available in the future, a more generalized approach might use patches directly as input to U-Net, as model sizes must stay consistent for transfer learning. In order to keep the large-scale context, large patch sizes of 500 × 500 could be used.

Acknowledgements

We would like to thank Fairfield Geotechnologies for permission to show the data and for the GPU resources.

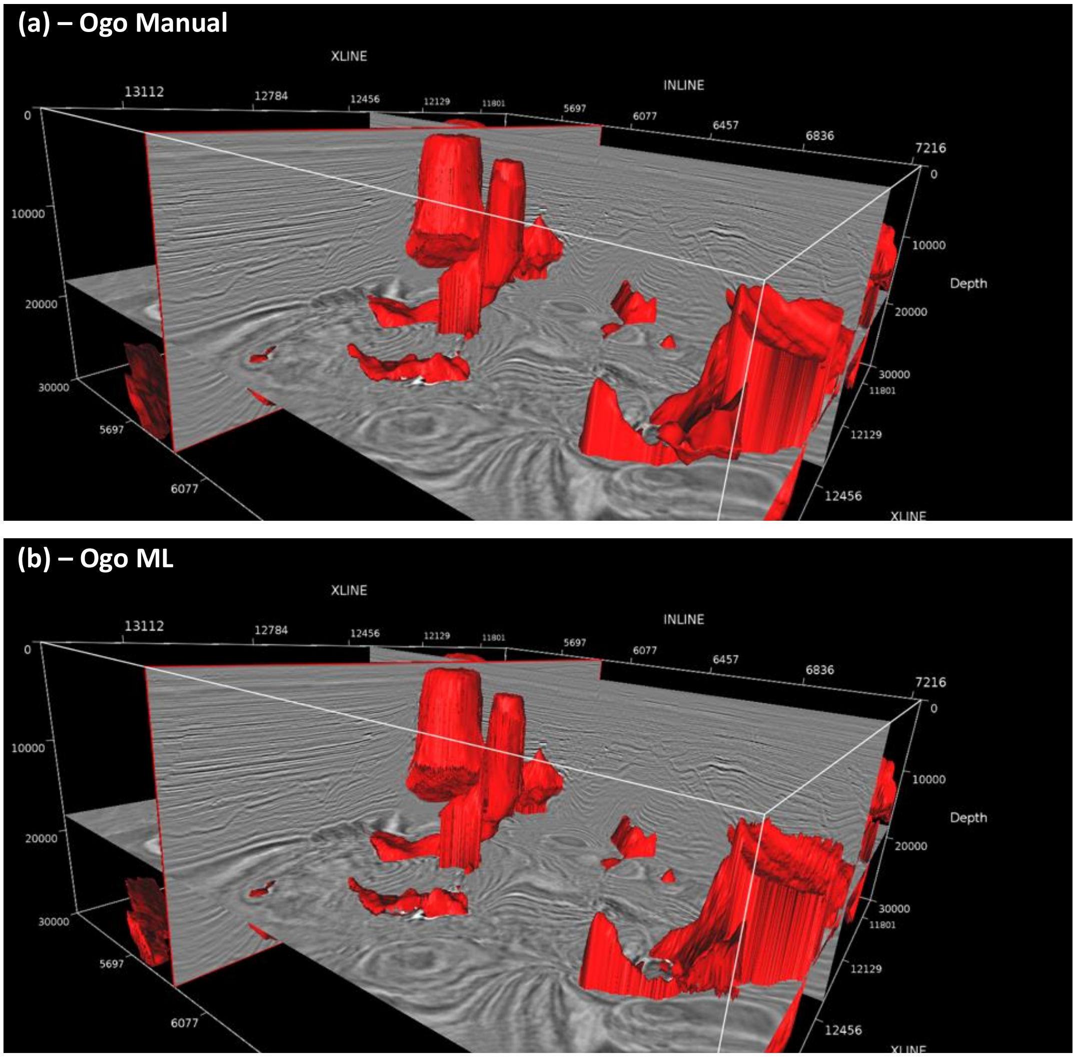

Figure 5: A 3D view of the manually picked salt bodies of the Ogo data (a) and the ML prediction of salt bodies (b). The sliced 3D migration volume is shown in grayscale on a sample inline, crossline, and depth slice. The depths displayed here are in feet.

REFERENCES

Chollet, F., 2018, Deep learning with Python: Manning Publications Co.

Guillen, P., G. Larrazabal, G. González, D. Boumber, and R. Vilalta, 2015, Supervised learning to detect salt body: 81st SEG Annual International Meeting, Expanded Abstracts, 1826–1829, doi: https://doi.org/10.1190/segam2015-5931401.1.

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 - MICCAI 2015, **9351**, 234–241, doi: https://doi.org/10.1007/978-3-319-24574-4_28.

Russakovsky, O., et al., 2015, ImageNet large scale visual recognition challenge: International Journal of Computer Vision, **115**, 211–252, doi: https://doi.org/10.1007/s11263-015-0816-y.

Sen, S., S. Kainkaryam, C. Ong, and A. Sharma, 2020, SaltNet: A production-scale deep learning pipeline for automated salt model building: Leading Edge, **39**, 195–203, doi: https://doi.org/10.1190/tle39030195.1.

Shafiq, M. A., Z. Wang, G. Alregib, A. Amin, and M. Deriche, 2017, A texture-based interpretation workflow with application to delineating salt domes: Interpretation, **5**, no. 3, SJ1–SJ19, doi: https://doi.org/10.1190/INT-2016-0043.1.

Shi, Y., X. Wu, and S. Fomel, 2018, Automatic salt-body classification using a deep convolutional neural network: 84th SEG Annual International Meeting, Expanded Abstracts, 1971–1975, doi: https://doi.org/10.1190/segam2018-2997304.1.

Wang, B., Y. Kim, C. Mason, and X. Zeng, 2008, Advances in velocity model-building technology for subsalt imaging: Geophysics, **73**, no. 5, VE173–VE181, doi: https://doi.org/10.1190/1.2966096.

Wu, X., 2016, Methods to compute salt likelihoods and extract salt boundaries from 3D seismic images: Geophysics, **81**, no. 6, IM119–IM126, doi: https://doi.org/10.1190/GEO2016-0250.1.

Wu, X., S. Fomel, and M. Hudec, 2018, Fast salt boundary interpretation with optimal path picking: Geophysics, **83**, no. 3, O45–O53, doi: https://doi.org/10.1190/geo2017-0481.1.

Zabihi Naeini, E., and H. Di, 2018, A comparative analysis of convolutional vs. deep neural networks: PETEX, Extended Abstract, 39–43.

Zeng, Y., K. Jiang, and J. Chen, 2019, Automatic Seismic Salt Interpretation with Deep Convolutional Neural Networks: Proceedings of the 2019 3rd International Conference on Information System and Data Mining, 16–20, doi: https://doi.org/10.1145/3325917.3325926.

10.1190/segam2020-3417484.1