Seismic Images Interpretation to Discover Salt Domes Using Deep Fully Convolutional Network

Shms Aldeen S. Al-Duri*1 and Amel H. Abbas*2

1Mustansiriyah University, Baghdad, Iraq.
2Mustansiriyah University, Baghdad, Iraq.

1Email: shms.aldeen@tu.edu.iq
2Email: de.amelhussein2017@uomustansiriyah.edu.iq

Abstract. Interpretation of seismic data of seismic information available in the seismic image is an important process for oil exploration. By using computer vision and semantic segmentation techniques, defined the goal to extract salt domes, Which is a trap created due to the penetration of rock layers from below by the inelastic boulders of salt. The steps of this research explain the proposed model based on the semantic segmentation technique using the Fully Convolution Network as a U-net, one of the deep learning techniques. To discover salt domes to assist geophysicists in interpretation and detect oil in seismic sections. This model is designed to interact with the TGS dataset. Presented the research steps using the Python programming language. From the proposed U-net Model, we got a value of 97.50 as a result of training accuracy and 94.80 as a result of testing accuracy. The testing accuracy result of the proposed method was compared with other methods in the field of computer vision techniques used in interpreting seismic data, and the comparison result showed that the proposed U-net Model gives better results.

Keywords: U-net, Semantic segmentation, Seismic interpretation, salt domes.

1. Introduction
There are numerous studies conducted in the field of interpretation of seismic data and analysis of seismic information available in the treated seismic sections. Many structural characteristics help in the search for oil and gas in the seismic section, including the channel, fault, and salt domes. This research will focus on discovering salt domes. [2,1] Salt domes, also called salt penetrators, is a trap that is created due to the penetration of the layers of rocks from below by the inelastic salt boulders, and their appearance is as in Figure 1. Can find hydrocarbons in the deformed layers and in the sides of salt domes. In order to distinguish salt domes from other geological deformations that have a salt core, it can be observed that salt domes have a circular or nearly oval shape in their cross section and also by noting that the horizontal dimensions are less or almost equal to their vertical dimensions. [3,1].
Computer vision technologies can be used in many of our daily life applications in a very large way. Computer vision techniques include classifying and dividing images. [4] In this paper, will use the term semantic segmentation to detect salt domes, as this technique is considered more commonly used in recent years. semantic segmentation is a fundamental task in computer vision, compared to image classification or target detection, Semantic Segmentation gives us a more detailed understanding of images as it focuses on the automatic extraction of information from images, and it can be expressed more broadly as the process of classifying each pixel in a particular image. It is a special case of the classification problem in that each input image contains multiple output classes for each pixel. Semantic segmentation is a more difficult problem than categorizing images because needs to know the exact location and surroundings of objects in a given image, not just what is in that image. The category group is preset and each pixel must be named as a category. [5]

2. Related work
There is much research in the field of seismic interpretation, including: Shengrong L. and et al. Suggested using CNN as a method for detecting the seismic fault by [6]. This network uses only a very small training set to do the seismic fault detection by semantic segmentation and the CNN experiment to do it. U-Net predicts Pixel-by-Pixel if fault or non-fault. When applied, good results were obtained from real data. In this method, only seven seismic sections were used to train CNN. It demonstrated that fault detection can be successful in any seismic section of the same size. Muhammad A. and et al [7], proposed a new seismic attribute, to simulate the process of visual interpretation, the gradient of textures, which can determine texture differences in 3D space. Based on the attribute size, apply a universal bound to mark areas containing salt dome boundaries. The experimental results showed that by making use of the strong coherence between adjacent seismic sections, the proposed method can define the surfaces of the salt boundaries more accurately than modern detection methods that label salt dome boundaries only in two-dimensional (2D) seismic sections. And Zhen W. and et al. [8], suggested the possibility of using deep learning in the interpretation of seismic sections. Researchers have focused mainly on the two most important geological structures, namely the salt domes and the Fault. The importance of this method in reducing the interpretation time through automatic interpretation based on machine learning algorithms, by applying a deep learning network CNN to detect the fault and salt domes in 3D seismic sections. Thilo W. [9], presented a method that highlights the possibility of using machine learning in seismic explanations that often require a lot of time and effort from the geophysicist or geologist. Three-dimensional seismic data were used in this method. This method included five main stages: the first includes extracting the features from the basic characteristics of the seismic data, the second manually performing the classification of the earthquake face on a thousand examples, the third training the system for a set of these models, the fourth testing these models on Other and fifth examples: Choosing the best model that has the highest accuracy and applying it.
3. Fully Convolutional Network FCN

A convolutional network (FCN) is a network that does not contain any fully connected layers in which the learnable layers are just convolutional layers. There are many advantages of FCN over fully connected multi-layer convolutional networks. First, the ability to use different image sizes as input to FCN, while fully connected layered convolutional networks can only accept fixed-size input images. Second, fully connected layers need a great deal of memory and numerical computation storage because many parameters have to be learned, while convolutional networks can learn good features and require much fewer parameters. Completely connected layers can be converted into convolutional layers by replacing completely connected layers with convolutional layers. After that, the entire network becomes completely convolutional. [10]

FCN or fully convolutional networks are typically used for semantic segmentation and were first introduced by Long et al. FCN “decodes” features extracted from convolutional and pooling layers into a segmentation map. Since feature extraction in CNN is a type of image-to-feature encoding, FCN is essentially an encoder/decoder network. The encoder (feature extraction) is effective at distinguishing between classes, and shorthand exists to reproduce features in a segmentation map with the same resolution as the original image. The reduction process is carried out through transfer convolutions (sometimes called expansive or reverse convolutions), which, as the name implies, is the opposite of convolutions. Since there is no fully connected layer at the end of the FCN and the transported convolutions is a fixed input size, FCN comes with the practical benefit of being able to take any image size as an input. The weights that affect the projections are learned from the transferred convolutions during training, which means that the network essentially learns how to decode (shorthand) its encoding feature (shorthand). [11, 12, 13]

While the fully connected layer of the image classification network completely ignores the spatial information and presents only one feature vector for the entire image, the entire convolutional layer produces a vector per pixel. Based on this feature map, a classification can be made in terms of pixels. This results in a probability map for each category. To restore the original image dimensions, this map is expanded with so-called deconvolutions. See Figure (2) to visualize the FCN architecture. [10,11]

![Figure 2. Structure FCN, a format of encoding/decoding, illustrated by [12].](image)

4. U-net

U-Net is a popular architecture that is widely used in the biomedical community. The name U-net denotes the U shape in which the layers are serialized. The idea behind this architecture is to combine lower and higher level feature maps through bypass connections, which will improve the localization of high-accuracy features. This network is based on the FCN
The network is designed so that it can obtain accurate segmentation using just a few training images. The non-convolutional layers increase the final resolution of the output map. In the path of the decoder, a large number of feature maps are applied to each convolutional layer. This will spread the contextual information to higher resolution layers. Figure (3) shows the U-Net architecture (for 32 x 32 pixels at the lowest resolution). Each blue square corresponds to a multi-channel feature map. The number of channels is indicated at the top of the square. The x-y-size is provided at the bottom left edge of the box. White squares represent maps of copied features. The arrows indicate the different operations .. [10,14, 15]

Figure 3. U-Net architecture, illustrated by [11].

U-Net usually starts training with starting weights at random. As known, to avoid overfitting the network training, the dataset must be large enough to hold millions of images. [16]

5. TGS dataset

The proposed model is designed to deal with a public data set that is available for download from [17]. The main properties of this data set are as follows:

1- The TGS dataset consists of 4000 seismic images and 4000 mask grayscale images which are actually the ground truth corresponding to each seismic image that indicates whether salt is in the image or not. The size of each seismic image and mask is (101 x 101) pixel and its format is PNG.
2- Used to predict the possibility of salt domes being present in the area in which a seismic survey is being conducted.

Figure (4) shows samples of seismic images with the corresponding mask in the TGS dataset, showing on the left side the seismic image and the corresponding mask on the right side, as the white color in the mask indicates the presence of salt domes. (The black border on the seismic image is for illustration, not out of the image).
6. Methodology

All the stages of the proposed model for U-net are shown in Figure (5). Seen the model begins with preparing the seismic image of the input that the model will process. This stage consists of the following sub-stages: First: Data splitting In this sub-stage, the input seismic image with its corresponding mask is divided into two subgroups: a training set and a testing set. Split the input dataset into two sets. As mentioned earlier that the TGS dataset is made up of 4000 seismic images, in this work the input dataset will be divided by 10% of the total data for the testing group and 90% for the training group. The second sub-stage Loading of image in which the seismic image is imported from the training set file, both its seismic image and it's ground truth. In the third sub-stage, two types of data augmentation were applied to the input training set. These are flip vertical and flip horizontal. This becomes the size of the training set consisting of three categories for each seismic image, with which the model will be trained. The fourth sub-stage in the preparing stage is to change the size of the input image to the model. The image size in this model has been changed from the original image size 101 × 101 pixels to 96 × 96 pixels. Where a size close to the size of the original image was experimented with and is divisible by 32 in order to obtain better results. Then, according to Figure (5), moved to the model building stage. This model consists of two sub-stages which are encoder and decoder and each stage consists of many different layers linked together. Table (1) and (2) illustrate the structure of the proposed U-net model. The number of layers for the form is twelve. Each layer in the model performs its own function to make the model perform seismic segmentation. Increasing the size of the network depth means increasing the resulting image resolution.
Table 1. U-net Encoder architecture

| Layers | Operation         | Kernel                  | Size          |
|--------|-------------------|-------------------------|---------------|
| Input  |                   |                         | $96 \times 96 \times 1$ |
| Layer1 | Convolution       | $3 \times 3 \times 32$  | $96 \times 96 \times 32$ |
|        | Max pool          | ; padding : same         | $48 \times 48 \times 32$ |
| Layer2 | Convolution       | $3 \times 3 \times 64$  | $48 \times 48 \times 64$ |
|        | Max pool          | ; padding : same         | $24 \times 24 \times 64$ |
| Layer3 | Convolution       | $3 \times 3 \times 128$ | $24 \times 24 \times 128$ |
|        | Max pool          | ; padding : same         | $12 \times 12 \times 128$ |
| Layer4 | Convolution       | $3 \times 3 \times 256$ | $12 \times 12 \times 256$ |
|        | Max pool          | ; padding : same         | $6 \times 6 \times 256$  |
| Layer5 | Convolution       | $3 \times 3 \times 512$ | $6 \times 6 \times 512$  |
|        | Max pool          | ; padding : same         | $3 \times 3 \times 512$  |
| Layer6 | Convolution       | $3 \times 3 \times 1024$| $3 \times 3 \times 1024$ |

Table 2. U-net decoder Architecture

| Layers | Operation         | Kernel                  | Size          |
|--------|-------------------|-------------------------|---------------|
| Layer 7| Deconvolution Layer 6 | $3 \times 3 \times 512$  | $6 \times 6 \times 512$ |
| Layer 8| Deconvolution Layer 7 | $3 \times 3 \times 256$  | $12 \times 12 \times 256$ |
| Layer 9| Deconvolution Layer 8 | $3 \times 3 \times 128$  | $24 \times 24 \times 128$ |
| Layer 10| Deconvolution Layer 9 | $3 \times 3 \times 64$   | $48 \times 48 \times 64$ |
| Layer 11| Deconvolution Layer 10 | $3 \times 3 \times 32$  | $96 \times 96 \times 32$ |
| Layer 12| Deconvolution Layer 11 | $3 \times 3 \times 1$   | $96 \times 96 \times 1$ |

In the training stage, the network for the proposed model is trained using a training data set (seismic image) in the first layer of the network and then passes through the overall network layers.

Adam Optimizer has been used in training, and it is considered one of the best optimization algorithms. The epoch number is set for training. That is, the number of times the network training will be repeated, at 100. To improve network performance, stop the training early was used. This is to bypass the overfitting condition. The training stage took about (36) hours to train, which equates to a day and a half. The training time also depends on the complexity of the network architecture and the size of the training data set. The network can use a much longer time in the training stage if the GPU is used. The dropout value was determined to be 0.3, which means an extra layer neglects a percentage of units as needed to bypass the overfitting state. The Batch size is set to 16 for each iteration.

Finally, the testing stage. The trained network was tested using a testing dataset from the TGS dataset. The test images use the trained model and pass through all its layers using parameters that include the weights reached to the network and the number of filters.

7. Experimental Results

Instructions for the Keras Library are a library that defines a simplified interface for implementing deep neural networks, found in the Jupyter Notebook python conda, to apply the above-mentioned methodology steps for semantic segmentation, which serves as an interpretation of salt domes in seismic sections. To train the proposed model, the training set
has been loaded. The training set is original size of 3600 and in order to get better accuracy have been increased the data set to the size of training set to 10800 seismic images. Figure (6) shows a sample seismic image and after applying the data augmentation to it. Then resized the seismic image in the training set from 101 pixel x 101 to 96 pixel x 96. Figure (7) shows the image after resizing.

![Image](image.png)

Figure 6. Shows the sample of data augmented: a) Orginal seismic image, b) Horizontal flip augmented, c) Vertical flip seismic image.

![Image](image.png)

Figure 7. Shows sample for the result of resizing the seismic image: a) seismic image, b) its a corresponding mask.

After the preparing stage, implement the proposed model using U-net layers defined as twelve layers. Each image in the training set goes through all of these layers. The complete 12-layer U-net architecture consists of six major convolutional layers. In the first convolutional layer, the number of filters used was 32 and the filter size specified \([3 \times 3]\) resulted in 32 different features map for a single input image. In the second convolutional layer, 64 filters and specified filter size \([3 \times 3]\) were used resulting in 64 different features map for a single input image. In the third convolutional layer, the number of filters used was 128 filters and the filter size specified \([3 \times 3]\) resulted in 128 different features map for a single input image. In the fourth convolutional layer, a number of filters 256 filters and the filter size defined as \([3 \times 3]\) were used resulting in 256 different features map for a single input image. In the fifth convolutional layer, 512 filters and a specified filter size \([3 \times 3]\) were used resulting in 512 different feature maps for a single input image. In the sixth convolutional layer, the number of filters used was 1,024 and the filter size specified \([3 \times 3]\) resulted in 1024 different features map for a single input image. The decoder for this network consists of six layers that deconvolution the previous layer with changing the filter size in reverse for the convolution process.

7.1 Training and Testing U-net

To train the network, the Adam optimizer and the binary crossentropy loss (module loss) function were used. Where the standard was applied to specify the binary_accuracy function of the optimal form of the metric unit. This scale shows the percentage of pixels that are correctly classified. Good results are achieved with binary_accuracy to select the optimal model.

To control training operations, callback objects have been implemented in Keras. The appropriate classes are part of the Callbacks module and the suggested solution used:
EarlyStopping, ReduceLROnPlateau, and ModelCheckpoint. An EarlyStopping callback was used, as the name suggests, to terminate training, a dropout value was chosen in the form 0.3. EarlyStopping was activated after 10 epochs when the binary accuracy in the validation set did not improve. And using ReduceLROnPlateau to reduce the learning rate by using a certain factor when a particular parameter that is being tracked has not improved in a certain number of periods. In the proposed solution, binary_accuracy was tracked in the validation set, and if there were no improvements after 5 periods of training, the learning rate decreased by 0.1. Finally, ModelCheckpoint is used to outcome the best-measured model with binary accuracy in the validation set. Every time accuracy is improved, the model is saved. So when training is finished, the model that has the best performance in the validation set is obtained. And Figure (8) shows the training results of randomly selected seismic images.

![Figure 8](image)

**Figure 8.** Shows samples training results for seismic images: a) Seismic images, b) Corresponding mask for the seismic image, c) Result of a training stage for semantic segmentation in a grayscale image, c) Result of a training stage for semantic segmentation in a binary image.

In the testing stage, as mentioned earlier, a 10\% test data set (TGS), consisting of 400 seismic images, was used. Seismic images have been entered into the trained U-net. Figure (9) shows the testing results of randomly selected seismic images.

Obtained best training accuracy value for the trained model was 97.50 by choosing the best loss value between seismic image and ground truth, which was 0.152. testing accuracy was 94.80 when the image size of 96 × 96 pixels and dropout value 0.3 was chosen as the input for the proposed model. This model was designed by choosing the image size that is close to the original image size as it is divisible by 32. And showed that the proposed model works efficiently when the size is The input image for U-net is 96 × 96 pixels.
Figure 9. Shows samples testing results for seismic images: a) Seismic images, b) Corresponding mask for the seismic image, c) Result of a testing stage for semantic segmentation in a grayscale image, c) Result of a testing stage for semantic segmentation in a binary image.

Several parameters of the model tested which resulted in different values for training accuracy and testing accuracy, all of these tests are shown in Table (3).

| No. | Image size | Size of data | No. of filter | Batch size | Dropout Value | Acc. train | Acc. Test |
|-----|------------|--------------|---------------|------------|---------------|------------|-----------|
| 1   | 96 × 96    | 18000        | 32            | 32         | 0.5           | 96.68      | 93.93     |
| 2   | 96 × 96    | 18000        | 32            | 128        | 0.5           | 96.41      | 93.57     |
| 3   | 128 × 128  | 18000        | 32            | 32         | 0.5           | 94.98      | 92.20     |
| 4   | 128 × 128  | 18000        | 32            | 128        | 0.5           | 95.92      | 92.77     |
| 5   | 96 × 96    | 10800        | 32            | 16         | 0.2           | 96.76      | 93.49     |
| 6   | 128 × 128  | 10800        | 32            | 16         | 0.2           | 96.23      | 94.11     |
| 7   | 128 × 128  | 10800        | 32            | 16         | 0.3           | 93.88      | 93.46     |
| 8   | 128 × 128  | 7200         | 16            | 16         | 0.5           | 94.18      | 92.02     |
| 9   | 128 × 128  | 7200         | 16            | 32         | 0.5           | 93.10      | 91.40     |
| 10  | 96 × 96    | 3600         | 16            | 16         | 0.5           | 95.52      | 93.68     |
| 11  | 96 × 96    | 3600         | 16            | 32         | 0.5           | 96.36      | 93.56     |

Table 3. Result of some experiments with different parameters input to Proposed Model

7.2 Comparison with Previous Studies

The results of this research were compared with several proposed methods of semantic segmentation using machine learning and deep learning techniques, each of these methods used different data sets to test and train the model. Table (4) shows the test accuracy values resulting from these different methods and using different data sets.

As Table (4) shows that the test accuracy value of the proposed U-net model is higher than other methods and the reason is due to the increase in the number of layer i.e. the increase in the proposed network depth.
Table 4. Comparison with Some Related Work Studies.

| Method Name                                           | Dataset | Testing accuracy |
|-------------------------------------------------------|---------|-----------------|
| 3D multidirection edge decoder [8]                    | 3D F3   | 89.64           |
| Encoder-Decoder CNN [10]                              | Real data | 91.20           |
| Gradient of textures [11]                             | 2D F3   | 93.50           |
| Proposed U-net method                                 | TGS     | 94.80           |

8. Conclusion

After applying the U-net model based on semantic segmentation and deep learning and comparing the results obtained in this thesis, it was concluded: The proposed U-net is, as in all deep learning techniques, highly dependent on the size and quality of the training data. Where using a large data set to train the model leads to better performance. Also by running many experiments to train the model using different parameters for dropout value, batch size, and the number of filters, it was found that the more filters the number of filters and the image size entered into the U-net is closer to the size. The original image get better accuracy. The reason for the superior performance of the proposed model is the depth of the U-net architecture which uses the power to extract different level features from the input images. That is why recommend that future researchers refine the proposed model by using a larger data set, filter number, and size greater than 32, and by using the GPU instead of the CPU and adopting a transform augmentation when more data is needed to train the network. Also, recommend implementing the proposed model in the automated system driver.

Acknowledgments

The authors would like to thank Mustansiriya University (www.uomustansiriyah.edu.iq) Baghdad – Iraq for its support in the present work.

References

[1] Enwenode O.,(2014), Seismic Data Analysis Techniques In Hydrocarbon Exploration, Elsevier.
[2] Ying J.,(2017), Detecting Geological Structures in Seismic Volumes Using Deep Convolutional Neural Networks, Master Thesis : Rheinisch-Westfälische Technische Hochschule Aachen.
[3] Abdullah Al-G., & et al., (1998), Geological Encyclopedia, Kuwait Foundation for the Advancement of Sciences.
[4] Bernd J.,(2000), Computer vision and applications, Academic Press.
[5] King-Sun F. and JK M., (1981), A survey on image segmentation, Pattern Recognition,13(1):3[16].
[6] Shengrong L., Changchun Y., Hui S., and Hao Z.,(2018), Seismic fault detection using an encoder–decoder convolutional neural network with a small training set, Journal of Geophysics and Engineering : 16, 175–189.
[7] Muhammad A. Sh., Zhen W., Asjad A., Tamir H., Mohamed D. and Ghassan A., (2015), Detection of Salt-dome Boundary Surfaces in Migrated Seismic Volumes Using Gradient
of Textures, SEG New Orleans Annual Meeting, [Online], Available: http://dx.doi.org/10.1190/segam2015-5927230.1

[8] Zhen W., Haibin D., Muhammad A. Sh., Yazeed Al., and Ghassan Al.(2018), Successful leveraging of image processing and machine learning in seismic structural interpretation: A review, the leading edge, Vol. 37(6), Pages: 402–480.

[9] Thilo W. , Indranil P. , Robert L. Gawthorpe , and Haakon F.(2018), Seismic facies analysis using machine learning, GEOPHYSICS, VOL. 83, NO. 5 ; P. O83–O95.

[10] Zhenyi W., (2018), Automatic Brain Tumor Segmentation by Deep Convolutional Networks and Graph Cuts, The University of Western Ontario, master thesis.

[11] Long, J., Shelhamer, E., Darrell, T., (2014), Fully convolutional networks for semantic segmentation, arXiv:1411.4038 [cs.CV], Online], Available: https://arxiv.org/abs/1411.4038.

[12] Fredrik S.,(2019), Semantic and Instance Segmentation of Room Features in Floor Plans using Mask R-CNN, Uppsala Universitet, master thesis.

[13] Jonathan L., Evan Sh. and Trevor D., (2015), Fully convolutional networks for semantic segmentation. In Proceedings of 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, USA.

[14] Snuverink I. A. F., (2017), Deep Learning for Pixelwise Classification of Hyperspectral Images, Delft University of Technology, master thesis, [Online], Available: https://repository.tudelft.nl/islandora/object/uuid%3Ae6ce206b-09d1-4511-8349-b5e67f96c2d2.

[15] Olaf R., Philipp F. and Thomas B., (2015), U-Net: Convolutional Networks for Biomedical Image Segmentation, Computer Science: ArXiv, [Online], Available: arXiv:1505.04597.

[16] Vladimir I. and Alexey Sh., (2018), TernausNet: U-Net with VGG11 Encoder Pre-Trained on ImageNet for Image Segmentation, Computer Science : ArXiv, [Online], Available: arXiv:1801.05746.

[17] https://www.kaggle.com/c/tgs-salt-identification-challenge/data.