

## Research Article

# Seismic Facies Segmentation Using Ensemble of Convolutional Neural Networks

**Bilal Abid, Bilal Muhammad Khan , and Rashida Ali Memon **

*National University of Sciences and Technology, Islamabad, Pakistan*

Correspondence should be addressed to Bilal Muhammad Khan; [bm Khan27@gmail.com](mailto:bm Khan27@gmail.com)

Received 30 January 2022; Accepted 10 March 2022; Published 26 March 2022

Academic Editor: Nawab Muhammad Faseeh Qureshi

Copyright © 2022 Bilal Abid et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The use of machine learning for seismic interpretation is a growing area of interest for researchers. Manual interpretation demands time and specialized effort. The use of machine learning model will expedite the process. The Convolutional Neural Networks (CNNs) are a class of deep learning algorithms used for images. In this paper, seismic facies segmentation using encoder-decoder architecture of CNNs is proposed. The proposed method filled the gap using a multimodel approach for seismic interpretation. The novelty of the model is that it is not limited to the current dataset and semantic segmentation models. The encoder-decoder architecture input and output size is the same, and it allows the labelling of each pixel of the image. Four models are trained on the open-sourced F3 block Netherlands dataset. Images of  $128 \times 128$  were extracted from the data. Data augmentation is used in two of the models to increase the data size for better model learning. Results of individual models and their ensemble are compared. Ensemble is performed by taking the average of the probabilities of the classes obtained from the trained models. Ensemble gave the superior results. Seven classes are segmented with a global pixel accuracy (GPA) of 98.52%, mean class accuracy (MCA) of 96.88%, and mean intersection over union (MIoU) of 93.92%.

## 1. Introduction

Discovery of the new reserves of oil and gas is strategic for countries. Seismic reflection surveying is used to obtain subsurface information to locate drilling locations in the oil and gas industry [1]. The structural and stratigraphic geometric features and potential hydrocarbon reservoirs can be configured from the seismic reflections. Accurate interpretation of seismic amplitude and zones is significant for the discovery of oil and gas [2]. Accurate interpretation leads to a lesser number of drills and significantly impacts the characterization of the reservoir. Seismic amplitude is manually interpreted by field experts and geophysicist to differentiate between bodies of rocks (strata) with different physical properties [3]. The geophysicist needs to manually analyse the images generated from seismic survey to mark the boundaries between different strata known as horizon. The process needs to be done for thousands of seismic images. Traditional method of manual interpretation for seismic delineation takes time, and there are false positives [4]. Accurate

interpretation saves economic resources, as resources are spent for the drilling of the reservoirs.

For data and image classification, methods in computer vision and pattern recognition look for unique features. Success of these methods depends on the feature extraction techniques [5]. Convolutional Neural Networks (CNNs) provided an alternative for automatically learning the domain specific features [6]. The CNNs consists of a large number of convolutional layers, which can learn the most useful features automatically and eliminate the need for manual feature extraction techniques. The initial layers of the CNN detect basic features like edges, colours, and shapes. The deeper layers then combine the basic features to extract more complex features that represent the whole image. The hierarchical structure in the convolutional nodes of CNNs identifies the complex objects.

A deep CNN contains millions of parameters and requires a large amount of data for training the network. The CNNs are used for various image applications such as image classification, object detection, localization, and object segmentation.

Semantic segmentation networks are the improvisation of CNNs in the form of encoder-decoder architecture [7]. The encoder is used for object detection and classification, and the decoder is used for object localization. Semantic segmentation is the process of taking an image and labelling each pixel in that image with a certain class.

Seismic facies interpretation is an important step for hydrocarbon exploration and exploitation [8]. A major issue of the application of deep learning to seismic data is lack of availability of labeled datasets. Limited datasets undermine the potential of deep learning. Annotating the seismic data is a time-consuming task and requires subject-level expertise. Very few labeled seismic datasets are available online. It is not feasible to use a trained model from one location and implement it directly to another location because the structures and dynamics are different for each location. Researchers annotated their own seismic facies datasets [9–12], annotated the Netherlands F3 block (North Sea) dataset for training deep neural networks, and open sourced it for further research. The authors used the 3D seismic reflection data in conjunction with the well log data to manually annotate the 3D seismic dataset.

The North Sea contains hydrocarbon deposits, and this area is well-studied. The continental shelf of the North Sea is situated in the waters of the Netherlands. A rectangular area of dimension 16 km  $\times$  24 km known as the F3 block is located in the North Sea. A 3D seismic survey was conducted in 1987 to search for hydrocarbons and understand the lithostratigraphy of the area. The data was open sourced by dGB Earth Sciences. The F3 dataset is used extensively in research and studies [12].

Fully Convolutional Network (FCN), SegNet, U-Net, ENET, and DeepLab are few of the semantic segmentation networks with their distinct architectures. DeepLab's variant DeepLabv3Plus and Seg-Net are fine-tuned in the proposed method using pretrained encoders: vgg-16 and ResNet-18. The results show a mean class accuracy of 96.88% and mean intersection over union of 93.92%.

*1.1. Artificial Intelligence.* Machine intelligence is another name for artificial intelligence. When humans are born, they have natural sensors in their bodies. They start to observe the world, their parents first; listen to voices; feel the hotness or coldness of a body, etc. With the help of such observations, they develop their responses and actions. This can be termed as training phase. With the passage of time, the brain develops, memory becomes stronger, and now, the actions and responses are well connected to the previous experiences. This example is given as an analogy to understand the human intelligence.

Similarly, a computer can be equipped with sensors and storage. The computer can then be trained by feeding thousands of images and voices. With the help of these images, the computer can learn the features differentiating them. After the training phase when an unseen (not from the images used to train the computer) image is shown to the computer, it can tell either it is from the class of images or not with very promising results using the modern techniques in AI especially deep CNN.

*1.2. Machine Learning.* Machine learning (ML) brings the promise of deriving meaning from all the data which is presently available in terms of audio, images, and datasets. Google Search is one of the examples of machine learning; each time Google Search is used, several ML models working in the background are activated to understand and correct the text and adjust the results according to user interests based on previous searches and other data being collected from other applications a user is using. There are several ML-based learning algorithms available to train the models which include Random Forest, Decision Tree, Naïve Bayes, SVM, and kNN to name a few.

## 2. Related Work

The application of computational techniques on seismic data followed the trend of development of the machine learning community. Initially, different mathematical features and techniques are calculated to assist geologists in making predictions. The popularity of machine learning techniques enabled researchers to feed the calculated features in different machine learning models to extract results. In the third stage, due to GPUs, it was possible to make extremely complex models which gave rise to Convolutional Neural Networks. With the CNNs, there is no requirement of feature engineering and images are directly fed to the network.

Initially, different computational techniques were used to classify an image by their geological attributes that presented an application of textural analysis to 3D seismic volumes [13]. In this paper, the authors combine the image textural analysis with a neural network classification to quantitatively map seismic facies in three-dimensional data. In 2011, for exploration geology and geophysics, seismic texture analysis was a developing concept and a large number of different algorithms were published in the literature. In [14], review of the seismic texture concepts and methodologies, focusing on latest developments in seismic amplitude texture analysis, with particular reference to the gray-level cooccurrence matrix (GLCM) and the texture model regression (TMR) methods is presented. There are discontinuities in seismic images with varying illumination and contrast. In [15], a solution using the congruency of phase in Fourier components is proposed. The proposed algorithm shows far more better and efficient results in terms of accuracy as compared to the texture-based methods for salt dome boundary detection. Data-driven algorithm is proposed in [16]. The proposed protocol overcame the limitations of existing texture attribute-based salt dome detection techniques which depend on the relevance of attributes to the geological nature of salt domes and the number of attributes used for classification. The authors used a gray-level cooccurrence matrix (GLCM) with attributes extracted from Gabor filter to delineate salt domes in seismic data. In [17], seismic attributes are combined with their spatial locations for unsupervised seismic analysis using fuzzy *c*-means algorithm. This method reduced the effect of seismic noise presented in discontinuous regions. A comprehensive evaluation of accuracy and performance of three texture descriptors, Gabor filters, GLCM, and Local Binary Patterns

(LBP), is presented in [18] for seismic image retrieval to assist human interpreter in selecting region of interest (ROI).

Before the deep learning techniques became popular, features used to be hand engineered and fed into machine learning algorithms such as Random Forest Regressor, Support Vector Machines, and Boosting algorithms. In [19], an extremely randomized tree to automatically identify salt boundaries is presented. The proposed protocol extracted the features of signal amplitude, curve length, and second amplitude for each of the voxels and made predictions using extremely randomized trees. After the prediction, a postprocessing step is added to further increase the accuracy. Reference [20] used a machine learning approach for classifying facies. The 3D seismic reflection data of the North Sea is used in the paper. Fifteen different attributes are extracted for each pixel, such as reflector dip, continuity, and frequency range. The attributes are trained on twenty ML models such as Support Vector Machines (SVM),  $K$ -nearest neighbours, regression trees, and neural networks. The best result of accuracy 98.3% is obtained using SVM. Reference [21] uses a 3D seismic survey data from New Zealand and calculated four features (peak spectral frequency, GLCM, homogeneity, and curvedness). These 4 features are input to Artificial Neural Networks (ANN) and Support Vector Machines (SVM), and ANN gives better accuracy on the test set as compared to SVM. Reference [22] uses 6-7 features (4-5 measured properties and two geologically derived features) and trained the model using  $K$ -nearest neighbours (KNN), Naive Bayes, fuzzy logic, and ANNs. ANN is the most effective model and gives the best result.

Unsupervised learning techniques like self-organizing maps and principal component analysis (PCA) are also used to classify lithostratigraphic zones. Reference [23] uses unsupervised learning techniques like  $K$ -means clustering, PCA, projection pursuit, vector quantization, and Kohonen self-organizing maps. Reference [24] uses competitive neural networks on seismic data to distinguish the distinct seismic behaviour of facies. Heterogeneity of classes was indicated in the results; however, the classes in the results were not labeled. In [25], Kohonen self-organizing maps are used to estimate the number of seismic facies and make maps. Wavelet transform was also used to detect seismic trace singularities. Others tried deep convolutional autoencoders (DCAE) for facies classification. Reference [26] uses DCAE as it can learn nonlinear, discriminant, and invariant features from unlabeled seismic data. The results show that DCAE outperforms conventional methods like  $K$ -means or SOMs. Moreover, the results of DCAE are of much higher resolution and highlight important information. Reference [27] uses sparse autoencoder architecture that can detect major geological features from unlabeled seismic data. The model is tested on real and synthetic seismic data in order to extract relevant structures from the data.

The development of computational power led to a greater emphasis on the use of supervised CNN algorithms for seismic applications. The CNN supervised approach can be divided into seismic classification and seismic seg-

mentation methodologies. Seismic classification uses the convolutional, max pooling, and fully connected layers to predict the class of the centre pixel of the image. In [28], a novel CNN consisting of six convolutional layers followed by a fully connected layer is presented. The network uses this architecture for salt detection. Cubes of  $65 \times 65 \times 65$  are extracted from the 3D seismic data, and the centre of the cube represented the class (salt or not salt) of that pixel. Reference [29] uses CNNs (vgg-16, ResNet-50, and Waldeland) to classify seismic images of the F3 dataset. The centre pixel for each image is classified, and this step is repeated such that all pixels are classified. The results for vgg-16 and Waldeland architectures show significant improvement towards accuracy however found to be ineffective for ResNet-50. An investigation over the use of fully supervised CNNs and semisupervised Generative Adversarial Networks (GANs) is presented in [8]. The models are tested on realistic synthetic images. Results shows that CNNs perform better in scenarios where abundant data is available; however, GANs work better on new sites with limited availability of data. Reference [30] uses a custom-built CNN architecture to detect faults from a 3D seismic cube. The input to the network consisted of three orthogonal slices of  $24 \times 24$ , and the voxel at the intersection is classified as fault or not fault. The network is trained using synthetic images and tested on both synthetic and real data. The results show that CNN obtains a classification accuracy of 74% on the real dataset.

In [31], an encoder-decoder structure is introduced for CNNs. The encoder learns the distinctive features from the image, and the decoder maps back the features semantically. In [2], a novel segmentation network (Danet-FCN) is proposed. Danet is combined with FCN for pixel classification. For validation, F3 and the Penobscot datasets are used. The mean IoU of more than 98% is obtained on both of the datasets. A modification of Danet-FCN is presented in [4] in order to propose a new architecture Danet-FCN2 and Danet-FCN3 by removing the fused connections. The Danet-FCN3 improves the IoU to 99% on the Penobscot dataset. Reference [32] uses U-Net architecture with dilated convolution and soft attention mechanisms. The soft attention mechanism allows the model to suppress noise and learn the main features. The authors trained the models from scratch. CNN results show improvement with the use of dilated convolutions and soft attention mechanisms. Reference [33] presents work on the TGS salt classification dataset at Kaggle. Semisupervised technique is used for salt classification using an ensemble of CNNs. An iterative process is used by which predictions at each stage are treated as pseudolabels and the network is retrained using the training data and confident pseudolabels. The ensemble of U-Net architecture with the encoder of ResNet34 and ResNeXt50 is used. The results show an IoU of 0.896. In [34], a modified U-Net architecture is proposed to detect salt from seismic images. The model is trained and tested on synthetic dataset, and the results show a mean IoU of 90.53%. A comparison of 3D-based patch model and encoder decoder architecture on the F3 dataset is presented in [9]. The dataset is divided into 9 facies and manually interpreted the data for training. The work draws conclusion that the encoder-decoder model

gives better results at near real-time speeds, at the expense of long training time. Reference [13] presents a fully annotated 3D geological model of the Netherlands F3 block. This model is based on the study of the 3D seismic data in addition to 26 well logs and is grounded on the careful study of the geology of the region. The study proposed two baseline models for facies classification based on deconvolution network architecture and made their codes publicly available. The first model is patch based; in this, patches are extracted from the crosslines, and in lines and the model is trained on it. The second approach is section based; in this, complete in lines and crosslines are fed to the model. Results shows that the section-based model (MCA of 0.817) gives better results than the section-based model (MCA of 0.705).

**2.1. Dataset.** A major challenge in seismic facies classification is the availability of annotated dataset. The data needs to be manually labeled by the domain experts. The process of labelling requires availability of a geophysicist and is subjective to human bias. Researchers working in the field of application of deep learning labeled their own datasets due to these limitations. The authors in [13] labeled the F3 Netherlands dataset and open sourced it for further research. In this paper, the same labeled F3 seismic dataset is used for model verification and results.

The Netherlands F3 block dataset is in 3D NumPy array format. First, the dataset is converted into image/label form (see Figure 1). There were a total of 22368 images, each of which is of the size  $128 \times 128$ . The dataset is split into train, validation, and test sets with a ratio of 60%, 20%, and 20%, respectively (see Table 1). The split is performed randomly. The image is divided into 7 facies which are as follows:

- (i) Upper North Sea Group (upperNS)
- (ii) Middle North Sea Group (middleNS)
- (iii) Lower North Sea Group (lowerNS)
- (iv) Rijnland/Chalk Group (rijanlandChalk)
- (v) Scruff Group (scruff)
- (vi) Zechstein Group (zechstein)
- (vii) Background

### 3. Methodology

In this paper, two deep learning models for semantic segmentation are used DeepLabv3Plus and SegNet. The models will be discussed with a brief overview of semantic segmentation and data augmentation.

Main elements of the framework are

- (a) Dataset preprocessing
- (b) Selection of semantic segmentation networks
- (c) Selection of pretrained CNNs to be used as encoders
- (d) Ensemble technique

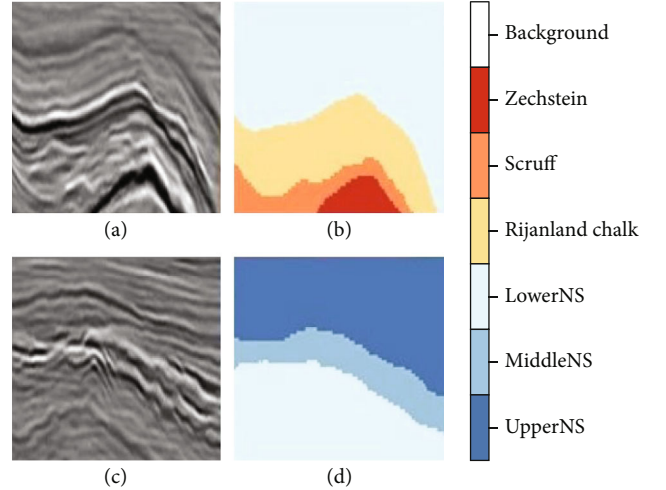


FIGURE 1: (a, c) Show the seismic images. (b, d) Are the labels for (a) and (c), respectively.

TABLE 1: Train, validation, and test splits.

Total patches	22368
Size	$128 \times 128 \times 3$
Train images	13421
Validation images	4473
Test images	4473

Dataset is converted from NumPy arrays to images and labels. This conversion makes the dataset readily usable on any platform. Images and corresponding labels are resized to support the available hardware resources. The resizing of images to a suitable size is an iterative process to avoid GPU memory issues while training. Through iterations, the size of  $128 \times 128$  is chosen. Dataset is split into three parts, namely, training, validation, and testing.

**3.1. Semantic Segmentation.** Semantic segmentation is the process of categorizing each pixel of the image to a class. Semantic segmentation is applicable to a variety of computer vision tasks like autonomous driving and medical image diagnosis. Semantic segmentation is one of the most challenging aspects of computer vision. The classification problem objective is to predict the presence of an object in the image [35]. The difference in segmentation problems is to predict the class of each of the pixel within the image. The image needs to be segmented into different objects like car, pedestrian, roads, and road signals for autonomous driving applications.

In [31], use of CNNs for semantic segmentation is proposed. The authors use skip connections to join the semantic information from a deep layer to the localization information from a shallow layer, to produce the pixel-wise segmentation of the image. The decoder part was implemented as bilinear interpolation. The method improved the PASCAL VOC results by 20%; however, one of the major drawbacks of this technique is that it tends to ignore small objects.

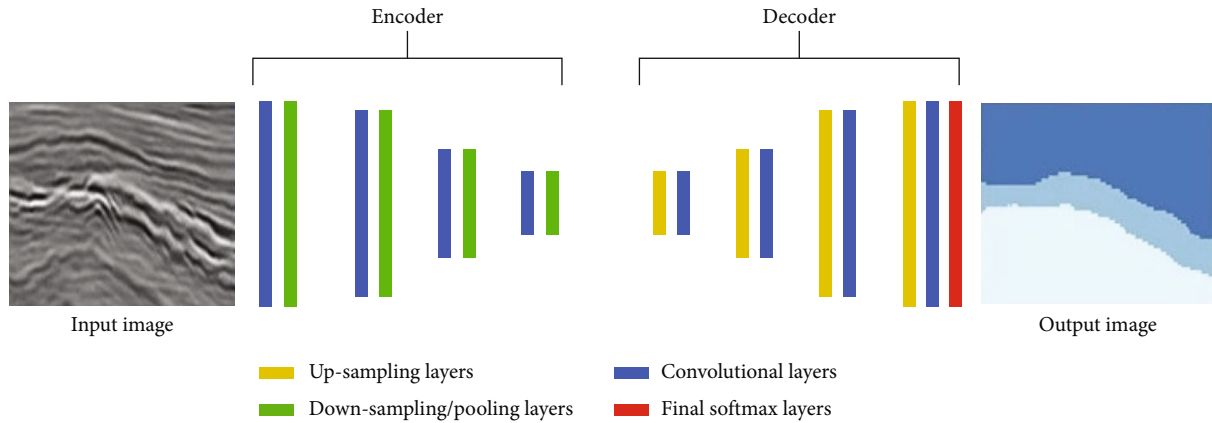


FIGURE 2: The architecture of the encoder-decoder network. The encoder consists of consecutive convolutional and pooling layers. This is followed by decoder with consecutive upsampling and convolutional layers. Softmax layer is used at the output to classify each pixel into a class.

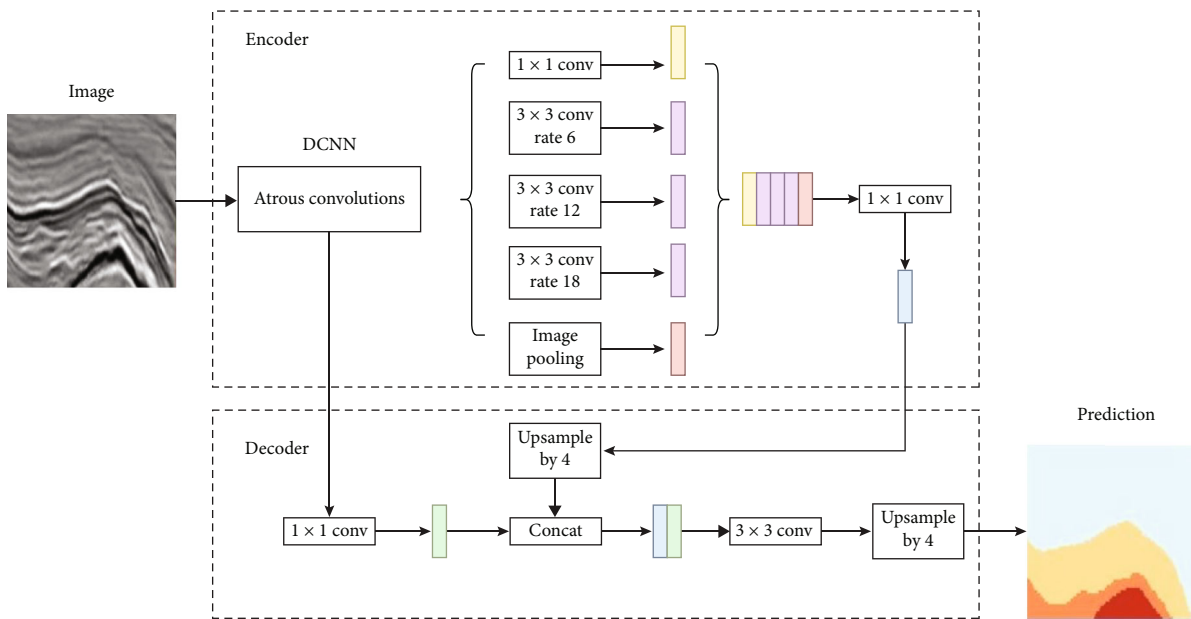


FIGURE 3: DeepLabv3Plus architecture. The encoder applies Atrous convolution at multiple scales [38].

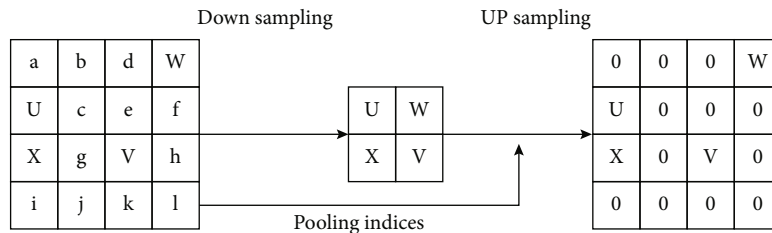


FIGURE 4: SegNet uses nonlinear upsampling strategy. The indices from the corresponding pooling filter from the encoder stage are passed to the upsampling filter.

In [36], a deep deconvolutional network for decoding, which consisted of deconvolution, unpooling, and activation layers is presented. The presented model performs better on the PASCAL VOC 2012 segmentation with an accuracy of 72.5% (see Figure 2). For the encoder-decoder architecture used for semantic segmentation in this paper, the encoder

phase is similar to the conventional CNN classification model and consists of multiple convolutional and pooling layers. Each convolutional layer first convolves its input and then also applies batch normalization and an activation function. The pooling layer is used to downsample the image. In the pooling layer, a sliding window is passed through the image and is

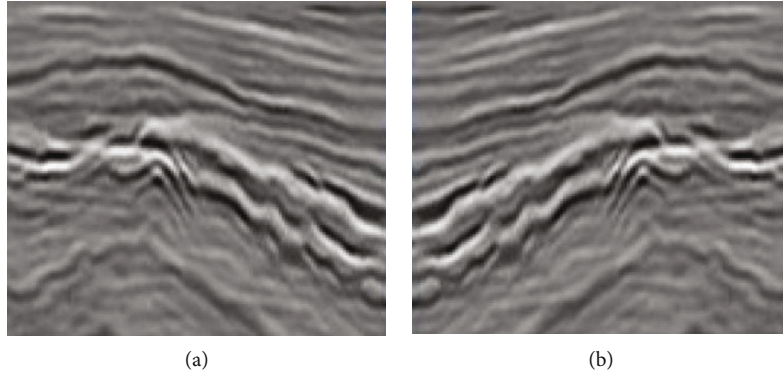


FIGURE 5: (a) Shows the original image. (b) Shows the new image produced by reflection along the  $x$ -axis.

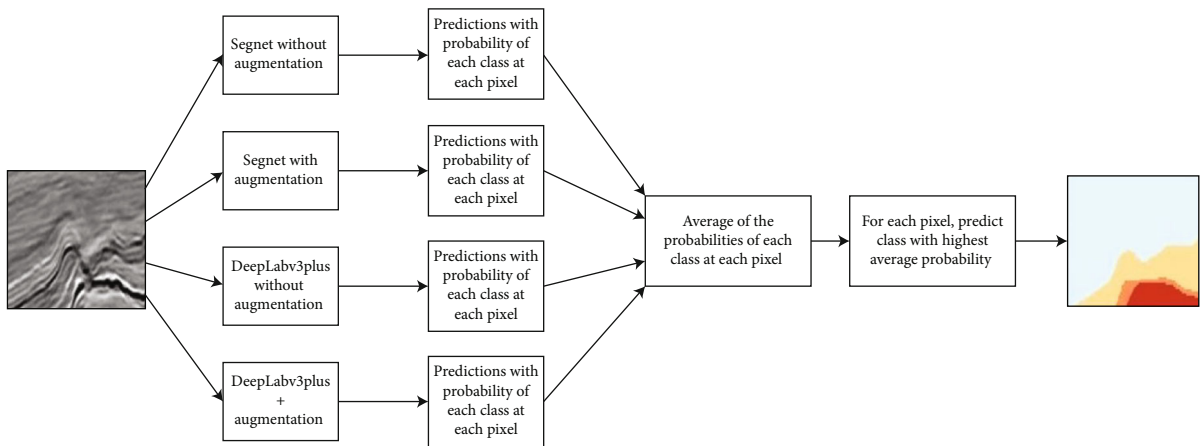


FIGURE 6: The architecture of the ensemble method. The predictions are calculated for all 4 of the models. The predictions are of the shape  $128 \times 128 \times 7$  and show the probability of each pixel belonging to each class. The average of the four  $128 \times 128 \times 7$  arrays is taken. The class with the highest average probability is labeled as the class for that pixel.

used to summarize the information by selecting minimum/maximum/average from the window. The encoder is used to classify the objects within an image. At the end of encoder stage, a low-resolution feature map is obtained. Encoder is followed by a decoder which works in an opposite manner to the encoder. It consists of multiple upsampling and convolutional networks to bring the output to the same size as that of the input. Decoder is used for the localization of objects to generate boundaries of the objects within the image.

For computer vision problems of semantic segmentation, the encoder-decoder architecture gave better results than other CNN architectures. We are using two of the encoder-decoder architectures that are DeepLabv3Plus and SegNet.

**3.2. DeepLabv3Plus.** DeepLab is one of the popular semantic segmentation architectures and is used in [37]. DeepLab is a model designed and open sourced by Google. It uses Atrous convolution in place of deconvolutional networks. Atrous convolution allows enlarging the field of view of filters without increasing the number of parameters or the amount of computation. Multiple Atrous convolutions are used in parallel to catch the contextual information at multiple scales.

Deep convolutional network consists of multiple layers due to which information of the smaller scale objects is lost.

This is because the input feature map reduces as we move in the network. Atrous convolutions are used in DeepLab in the convolutional layers to counter this problem. Atrous convolution consists of an additional parameter of rate  $r$  which is the stride at which input signal is sampled. The normal convolution is a specific case for  $r = 1$ . In [38], denser features are extracted by the use of Atrous convolutions without the need of extra parameters.

DeepLabv3Plus consists of multiple Atrous convolutions (see Figure 3). The Atrous rate applied to each of the convolutional layers is different which enables to extract features at different resolutions. This enables the extraction of denser features. DeepLabv3Plus obtained an accuracy of 89% on PASCAL VOC 2012 test datasets.

This research work uses DeepLabv3Plus upgraded versions; the results show improvement across object boundaries. ResNet-18 is used as an encoder for DeepLabv3Plus. Two variants of the model are trained; one is with data augmentation, and the other one is without data augmentation.

**3.3. SegNet.** SegNet is a semantic segmentation model developed by the University of Cambridge [7]. The encoder consists of 13 convolutional layers in the VGG16 network. At the decoder stage, it performs nonlinear upsampling, which

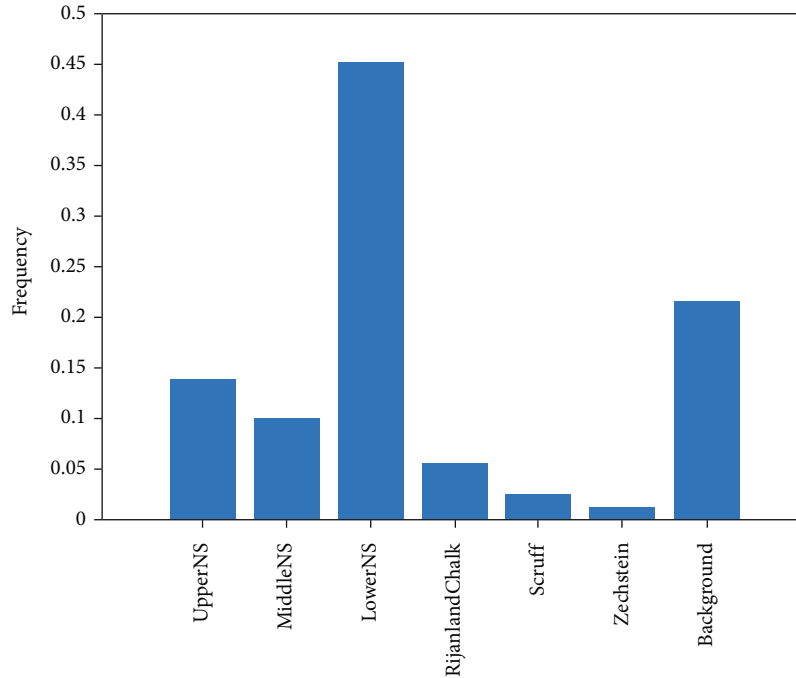


FIGURE 7: Visualization of the ratio of the pixel counts of classes. Lower NS is the class with the highest ratio of pixels at about 45%.

TABLE 2: Median frequency class weights are calculated based on the frequency of pixel counts of each class. Middle NS is taken as the reference and is given a weight of 1.00.

Class name	Pixel count	Image pixel count	Class weights
UpperNS	$5.0432e + 07$	$1.5e + 08$	0.5340
MiddleNS	$3.6849e + 07$	$2.0524e + 08$	1.0000
LowerNS	$1.6513e + 08$	$3.6053e + 08$	0.3920
RijnlandChalk	$2.0753e + 07$	$1.5661e + 08$	1.3549
Scruff	$9.1123e + 06$	$9.18e + 07$	1.8087
Zechstein	$4.6377e + 06$	$3.4882e + 07$	1.3504
Background	$7.9559e + 07$	$1.8937e + 08$	0.4273

uses indices computed in the max-pooling step of the corresponding encoder.

The major difference between SegNet and other encoder-decoder architectures is in the way it upsamples data. During the downsampling in the encoder stage, the pooling indices are stored in SegNet. During the decoder stage, the pooling indices are used to place the values in their original position as before the downsampling. In [7], such case is presented where the information of the pooling indices is passed on to the upsampling stage to produce dense feature maps (see Figure 4). The U-Net presented in [39] on the other hand transfers entire feature map from the encoder to the decoder. U-Net requires greater memory and training time due to this task. The first advantage of SegNet is that in the decoder, the upsampling layer is used which keeps intact the high-frequency details. The second advantage is convolutional layers are used in place of fully connected layers. The convolutional layers can remember the indices of image features as discussed in [38, 40–43].

In this paper, SegNet with VGG16 encoder is used in two variants. One of the variants is with data augmentation, and the other is without data augmentation.

**3.4. Data Augmentation.** CNNs require a large quantity of data to learn from the images and perform well. Data augmentation is a technique to increase the size of data from the original data. New artificial training data is created in this process. The original training data is transformed to produce new training data. The transformations include a range of mathematical operations such as flipping, rotation, padding, scaling, cropping, and changing its colour.

The purpose of data augmentation in the presented models of this research work is to increase the data size. This allows the model to better generalize and learn from the data (see Figure 5) for the effect of reflection in the data along the  $x$ -axis. Translation and reflection are used as the data augmentation techniques.

**3.5. Application.** In this paper, 5 models are developed for seismic facies segmentation of the generated images from the F3 Netherlands block.

- (i) DeepLabv3Plus with ResNet-18 encoder (without augmentation)
- (ii) DeepLabv3Plus with ResNet-18 encoder (with augmentation)
- (iii) SegNet with VGG16 encoder (without augmentation)
- (iv) SegNet with VGG16 encoder (with augmentation)
- (v) Ensemble of the models 1-4

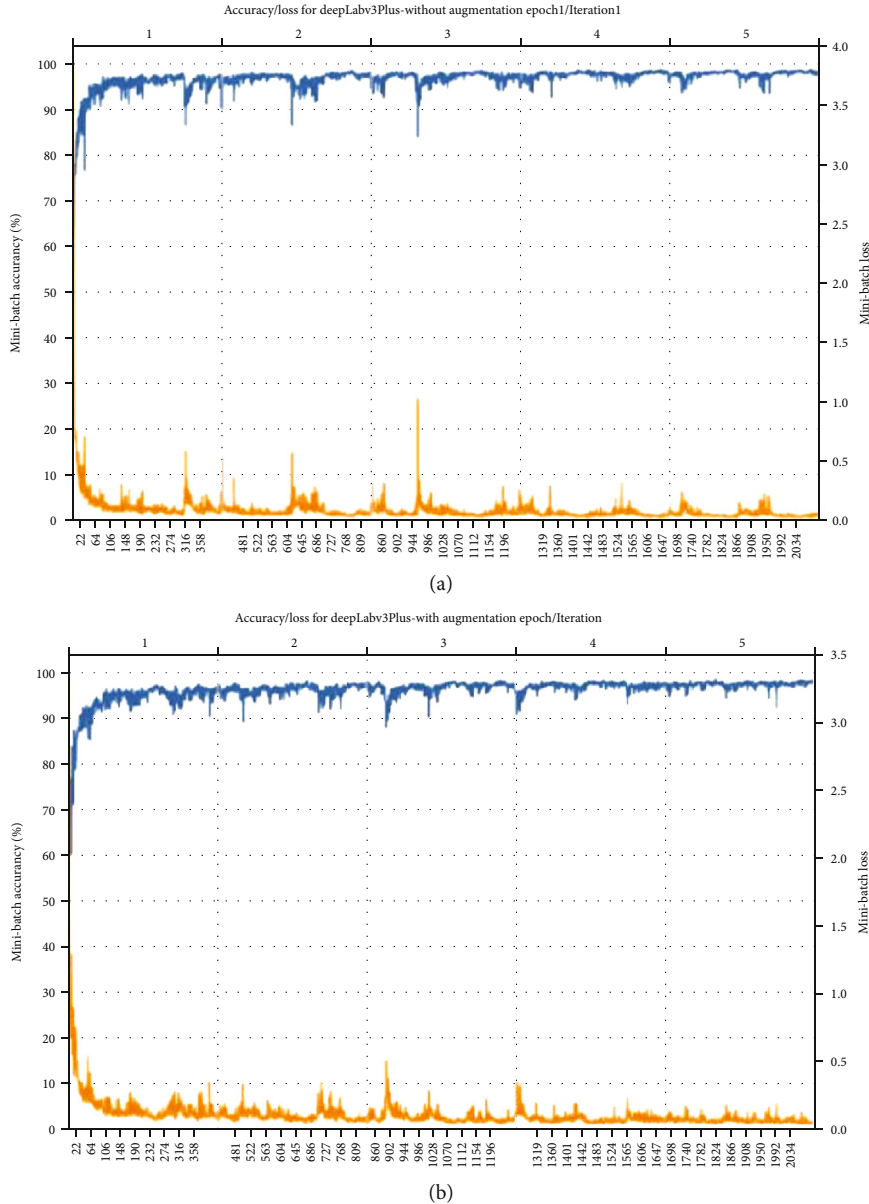


FIGURE 8: The training accuracy and loss plots for DeepLabv3Plus models. (a) Shows the model without segmentation, and (b) shows the model with data augmentation.

Two semantic segmentation networks, DeepLabv3Plus and SegNet are used. Both are based on encoder-decoder architecture. Two models are trained using DeepLabv3Plus with encoder as ResNet-18. One model is trained without data augmentation, and the second was trained with data augmentation (translation and reflection). Two more models are trained using SegNet with encoder as vgg16, one with training data augmented and the other without augmentation. Ensembles of four models are created. For ensemble, averages of the predicted scores of the four models, corresponding to each class for a single pixel, are taken. The class with the highest average probability represented the pixel (see Figure 6). Results of the individual models and the ensemble are compared.

The numbers of pixels of seven classes in generated training patches are imbalanced (see Figure 7). This imbalance is detrimental to the learning process because the learning is biased in favor of the dominant classes. Class weighting is used to handle this issue. Median frequency class weights were calculated (see Table 2)

Weights in the pixel classification layers in encoders for both DeepLabv3Plus and SegNet are replaced with median class weights calculated (see Table 2) to compensate the class imbalance for training to be unbiased. Training options used for four models are as follows: Adam optimizer is used as weight optimization, initial learning rate is set to 0.001, squared gradient decay factor is set to 0.99, and minibatch size was 32. Due to pretrained encoders, training for all four models converged in few



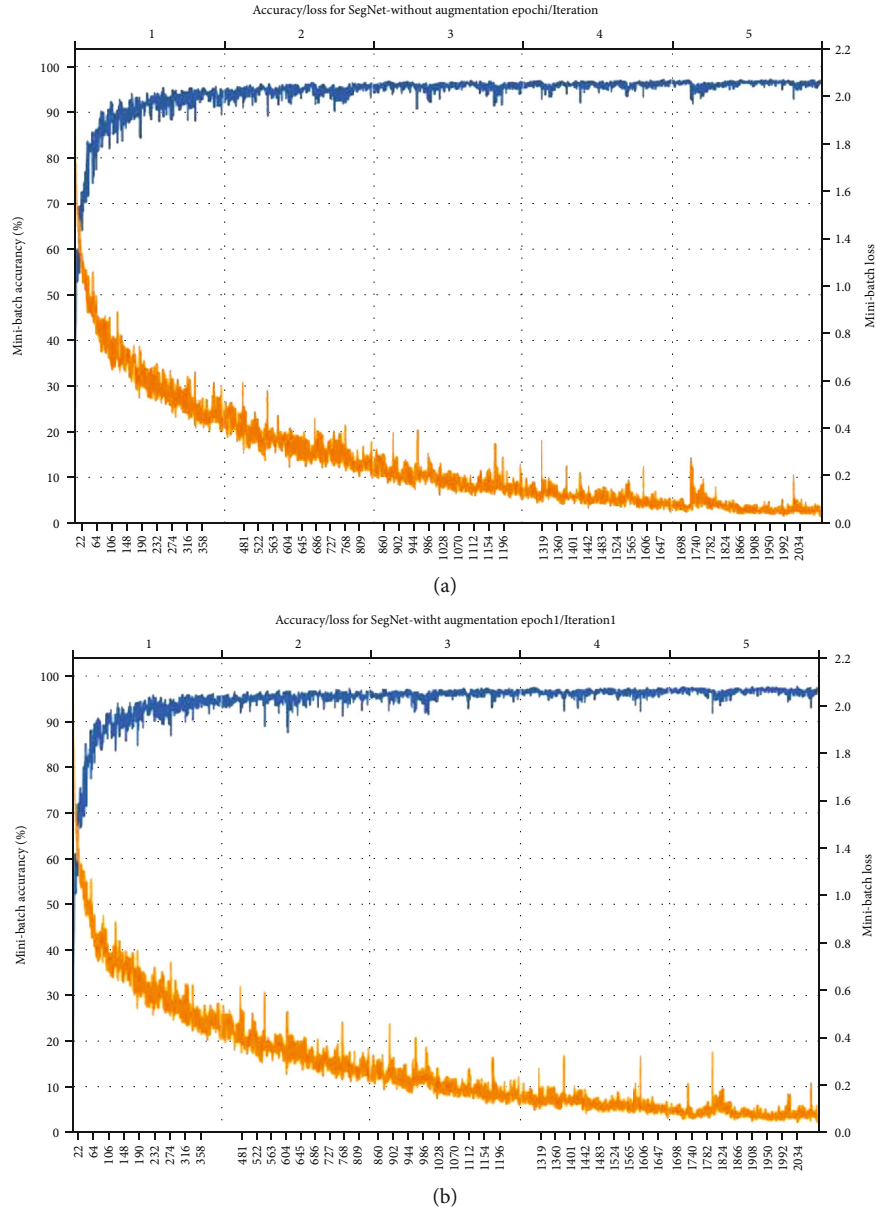


FIGURE 9: The training accuracy and loss plots for SegNet models. (a) Shows the model without segmentation, and (b) shows the model with data augmentation.

epochs. Training of each model is started with ten epochs but due to no change in accuracy and RMSE after five epochs, it is stopped early.

The validation loss and accuracy plots with epochs are presented in Figures 8 and 9) for DeepLabv3Plus and SegNet models, respectively. There is no change in RMSE and accuracy in all of the models after five epochs. Early stopping is applied on all four of the models after five epochs.

#### 4. Results

To assess the performance of the proposed architecture, the following evaluation metrics are used.

- (i) Class accuracy (CA)

The percentage of the correctly classified pixels in a class  $i$ , is called class accuracy.

$$CA_i = \frac{P_i \cap T_i}{T_i}, \quad (1)$$

where  $i$  represents class,  $P$  represents predicted pixels, and  $T$  represents true pixels.

- (ii) Intersection over union (IU)

This evaluation metric measures how to perfectly predict pixel overlap in ground truth. An output of 1 means perfect

TABLE 3: SegNet+augmentation with basenet vgg-16 (Model 1).

	Class accuracy	IoU
UpperNS	0.9781	0.9681
MiddleNS	0.9398	0.8914
LowerNS	0.9832	0.9713
RijlandChalk	0.9514	0.8576
Scruff	0.8257	0.7674
Zechstein	0.9392	0.7356
Background	0.9992	0.9947
<b>Global pixel accuracy</b>	<b>Mean class accuracy</b>	<b>Mean IoU</b>
0.9755	0.9452	0.8837

TABLE 4: SegNet+no augmentation with basenet vgg-16 (Model 2).

	Class accuracy	IoU
UpperNS	0.9851	0.9734
MiddleNS	0.9457	0.9053
LowerNS	0.9919	0.9804
RijlandChalk	0.9146	0.8870
Scruff	0.9520	0.7708
Zechstein	0.7879	0.7378
Background	0.9999	0.9998
<b>Global pixel accuracy</b>	<b>Mean class accuracy</b>	<b>Mean IoU</b>
0.9800	0.9396	0.8935

TABLE 5: DeepLabv3Plus+augmentation with basenet ResNet-18 (Model 3).

	Class accuracy	IoU
UpperNS	0.9856	0.9736
MiddleNS	0.9610	0.9068
LowerNS	0.9882	0.9806
RijlandChalk	0.9493	0.9176
Scruff	0.9445	0.8652
Zechstein	0.9302	0.8902
Background	0.9998	0.9984
<b>Global pixel accuracy</b>	<b>Mean class accuracy</b>	<b>Mean IoU</b>
0.9837	0.9655	0.9331

match, and 0 means no match.

$$IU_i = \frac{P_i \cap T_i}{P_i \cup T_i}. \quad (2)$$

(iii) Mean intersection over union (MIU)

The mean of IU of all classes is mean IU.  $n_c$  is total number of classes.

$$MIU = \frac{1}{n_c} \left( \sum_{\boxtimes} IU_i \right). \quad (3)$$

TABLE 6: DeepLabv3Plus+no augmentation with basenet ResNet-18 (Model 4).

	Class accuracy	IoU
UpperNS	0.9831	0.9713
MiddleNS	0.9250	0.8863
LowerNS	0.9912	0.9755
RijlandChalk	0.9287	0.9034
Scruff	0.9566	0.8318
Zechstein	0.9319	0.8696
Background	0.9969	0.9918
<b>Global pixel accuracy</b>	<b>Mean class accuracy</b>	<b>Mean IoU</b>
0.9795	0.9590	0.9185

TABLE 7: Ensemble of four models.

	Class accuracy	IoU
UpperNS	0.9852	0.9752
MiddleNS	0.9512	0.9101
LowerNS	0.9921	0.9825
RijlandChalk	0.9571	0.9300
Scruff	0.9518	0.8799
Zechstein	0.9443	0.8976
Background	0.9999	0.9994
<b>Global pixel accuracy</b>	<b>Mean class accuracy</b>	<b>Mean IoU</b>
0.9852	0.9688	0.9392

Moreover, the results of the individual models are presented in Tables 3–6. The results show a comprehensive analysis of presented models in terms of accuracy and intersection over union.

In Table 7, results of the ensemble for the proposed four models are presented (see Table 7). Ensemble gives the best result for GPA, MCA, and MIoU. Global pixel accuracy (GPA) is the percentage of pixels over all classes that are correctly classified. Mean class accuracy (MCA) is the average of class accuracy over all classes whereas class accuracy for a class is the percentage of pixels that are correctly classified in that class. Intersection over union (IoU) measures the overlap between the two sets, and it should be 1 if and only if all pixels were correctly classified. Further, averaging IoU over all classes gives the mean intersection over union (mean IoU).

The MCA and MIoU achieved with the ensemble method are 0.9655 and 0.9392 which are the highest amongst all five of the models. This shows that using an ensemble of various models is a useful technique that improves the results. The error in individual models generally occurs at the boundaries of various classes. The ensemble takes the average for each pixel, by which a wrong prediction made by one model can be compensated by the right prediction made by the other models.

Amongst the individual models, DeepLabv3Plus with basenet of ResNet-18 with data augmentations gives far better results, with a MCA and MIoU of 0.9655 and 0.9355, respectively. For both DeepLabv3Plus and SegNet, the results with data augmentation are better than the results

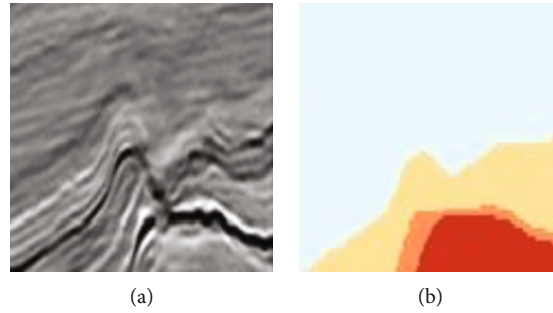


FIGURE 10: An image was randomly selected from the test set. (a) Shows the seismic image and (b) shows the ground truth/labels.

TABLE 8: MIoU calculated on a random test image using individual models and ensemble IoU.

Classes	Model 1	Model 2	Model 3	Model 4	Ensemble
UpperNS	NaN	NaN	NaN	NaN	NaN
MiddleNS	NaN	NaN	NaN	NaN	NaN
LowerNS	0.9856	0.9809	0.9860	0.9721	<i>0.9878</i>
RijlandChalk	0.8701	0.8139	0.8848	0.8504	<i>0.8849</i>
Scruff	0.4740	0.5227	0.5391	0.5582	0.5289
Zechstein	0.9496	0.8369	0.9443	0.9224	0.9418
Background	NaN	NaN	NaN	NaN	NaN
MIoU	0.6558	0.7886	0.8385	0.8257	0.8358

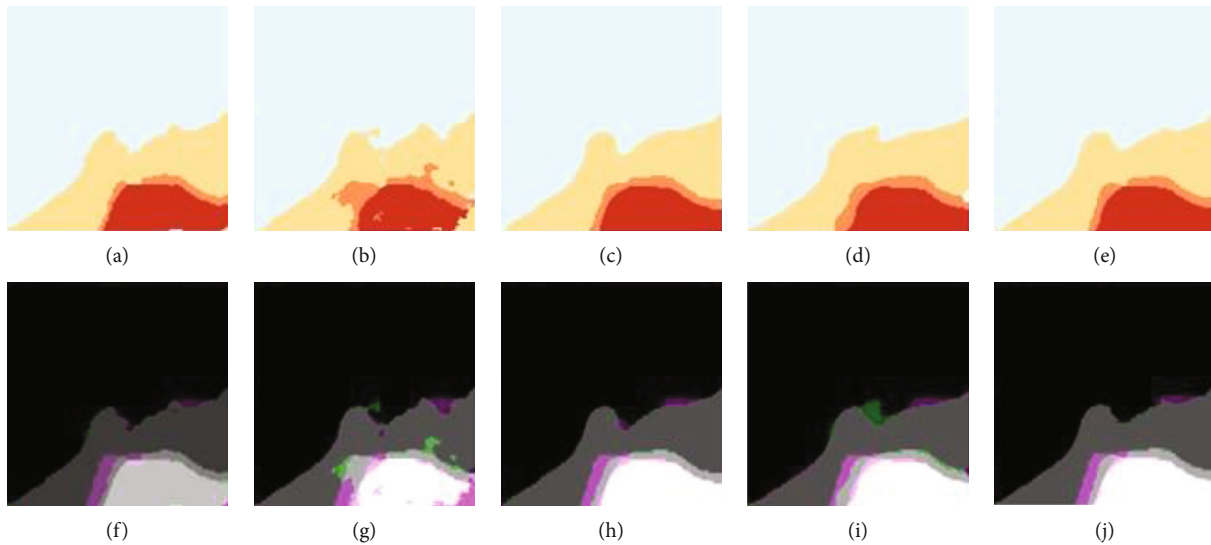


FIGURE 11: (a–e) Predictions by SegNet+augmentation, SegNet+no augmentation, DeepLabv3Plus+augmentation, DeepLabv3Plus+no augmentation, and ensemble. (f–j) Green and magenta colours show the difference between ground truth and prediction of the corresponding model from (a) to (e).

without augmentation. This proves that augmentation is a useful technique that can be used to increase the data size for seismic applications. The model is able to learn better with this increased data size and gives better results.

A random image and its labels are taken from the test set to calculate MIoU using individual models and ensemble (see Figure 10). Ensemble did not give the highest MIoU on this random test image (see Table 8).

Visual results of the ensemble are better than those of other models (see Figure 11) because ensemble gives two

classes the highest IoU whereas other models gives the highest IoU either to one class or none highlighted in italic. In Figure 11, when predicted pixels of a class went beyond the boundary of that class, those pixels are marked as green. When predicted pixels of a class did not reach the boundary of that class, the gap is marked as magenta.

When pixels of a certain class are present in both prediction and ground truth but in nonoverlapping regions, then the IoU is calculated as 0. When a pixel of a certain class is not present in both prediction and ground truth, IoU is

calculated as NaN. Classes with IoU as NaN are not included in calculating GPA, MCA, or MIoU.

## 5. Conclusions

The ensemble of semantic segmentation networks gave better results as compared to individual models. The ensemble of Fully Convolutional Network (FCN), SegNet, U-Net, ENET, and DeepLabv3Plus with ResNet-50 and ResNet-101 as encoders is proposed for future works. Dataset in the form of images is open sourced so that researchers may try other semantic segmentation networks and ensemble them.

The automatic seismic facies segmentation proves to be a promising alternate to manual labelling of seismic facies by geologists. For manual labelling, a high degree of subject expertise is required, which can introduce into the results. The process is very complicated and computationally exhaustive and requires high degree of accuracy. It is not feasible for the geologists to label the entire area, and so usually, labelling is performed on only few of the images or portions of the block. These limitations can be countered by using deep learning techniques since they prove to give accurate results. Deep learning will require the labelling of some of the images, on which the models can be trained. After this, it could be applied to the complete area to make predictions.

## 6. Future Work

For future work, pretrained CNNs like VGG-19, ResNet-34, ResNet-50, ResNet-101, ResNet-152, Inception-v1, Inception-v3, SE-ResNet, ResNext, SENet-154, DenseNet-121, DenseNet-169, and DenseNet201 can be used as encoders in semantic segmentation networks like U-Net, Linknet, PSPNet, and FPN. Moreover, average ensemble results can be compared with voting ensemble. Success of the proposed architecture in the paper encourages improvising not only with other semantic segmentation networks and encoders but also with the data augmentation. Also, in the proposed architecture, crossentropy loss function is used. This loss function can be replaced with other loss functions to check if improvement in IU can be achieved.

## Data Availability

The dataset is publicly available in [12].

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## References

- [1] M. Hafner, S. Tagliapietra, L. de Strasser, M. Hafner, S. Tagliapietra, and L. de Strasser, "The role of hydrocarbons in Africa's energy mix," in *Energy in Africa*, pp. 23–45, Springer, Cham, 2018.
- [2] D. Civitarese, D. Szwarcman, R. M. D. Silva, and E. V. Brazil, "Seismic facies segmentation using deep learning," *AAPG Annual and Exhibition*, 2018.
- [3] D. Cristea, <sup>PS</sup> *The use of seismic data in analyzing offshore reservoirs in the Black Sea*, 2018.
- [4] D. Civitarese, D. Szwarcman, E. V. Brazil, and B. Zadrozny, "Semantic segmentation of seismic images," 2019, <http://arxiv.org/abs/1905.04307>.
- [5] A. Voulodimos, N. Doulamis, A. Doulamis, and E. Protopapadakis, "Deep learning for computer vision: a brief review," *Computational Intelligence and Neuroscience*, vol. 2018, pp. 1–13, 2018.
- [6] N. Aloysius and M. Geetha, "A review on deep convolutional neural networks," *2017 international conference on communication and signal processing*, 2018, Chennai, India, April 2017, 2018.
- [7] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: a deep convolutional encoder-decoder architecture for image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.
- [8] M. Liu, W. Li, M. Jervis, and P. Nivlet, "3D seismic facies classification using convolutional neural network and semi-supervised generative adversarial network," *SEG Technical Program Expanded Abstracts 2019. Society of Exploration Geophysicists*, pp. 4995–4999, 2019.
- [9] T. Zhao, "Seismic facies classification using different deep convolutional neural networks," *SEG Technical Program Expanded Abstracts 2018. Society of Exploration Geophysicists*, pp. 2046–2050, 2018.
- [10] H. Di, Z. Wang, and G. AlRegib, "Real-time seismic-image interpretation via deconvolutional neural network," *SEG Technical Program Expanded Abstracts 2018. Society of Exploration Geophysicists*, pp. 2051–2055, 2018.
- [11] C. R. Ildstad and P. Bormann, "MalenoV: tool for training and classifying SEG Y seismic facies using deep neural networks," 2017, <https://github.com/bolgebrygg>.
- [12] Y. Alaudah, P. Michałowicz, M. Alfarraj, and G. AlRegib, "A machine-learning benchmark for facies classification," *Interpretation*, vol. 7, no. 3, pp. SE175–SE187, 2019.
- [13] B. West, S. May, J. Eastwood, and C. Rossen, "Interactive seismic facies classification of stack and avo data using textural attributes and neural networks," *2001 SEG Annual Meeting*, vol. 21, pp. 1042–1049, 2001.
- [14] D. Gao, "Latest developments in seismic texture analysis for subsurface structure, facies, and reservoir characterization: a review," *Geophysics*, vol. 76, no. 2, pp. W1–W13, 2011.
- [15] M. A. Shafiq, Y. Alaudah, G. AlRegib, and M. Deriche, "Phase congruency for image understanding with applications in computational seismic interpretation," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing*, New Orleans, LA, USA, March 2017.
- [16] A. Amin and M. Deriche, "Salt-dome detection using a codebook-based learning model," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 11, pp. 1636–1640, 2016.
- [17] C. Song, Z. Liu, H. Cai, Y. Wang, X. Li, and G. Hu, "Unsupervised seismic facies analysis with spatial constraints using regularized fuzzy c-means," *Journal of Geophysics and Engineering*, vol. 14, no. 6, pp. 1535–1543, 2017.
- [18] A. B. Mattos, R. S. Ferreira, R. M. D. G. Silva, M. Riva, and E. V. Brazil, "Assessing texture descriptors for seismic image retrieval," in *2017 30th SIBGRAPI Conference on Graphics, Patterns and Images*, Niteroi, Brazil, Oct. 2017.
- [19] P. Guillen, G. Larrazabal, G. González, D. Bumber, and R. Vilalta, "Supervised learning to detect salt body," *SEG*

- Technical Program Expanded Abstracts*, vol. 34, pp. 1826–1829, 2015.
- [20] T. Wrona, I. Pan, R. L. Gawthorpe, and H. Fossen, “Seismic facies analysis using machine learning,” *Geophysics*, vol. 83, no. 5, pp. O83–O95, 2018.
- [21] T. Zhao, V. Jayaram, A. Roy, and K. J. Marfurt, “A comparison of classification techniques for seismic facies recognition,” *Interpretation*, vol. 3, no. 4, p. SAE29–SAE58, 2015.
- [22] M. K. Dubois, G. C. Bohling, and S. Chakrabarti, “Comparison of four approaches to a rock facies classification problem,” *Computers and Geosciences*, vol. 33, no. 5, pp. 599–617, 2007.
- [23] T. Coléou, M. Poupon, and K. Azbel, “Unsupervised seismic facies classification: a review and comparison of techniques and implementation,” *Leading Edge (Tulsa, OK)*, vol. 22, no. 10, pp. 942–953, 2003.
- [24] M. M. Saggaf, M. N. Toksöz, and M. I. Marhoon, “Seismic facies classification and identification by competitive neural networks,” *Geophysics*, vol. 68, no. 6, pp. 1984–1999, 2003.
- [25] M. C. de Matos, P. L. Osorio, and P. R. Johann, “Unsupervised seismic facies analysis using wavelet transform and self-organizing maps,” *Geophysics*, vol. 72, pp. 9–21, 2007.
- [26] F. Qian, M. Yin, X. Y. Liu, Y. J. Wang, C. Lu, and G. M. Hu, “Unsupervised seismic facies analysis via deep convolutional autoencoders,” *Geophysics*, vol. 83, no. 3, pp. A39–A43, 2018.
- [27] M. A. Shafiq, M. Prabhushankar, H. Di, and G. AlRegib, “Towards understanding common features between natural and seismic images,” *SEG Technical Program Expanded Abstracts*, pp. 2076–2080, 2019.
- [28] A. U. Waldeland, A. C. Jensen, L. J. Gelius, and A. H. S. Solberg, “Convolutional neural networks for automated seismic interpretation,” *The Leading Edge*, vol. 37, no. 7, pp. 529–537, 2018.
- [29] J. S. Dramsch and M. Lüthje, “Deep learning seismic facies on state-of-the-art CNN architectures,” *Seg Technical Program Expanded Abstracts*, pp. 2036–2040, 2018.
- [30] W. Xiong, X. Ji, Y. Ma et al., “Seismic fault detection with convolutional neural network,” *Geophysics*, vol. 83, no. 5, pp. O97–O103, 2018.
- [31] E. Shelhamer, J. Long, and T. Darrell, “Fully convolutional networks for semantic segmentation,” *Ieee Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 640–651, 2017.
- [32] Z. Wang, F. Li, T. R. Taha, and H. R. Arabnia, “Improved automating seismic facies analysis using deep dilated attention autoencoders,” in *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019.
- [33] Y. Babakhin, A. Sanakoyeu, and H. Kitamura, “Semi-supervised segmentation of salt bodies in seismic images using an ensemble of convolutional neural networks,” *German Conference on Pattern Recognition*, Cham, 2019Springer, 2019.
- [34] J. Ma, F. Yang, and W. Wang, “Automatic salt detection with machine learning,” *80th EAGE Conference and Exhibition*, vol. 2018, pp. 1–5, 2018.
- [35] Y. Guo, Y. Liu, T. Georgiou, and M. S. Lew, “A review of semantic segmentation using deep neural networks,” *International journal of multimedia information retrieval*, vol. 7, no. 2, pp. 87–93, 2018.
- [36] H. Noh, S. Hong, and B. Han, “Learning deconvolution network for semantic segmentation,” in *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 1520–1528, Santiago, Chile, 2015.
- [37] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, “DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, 2018.
- [38] Y. Zhang, Y. Liu, H. Zhang, and H. Xue, “Seismic facies analysis based on deep learning,” *IEEE Geoscience and Remote Sensing Letters*, vol. 17, no. 7, pp. 1119–1123, 2020.
- [39] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: convolutional networks for biomedical image segmentation,” in *In International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241, Springer, Cham, 2015.
- [40] H. Wu and B. Zhang, “A deep convolutional encoder-decoder neural network in assisting seismic horizon tracking,” 2018, <http://arxiv.org/abs/1804.06814>.
- [41] F. Li, H. Zhou, Z. Wang, and X. Wu, “ADDCNN: an attention-based deep dilated convolutional neural network for seismic facies analysis with interpretable spatial-spectral maps,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 2, pp. 1733–1744, 2021.
- [42] M. Quamer Nasim, T. Maiti, A. Srivastava, T. Singh, and J. Mei, “Seismic facies analysis: a deep domain adaptation approach,” *IEEE Transactions on Geoscience and Remote Sensing*, p. 1, 2022.
- [43] A. Mustafa and G. AlRegib, “Man-recon: manifold learning for reconstruction with deep autoencoder for smart seismic interpretation,” *IEEE International Conference on Image Processing (ICIP)*, pp. 2953–2957, Anchorage, AK, USA, Sept. 2021.