



Research paper

Deep convolutional neural network for automatic fault recognition from 3D seismic datasets[☆]Yu An ^{a,*}, Jiulin Guo ^b, Qing Ye ^c, Conrad Childs ^d, John Walsh ^{d,e}, Ruihai Dong ^{a,*}^a The Insight Centre for Data Analytics, School of Computer Science, University College Dublin, Dublin, Ireland^b C&C Reservoirs, Brunel House, Reading, United Kingdom^c Key Laboratory of Tectonics and Petroleum Resources, Ministry of Education, China University of Geosciences, Wuhan 430074, China^d Fault Analysis Group, School of Earth Sciences, University College Dublin, Dublin, Ireland^e iCRAG (Irish Centre for Research in Applied Geosciences), Ireland

ARTICLE INFO

Keywords:

Fault recognition
Seismic interpretation
Deep learning
Computer vision
Image processing

ABSTRACT

With the explosive growth in seismic data acquisition and the successful application of deep convolutional neural networks (DCNN) to various image processing tasks within multidisciplinary fields, many researchers have begun to research DCNN based automatic seismic interpretation techniques. Due to the vast number of parameters considered in deep neural networks, deep learning methods usually require a large amount of data for training. However, collecting a large number of expert interpretations is very time consuming, so related research usually uses synthetic datasets and ignores the practical problems of field datasets. In this paper, we open-source a multi-gigabyte expert-labelled field dataset in response to the challenge of accessing large-scale expert-labelled field datasets. We show that 2D fault recognition within this dataset is an image segmentation or edge detection problem in the computer vision field, that can be expressed as a pixel-level fault/non-fault binary classification. Both types of DCNNs are compared, and we propose a novel fault recognition workflow, which involves processing and screening of seismic images and labels, training DCNNs and automatic numerical evaluation. We have also demonstrated for three case study datasets that effective image augmentation methods can reduce the required labelled crosslines while maintaining satisfactory performance. Our experimental results show that our workflow not only outperforms two state-of-the-art DCNN solutions but also achieves performance comparable to humans on an expert labelled image dataset, even predicting subtle faults that an expert interpreter did not annotate. We suggest that the proposed workflow could reduce the fault interpretation life cycle from months to hours and improve the quality, and define the confidence, of fault interpretation results.

1. Introduction

Faults are prominent geological features formed in the upper part of the earth's crust due to brittle deformation (Fossen, 2010). The mapping of faults can be a critical issue in predicting the distribution and size of natural resources; or mitigating risks associated with geohazards (Richards et al., 2015; Fossen, 2010). Mapping was traditionally achieved by studying faults in natural outcrops or mines (Lisle, 2004). Since many study areas are either geographically remote or poorly exposed, remote-sensing and borehole (i.e. well) data have become increasingly used in fault mapping (Csillag and Stogicza, 1987). The most significant recent advance in fault mapping has been the routine availability of 3D seismic reflection surveys that provide detailed images of large volumes of rock and the often complex 3-D networks

of faults within them. A seismic reflection survey provides an image of the subsurface generated due to density contrasts between rock layers so that the interfaces between layers of different density generate continuous reflections. Faults, while not explicitly imaged, are recognized because they cause discontinuities in these otherwise continuous reflections. A significant part of the subsurface interpretation from seismic data involves the manual digitization of faults defined by lines along which several reflections are offset (Robein, 2010; Biondi et al., 2007).

Conventionally, the interpretation of faults from seismic reflection data is a manual process that relies on data quality and experience of interpreters (Alcalde et al., 2017; Gibson et al., 2012). Fault mapping from good quality seismic data is relatively objective but can be both

[☆] Code: <https://github.com/anyuzoey/CNNforFaultInterpretation>.

* Corresponding authors.

E-mail addresses: yu.an@insight-centre.org (Y. An), ruihai.dong@insight-centre.org (R. Dong).

time-consuming and manually repetitive, with the interpretation of a single dataset typically taking weeks to months to complete (Gibson et al., 2012). As most seismic datasets, even good quality ones, have signal disturbance zones, particularly in heavily faulted areas, discontinuities in seismic reflectors can be poorly resolved, resulting in imprecise location or misinterpretation (of faults) (Badley et al., 1991; Iacopini et al., 2016; Alcalde et al., 2019). In these circumstances, interpretation results will have a higher degree of uncertainty which is often overlooked and difficult to quantify.

Meanwhile, the rapid growth in seismic data acquisition has created Big Data challenges for many oil & gas companies and research groups (Mohammadpoor and Torabi, 2018). Associated increasing demands for automatic/semi-automatic seismic fault interpretation reflect geologists and geophysicists' requirements within the oil and gas industry in particular. In some early work, researchers were mainly focused on creating and utilizing several derivative volumes and attributes to supplement the standard fault interpretation workflow, including dip/azimuth, coherence (Marfurt et al., 1999), curvature (Roberts, 2001), variance (Silva et al., 2005) and semblance (Marfurt et al., 1998). With improved computer capabilities, many computer-aided semi-automatic solutions were proposed which use simple feature analysis techniques, such as Ant Tracking (Silva et al., 2005) and Hough transform (Wang and AlRegib, 2014). However, because most of those features/methods are sensitive to noise (Cohen et al., 2006; Marfurt et al., 1998; Silva et al., 2005; Yan et al., 2019), seismic datasets often need to be preprocessed to a certain signal-to-noise level (Yuan et al., 2019), which in view of the unique geological structure and data quality of different datasets, it is usually necessary to manually adjust parameters by trial and error.

The above methods manually select one or a small selection of features to assist fault interpretation, a decision which can be highly subjective. Machine learning methods involving multiple features, such as support vector machine (SVM) and multilayer perceptron (MLP) therefore have been investigated to achieve better interpretation results (Di et al., 2017; Guitton et al., 2017; Di et al., 2019; Kortström et al., 2016). Although associated multi-feature methods reduce manual intervention to a certain extent, continued developments are required to improve accuracy, efficiency and user-friendliness. Over the last decade, DCNNs have emerged as an approach that can outperform humans on many computer vision tasks, as illustrated by the well-known ImageNet large-scale visual recognition challenge (Russakovsky et al., 2014). In recent years, advances in deep learning algorithms in fault interpretation have shown signs of progress but are still in the early development stages. Since most of the research at this stage uses synthetic data, ignoring the complex geological structure and noise of the field data, manual interpretation of seismic data is still the industry's mainstream workflow and academia.

In this paper, we suggest that computer vision approaches can be successfully applied to the challenge of fault interpretation in seismic data. We first review some conventional automatic or semi-automatic seismic interpretation methods and state-of-the-art deep learning based methods in Section 2. We then propose a procedure for processing field datasets and interpretations in Section 3, and following a description of the leading edge computer vision neural networks and how those models are trained in Section 4, we analyse and evaluate model performance in Section 5. In Section 6, we present field examples for each model and discuss the merits of each from a practical perspective. Finally, in Section 7 we show comparative results with those of methods advanced in two recent studies (Wu et al., 2019b; Cunha et al., 2020).

2. Related work

2.1. Conventional methods

The basic concept in seismic fault interpretation is to identify patterns on seismic images that resemble brittle deformation features

observed in outcrops, in which geological layers are displaced relative to each other. It involves using established conceptual geological models of fault geometries and displacements for performing, and in some cases, validating the interpretation (Alcalde et al., 2019; Gibbs, 1984; Badley et al., 1991; Walsh et al., 1996). Faults are typically interpreted on 2-D cross-sectional (i.e. vertical) slices of a 3-D seismic volume by the alignment of reflector discontinuities that have broadly consistent sense and amount of displacement (e.g. James et al., 1994; Gibson et al., 2012; Camanni et al., 2019), although the use of map-based fault mapping has become popular (e.g. Giba et al., 2012). Faults that are interpreted on adjacent seismic lines that are judged to be from the same fault are assigned the same label so that the individual fault traces are correlated to define the geometry of a fault surface in three dimensions. Hundreds, if not thousands, of slices, may be interpreted to provide sufficient geometric constraints for creating smooth fault surfaces. Individual faults in good quality seismic datasets should, ideally, provide well-defined interpretations that are more data-driven than model-driven (Walsh et al., 1999). Even on good quality seismic data, with high signal-to-noise ratios, interpretation can be made more difficult by a paucity of reflectors within a given sequence and increasing structural complexity. Poor quality seismic data is of low signal-to-noise ratio, and faults are often difficult to interpret (Boult et al., 2016; Iacopini et al., 2016). In this scenario, fault surfaces can be inferred from displaced and differently dipping seismic reflectors, but uncertainty quantification of the position of inferred faults can be very subjective (Gibson et al., 2012; Alcalde et al., 2017). With a progressive degradation in seismic quality, interpretations become more model-driven, perhaps accounting for the regional setting and anticipated structural style within a study area, with the robustness of the interpretation depending on the validity and application of the conceptual model. Given the significant uncertainty attached to poor quality datasets, multiple interpretations have been recommended, even if they are anchored in one particular conceptual model (Alcalde et al., 2019). The application of quantitative fault mapping concepts, including displacement variations, can also better inform fault mapping exercises and help define interpretation uncertainty (Badley et al., 1991; Boult et al., 2016; Walsh et al., 1996), but are usually only used by specialist structural geologists. Because conventional workflows are very time-consuming and manually repetitive, faults are often mapped selectively to save time, and little effort is made to test alternative hypotheses. In practice time constraints in delivering alternative interpretations within a fast-paced working environment means that little account is taken of interpretation uncertainty, with a premium set on the timely completion, rather than quality, of the interpretation. There are, therefore, significant potential benefits arising from the use of techniques which increase both the quality and speed of interpretation.

Automated fault interpretation involves defining methods for processing the seismic image to derive maps of the likelihood of occurrence of a fault, screening these maps for false fault identification then fitting discrete lines or surfaces through the regions of high fault likelihood. Practical workflows to perform automated fault interpretation have been described in the literature (Hale, 2013; Wu and Hale, 2016; Wu and Zhu, 2017) and examples of successful applicants have been presented. However, these workflows are not yet incorporated into the standard software employed in the industry. A key element of a successful workflow for automated fault interpretation is achieving good quality maps of fault likelihood. Different approaches have been adopted, and these were initially based on attributes derived directly from the seismic data (Roberts, 2001; Di et al., 2018b; AlRegib et al., 2018) but in recent years, the principles of image analysis and artificial intelligence have been applied to this area (Di et al., 2018b; AlRegib et al., 2018; Lu et al., 2018). Nonetheless, most research is at an early stage, using synthetic data to train a deep learning model without comparing different models. In this paper, we explored and analysed the performance of four (two types) different deep learning image processing models on real data to contribute to the rapid development of automatic fault recognition practice.

2.2. Deep learning based methods

A limited number of recent publications have described approaches for incorporating deep learning techniques into automatic seismic interpretation workflows. The main advantage of deep-learning-based methods, e.g. DCNNs, is that they can automatically identify and delineate geological features and distinguish them from seismic noise and processing artefacts (Di et al., 2018b; AlRegib et al., 2018; Souza et al., 2019). Deep learning technology also benefits from the availability of high-performance computers, which provide an excellent basis for applying fault surface generation algorithms and reducing the interpretation life cycle from months to hours/minutes, depending on whether a trained neural network is used (Lu et al., 2018). For the problem of fault interpretation, Araya-Polo et al. (2017) proposed a novel workflow using a simple fully-connected neural network to interpret unmigrated synthetic seismic data but bypassing the migration step in the attempt to perform a 'real-time' fault detection during seismic acquisition. As the neural network was trained using limited synthetic fault labels, its performance in real datasets of various geological settings remains mostly unknown. By contrast, our workflow starts with processed seismic data reflecting the standard workflow carried out by geologists or geophysicists.

Many previous studies have developed methods underpinned by the notion that fault recognition is an image classification problem (e.g. Di et al., 2018a; Wu et al., 2019c; Pochet et al., 2019; Zhang et al., 2019; Cunha et al., 2020). Di et al. (2018a) checks whether the centre pixel belongs to a fault by padding the seed pixels to 31×31 pixel patches, and then performs binary image classification, while Wu et al. (2019c), Zhang et al. (2019) classify the synthesized input patches into 17 categories by checking if the fault passes through the centre pixel at an appropriate fault dip angle. However, a problem shared by these methods is that they only label one pixel at a time, introducing significant redundancy. Other studies treat fault recognition as an image segmentation task (Di, 2018; Wu et al., 2019b; Shengrong et al., 2019; Wu et al., 2019a, 2020) and apply a classic image segmentation network called U-Net (Ronneberger et al., 2015; Çiçek et al., 2016). Image segmentation divides an image into multiple regions (pixel sets). Di (2018) developed a CNN that automatically interprets 12 seismic patterns simultaneously. Owing to each class's pixels are highly imbalanced, the model does not perform well in recognizing patterns associated with faults. Wu et al. (2019b,a, 2020) train 3D U-Net using synthetic data and then test on synthetic and actual field data. Their 3D U-Net model performs better than conventional methods on synthetic data and field data based on an expert assessment, but because of the lack of densely labelled field datasets, they were unable to quantify the performance. In Section 7, we apply their method to the Thebe dataset and compare its performance with our workflow. We propose treating it as an edge detection task in addition to taking fault recognition as image classification and image segmentation tasks. Edge detection in computer vision is intended to identify discontinuities in digital images, a concept that is very similar to that used in manual fault interpretation.

Although CNN based methods have shown promising results on fault interpretation tasks, we have identified and summarized some common challenges. Firstly, open-source large-scaled expert-labelled field datasets are rarely available with all of the foregoing research either using synthetic data or a few annotated sections as training data. Consequently, the performance of deep learning models on actual field datasets still remains unknown. Secondly, artificial bias is inevitable. False or missing interpretations create deviations or noises, which will affect the performance of the model. Thirdly, no appropriate evaluation metrics are available. With expert assessment being the most common assessment method, the result of fault interpretation is difficult to evaluate without introducing human bias.

In this paper, we give some possible solutions to the above problems. First, we open-source a multi-gigabyte expert-labelled field dataset for use by researchers testing their algorithms. Secondly, we use

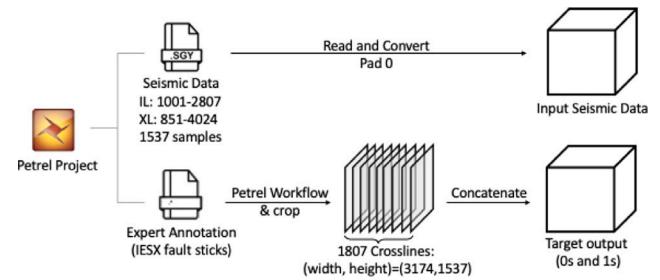


Fig. 1. The procedure for converting Petrel projects into processed datasets.

a filtering measure to minimize the negative effects of manual errors on the model. Finally, we introduce an automatic and comprehensive evaluation method (initially used by edge detection tasks) to evaluate the performance of various fault recognition algorithms.

3. Data preparation

The development of deep learning models generally involves dividing datasets into training, validation and test sets. A training set provides input to the model learning parameters; the validation set is used to fine-tune the model parameters; the test set is used to evaluate the model performance. The primary dataset used in this experiment is a public domain dataset from the Thebe Gas Field in the Exmouth Plateau of the Carnarvan Basin on the NW shelf of Australia. The faults were manually labelled by expert interpreters (Fault Analysis Group, University College Dublin) of faults from seismic data to investigate the structural style and associated evolution of the basin.

The dataset is in a numpy array format version of the seismic volume pixelated at the resolution of the seismic data to give a matrix $1803[\text{crossline}] \times 1537[\text{sample}] \times 3174[\text{inline}]$ (the first 4 out of 1807 crosslines are deleted due to lack of corresponding annotations) and normalized using min-max normalization. The corresponding fault interpretations were defined by binary images of each crossline at the same resolution as the seismic data that identified pixels with (1) or without (0) a fault. The processed dataset obtained via Fig. 1 can be accessed through this link.¹

Random splitting is the most common training and test splitting method used in machine learning. However, geological faults distributions on adjacent slices are almost identical so that random splitting is not appropriate as it is equivalent to peeking through the test set, which would result in an overly optimistic model. Thus, we divide the dataset into blocks by a ratio of approximately 5 : 1 : 4 to define the training, validation, and test set, respectively. More precisely, the first 900 pairs of crosslines representing the seismic data and the faults form the training set, the following 200 pairs are the validation set, and the last 703 pairs are the test set.

Geological interpretation of the Thebes dataset was focused on faults with vertical displacements greater than 20 m within a particular area of interest and depth range (ca 2 km to 4 km), generally ignoring shallower or deeper parts of the seismic volume. To avoid confusion caused by unlabelled segments and to limit the use of GPU memory resources, but at the same time obtain a sufficiently large training set, we used a slicing window algorithm with a stride of 48 pixels to crop 96×96 pixels patches. Patch pairs where the proportion of labelled pixels is less than 3% were filtered out. Reflective padding was introduced to get an integer number of patch pairs. Finally, we obtained 181 029 and 64 317 pairs of patches as the training set and the verification set, respectively.

¹ Processed Thebe dataset: https://drive.google.com/open?id=1RRtVhnAR2v39sbQ0_dbQhP2ghlFCBUU1.

4. Train and validate fault recognition models

Typical CNNs (such as the classic VGG network [Simonyan and Zisserman, 2014](#)) implement feature selection and prediction by combining a series of layers (convolutional layer, activation layer, pooling layer). The convolutional layer extracts suitable features by computing convolution between the learnable filter and input or output from the previous layer. The activation layer is responsible for introducing non-linear characteristics. The pooling layer performs down-sampling of features, merges features, and reduces computational complexity. Randomly initialized parameters, such as the learnable filter in the convolutional layer, are updated through an error feedback mechanism. This feedback mechanism optimizes the parameters by calculating the gradient of the difference between model predictions and labels, where the difference is obtained using a loss function. In the model training phase, in order to facilitate fast calculation, the training data is divided into small batches (here is 64) to train parameters. Traversing all the training sets once is called an epoch. Overfitting, in which a model loses its generalization ability, is a common problem in deep-learning model training. We use a validation set combined with an early-stopping-scheduler and the dropout layer to avoid the overfitting problem. We set 20 as the patience of the early-stopping-scheduler so that if the performance of the validation set does not improve for 20 epochs, the training process will stop and the best performance epoch will be saved. The dropout layer randomly deactivates some of the neurons in the network during training to simulate multiple parallel networks ([Srivastava et al., 2014](#)). The default dropout rate is 0.5. The training and validation sets then input to four selected DCNNs, two of which are used for image segmentation tasks and the rest are used for edge detection tasks.

4.1. Image segmentation models

In Kaggle's 2018 TGS salt recognition challenge,² image segmentation networks achieved the best performance in salt body interpretation. This challenge inspired us to interpret faults using image segmentation networks. We selected a widely used image segmentation network called U-Net as shown in [Fig. 2\(a\)](#) ([Ronneberger et al., 2015](#)). This model was built upon the typical DCNN, with multiple 3×3 convolution layers, 2×2 max-pooling layers and rectified linear unit (ReLU) activation layers. U-Net adds a symmetric extension path that uses 2D transposed convolution to gradually upsample the feature map back to the input dimension and then copy and concatenation shortcuts to expand location information to pinpoint the target object. For our small input patch size, we simplified the initial complex U-Net architecture by reducing some convolutional layers (similar to [Wu et al., 2019b](#)) and adding dropout layers and batch normalization steps for better generalization.

We also experiment with a state-of-the-art semantic image segmentation architecture called DeepLab ([Chen et al., 2014, 2016, 2017, 2018](#)). We applied the latest version DeepLabV3+³ to our dataset. Limited by GPU memory, we select MobileNetV2 ([Sandler et al., 2018; Howard et al., 2017](#)), a lightweight DCNN as a backbone network. The final network referred to as Mobile DeepLabV3+, see [Fig. 2\(b\)](#). Compared with U-Net, DeepLabV3+ uses atrous spatial pyramid pooling (ASPP) and a deeper network to gain richer contextual information, see [Fig. 2\(b\)](#). ASPP uses parallel multi-stride atrous convolutions (also known as dilated convolutions) to obtain context information at different scales.

A sigmoid layer was added at the bottom of the two segmentation networks to predict the fault probability (range from 0 to 1). The sigmoid function is illustrated by Eq. (1), where z is the original output.

The loss function for the two segmentation networks is binary cross-entropy, which is calculated from Eq. (2). l_n stands for loss of the n th pixel, w_n represents the weights for this pixel. \hat{y}_n stands for model prediction on pixel n while y_n stands for the target value (0 or 1). Here, w_n is equal to 1 for all pixels (n).

$$\phi(z) = \frac{1}{1 + e^{-z}} \quad (1)$$

$$l_n = -w_n [y_n \cdot \log \hat{y}_n + (1 - y_n) \cdot \log (1 - \hat{y}_n)] \quad (2)$$

4.2. Edge detection models

Polyline interpretations are very thin and are seen as abrupt variations in three-dimensional reflection seismic signals, which are similar to finding abrupt variations (i.e. edges) in images. Therefore, we suggest that the application of edge detection neural networks to the automatic interpretation of faults. A neural network designed by [Xie and Tu \(2015\)](#), referred to as the holistic edge detection (HED) network, shows performance close to human ability on a standard BSDS500 edge detection dataset. It is designed on the basis of the classic image classification network called VGG16. The last three fully connected layers of VGG16 was replaced by five up-sample layers, one convolution layer and one sigmoid layer. As shown in [Fig. 2\(c\)](#), the outputs from the last convolution in each stage are first up-sampled to the input dimensions to obtain five prediction maps that are then concatenated fused to produce the final prediction map. The differences between all six prediction maps and the labels are then used to update the parameters backwards. Weighted cross-entropy is used as the loss function to deal with the highly imbalanced edge/non-edge or fault/non-fault pixels. Here, weights (w_n) in Eq. (2) is calculated from Eq. (3), where Y stands for the number of pixels of the input image. $|Y_-|$ and $|Y_+|$ denote the fault and non-fault label sets, respectively. It is worth noting that weighted cross-entropy is used for the five side prediction maps, while normal cross-entropy is used for the fused prediction map. Since the last prediction map uses cross-entropy, its loss value is much smaller than the weighted cross-entropy, which results in the last layer not being well trained and providing an output that is not ideal. The penultimate prediction map is, therefore, selected as the output.

$$w_n = \begin{cases} |Y_+|/|Y|, & y_n = 0 \\ |Y_-|/|Y|, & y_n = 1 \end{cases} \quad (3)$$

[Liu et al. \(2016\)](#) later improved the HED architecture by using information generated from all convolutional layers instead of every last convolution layer and by using weighted cross-entropy for all prediction maps. [Fig. 2\(d\)](#) shows details of the proposed architecture to which we refer as RCF (Richer Convolution Features). Similarly, dropout layers and reducing the feature map for each layer, are used to reduce overfitting.

Adam optimizer, with an initial learning rate (lr) of 0.01, is applied in the two segmentation networks using SGD optimizer with an lr of 1e-6 and momentum of 0.9. The weight decay is set to be 0.0002 as suggested in the original paper on HED and RCF ([Xie and Tu, 2015; Liu et al., 2016](#)). If the validation set loss does not decrease for five consecutive epochs, the ReduceLROnPlateau scheduler will reduce the lr by a factor of 0.1. Considering reproducibility, we fix the random seed to 1 and set cudnn (NVIDIA CUDA® Deep Neural Network library) to the deterministic mode. All models are trained with one GeForce GTX 1080 Ti graphic card.

5. Evaluation

We propose a novel automatic process to evaluate the performance of different networks. This process not only reduces the workload of domain experts but also performs an objective evaluation as it evaluates multiple crosslines. The process first stitches overlapping tiles (patches), then selects the densely labelled area (800–1300 samples out of 1537 samples for Thebe dataset), and finally introduces the

² <https://www.kaggle.com/c/tgs-salt-identification-challenge>

³ <https://github.com/jfzhang95/pytorch-deeplab-xception>

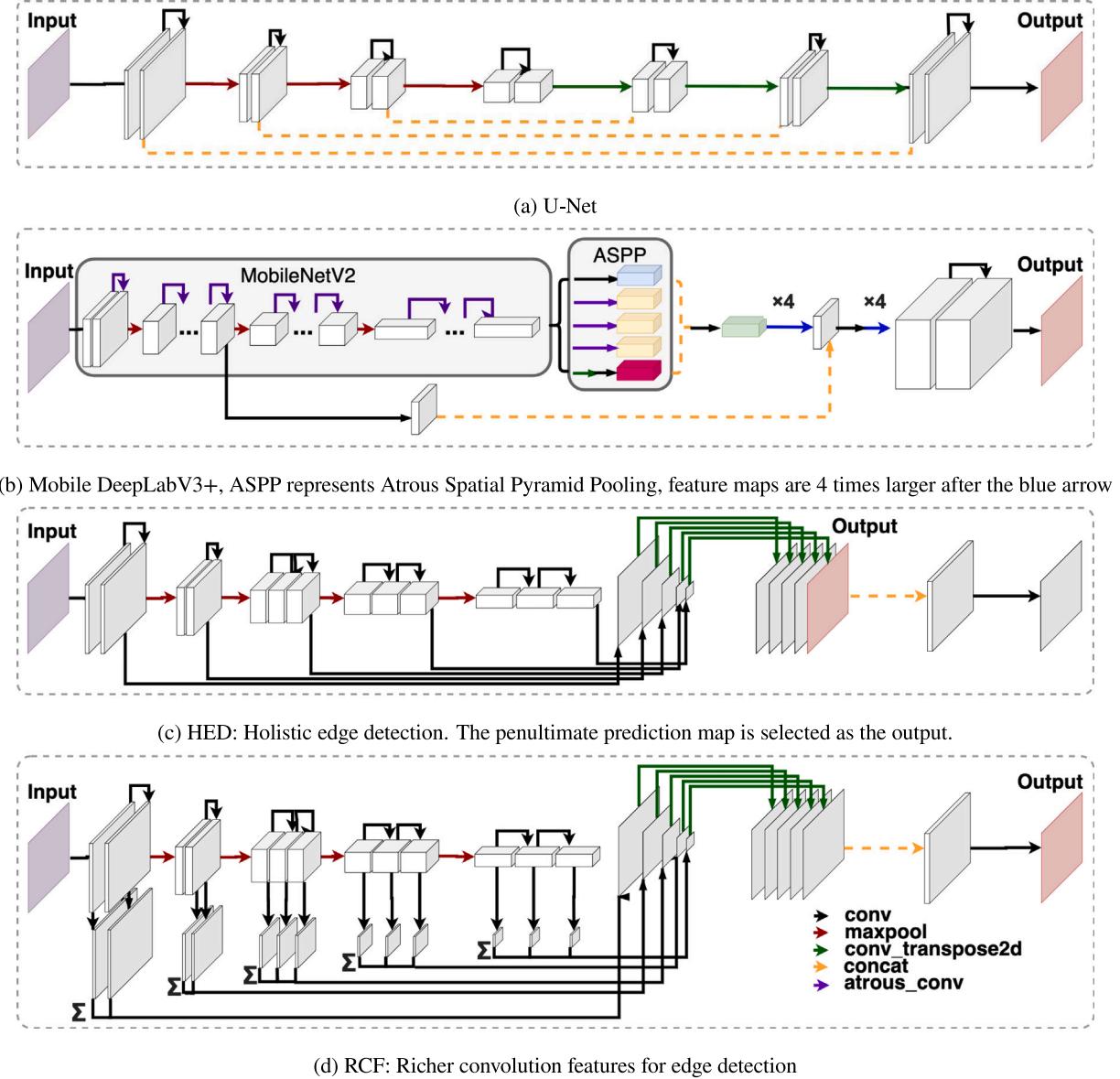


Fig. 2. CNN's Architecture, with different arrows representing the following: black arrow — standard 3×3 convolution layer, red arrow — 2×2 maxpooling layer, green arrow — transposed convolution layer, yellow arrow — copy and concatenate layer and purple arrow — atrous convolution layer.

Table 1
Fault confusion matrix.

Interpreter	Machine	
	Fault	NonFault
	TP	FN
NonFault	FP	TN

$$Precision = \frac{TP}{TP + FP} \quad (5)$$

$$Recall = \frac{TP}{TP + FN} \quad (6)$$

Given the similarity of faults to edges, we introduce the evaluation method of the classic edge detection BSDS500 dataset (Arbeláez et al., 2011). Three standard evaluation metrics are commonly applied: fixed contour threshold (ODS), per-image best threshold (OIS), and average precision (AP). The ODS and OIS calculate the F1 score by the best threshold of the entire dataset and the best threshold of each image, respectively. The F1 score is calculated using Eqs. (4)–(6). They are functions of the relative occurrence of four classification scenarios. As highlighted in Table 1, these scenarios are; true-positive (TP) pixels that both the machine and the interpreter consider to be fault pixels; true-negative (TN) pixels that the machine and the interpreter both predict to be non-fault pixels; false-positive (FP) pixels that machine considers to be fault pixels while the interpreter does not, and false negative (FN) pixels that the interpreter classifies as fault pixels while the machine does not.

AP gives the area under the precision-recall curve (AUPRC). As the model prediction of each pixel ranges from 0 to 1, with 99 thresholds,

evaluation methods commonly used in edge detection tasks to fault interpretation. Since the prediction quality decreases from the centre of the patch to the edge (Iglovikov et al., 2017), we directly average the overlapping area (Fig. 3(c)) or crop centre (Fig. 3(d)) results in mesh shape noise and discontinuity respectively, therefore introducing a smooth merge method (Fig. 3(e)) to our problem (Chevalier, 2017). This method uses a squared spine window function to merge adjacent patches smoothly.

$$F_1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (4)$$

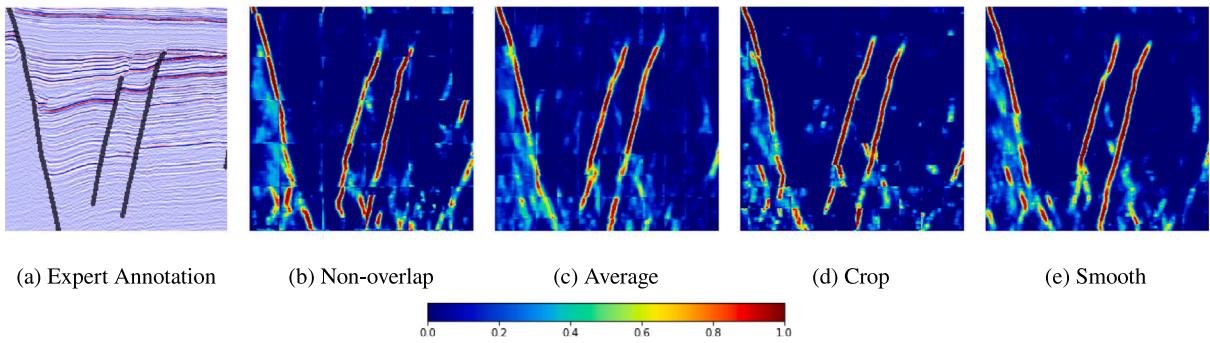


Fig. 3. Comparison of Merge Method. (a) Input seismic image with expert annotation (black line) (b) Non-overlap. Patch inputs are not overlapped. (c) Average merge. Predictions at overlap area are averaged. (d) Crop centre and then merge. Predictions at boundary area are cropped. (e) Smooth merge. Predictions at overlap area are weighted averaged. The colours indicate fault likelihood, in which the closer the closer it is to red, the more likely a fault passes through a pixel.

Table 2
Model performance comparison on test set.

ModelName	AP	ODS	OIS
U-Net	0.757	0.766	0.769
Mobile DeepLabV3+	0.784	0.756	0.759
RCF	0.794	0.800	0.806
HED	0.823	0.806	0.811

we can get a comprehensive illustration of model performance irrespective of any particular threshold. The maximum values for these three metrics are 1. They are much more appropriate for highly imbalanced datasets like fault interpretation datasets than other standard accuracy metrics. For example, if the model predicts there is no fault on the entire seismic crossline, it can still get over 90% accuracy. Considering the trade-off between accuracy and time cost, we sample the test set every five crosslines and resizing inputs to one-third of the original size. As Table 2 illustrates, all four models performed well on the test set. Of these models, HED performed best on all three evaluation metrics, showing the AP value of 0.823, the ODS of 0.806, and the OIS of 0.811. A test set example is provided and discussed in Section 6.

5.1. Image augmentation

Image augmentation methods are widely used in image processing tasks to increase the volume of training data by generating derived data from the original data through the application of some geometric manipulation. Recent work by Guillon et al. (2019) also proved that data augmentation could effectively reduce training set bias for automatic fault detection. Here, we selected ten augmentation methods from the work of Buslaev et al. (2018): horizontal flip, vertical flip, horizontal and vertical flip, optical distortion, elastic transform, grid distortion, sharpen, emboss, random brightness contrast and shift scale rotate. We designed a controlled experiment on the best model HED by doubling the training dataset using only one augmentation method at a time. The rest of the code remains the same. The control group, referred to as noaug, uses only the original training set. The comparison results are shown in Table 3 and indicate that under the given random seed, eight of the ten augmentation methods improve the performance of the vanilla HED model, of which the evaluation results (i.e. all three evaluation metrics) are substantially improved using vertical flip, emboss and grid distortion methods. Among them, the elastic transform method, which obtains the highest score in 2 out of 3 evaluation metrics, shows great potential in fault interpretation.

Since fault interpretation to generate a training dataset is still a time-consuming process, there is a requirement to reduce the size of the training set so that the proposed process can be transferred to other datasets relatively easily. Thus, we performed another control experiment by selecting a subset of the training set and validating

Table 3
Augmentation method performance.

ModelName	AP	ODS	OIS
noaug	0.823	0.806	0.811
VerticalFlip	0.872	0.823	0.827
IAAEmboss	0.861	0.812	0.816
GridDistortion	0.853	0.808	0.813
OpticalDistortion	0.839	0.802	0.806
RandomBrightnessContrast	0.823	0.814	0.819
ElasticTransform	0.801	0.842	0.847
Horizontal&VerticalFlip	0.787	0.812	0.817
ShiftScaleRotate	0.776	0.838	0.842

Table 4
Augmentation method performance various step size.

StepSize	AP	ODS	OIS
1	0.823	0.806	0.811
10	0.813	0.836	0.841
50	0.842	0.797	0.803
100	0.803	0.791	0.795

set with a larger step size. The required manual labelled training set can be reduced to one step size of the original size, while the rest of the training set is filled by augmented data generated using all ten augmentation methods, with the augmentation method randomly chosen with equal probability. If the step size is 10, we build a new training set and validation set by selecting every ten crosslines of the original training set and validation set, respectively. A total of 90 original crosslines and 810 augmented crosslines are set for training, with 20 crosslines for validating and test set remaining unchanged. Three step sizes are used, and Table 4 shows that the model performance increases first and then decreases as the step length increases. The step size of 10 performs better on 2 out of 3 metrics than the model using all training data. These results show that data augmentation can indeed improve model robustness and also demonstrates that we can achieve satisfactory performance with less than 1% of the data interpreted by the expert.

6. Performance of 2D fault recognition model

In this section we provide test examples on the Thebe dataset (Figs 4 & 5) and a UK dataset (Fig. 6) the Beatrice oil field from the Inner Moray Firth, Scotland (Walsh et al., 2003). For this exercise, the Beatrice dataset was an unlabelled (i.e. uninterpreted) dataset used to view the generalization capabilities of the model. In all cases, expert interpretations are shown on input seismic images (as black lines) to permit comparison of the relative performance of expert and machine (Figs 4 & 5 & 6(a)). Model predictions are expressed using probability maps, where each pixel has a range of 0 to 1, in which the closer the

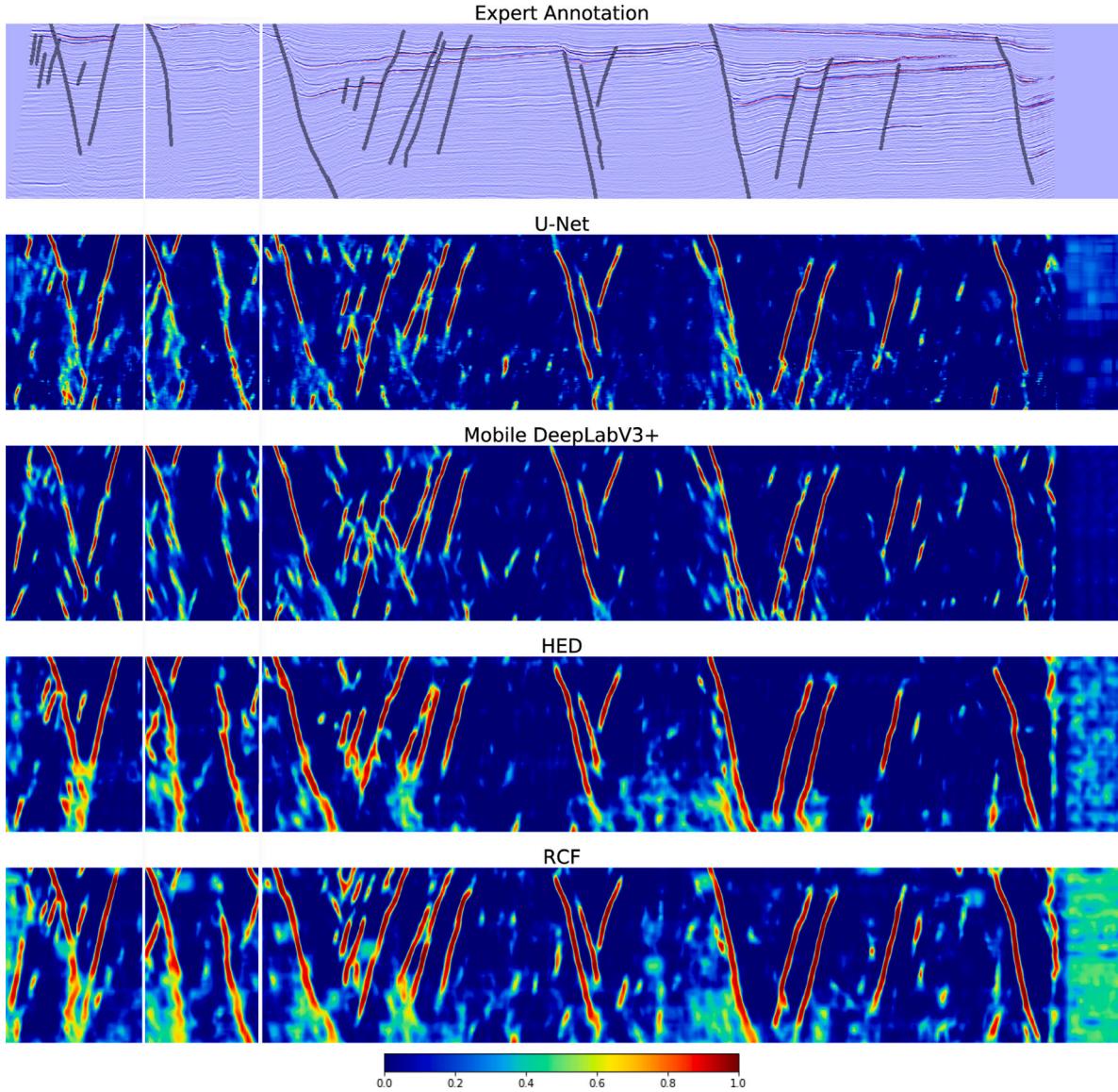


Fig. 4. The comparison of the models on the 100th crossline of Thebe test set shows the probabilistic images predicted by each of the four main models, where closer to 1 indicates that a fault is more likely to pass a pixel.

predicted value is to 1 (the closer it is to red), the more likely a fault passes through a pixel. Interpretation performance for all four models is good and are even comparable to those of expert interpreters.

Fig. 4 shows the output for a selection of methods for a seismic line from the Thebe dataset. Comparison with the expert interpretation is very encouraging though there are significant discrepancies between the expert interpretations and the machine's predictions in some areas (see the white box on Fig. 4). Later examination confirmed that the machine's predictions were correct and the interpreter neglected to interpret a fault, an outcome demonstrating the utility of our proposed workflow. Furthermore, because the output of the model is fault likelihood, geologists can adjust the details according to different datasets or different sections through threshold adjustment bars to achieve more accurate interpretation. The predictions of U-Net and Mobile DeepLabV3+ for the Thebe dataset (Fig. 4) provide more but relatively discontinuous faults, while the HED and RCF models contain fewer but more continuous faults because of the weighted cross-entropy loss function. There are places where the model RCF predictions may appear too aggressive, such as the artefact at the right-hand end of the seismic line where pixels with zero or noise are labelled as faults. The RCF and HED provide the most continuous fault interpretations which are closer

to the expert interpretation. Expert interpreters join together what appear to be discrete offsets of occasional reflectors along the length of a fault on a seismic line because these are often-times an artefact of the seismic technique which does not image the fault, but instead images offset reflectors. In that sense, the more continuous faults are the best outcome for the Thebe dataset, while the more discontinuous, apparently segmented, the response of other models is to some extent reminiscent of the nature of conventional seismic attributes (such as coherence, variance) that are strongly tied to individual reflectors. Our methods, HED and RCF, are therefore an advance on conventional approaches because they better define continuous faults rather than discontinuous signals or attributes. Fig. 5 shows that although the interpretation performance of a model declines with increasing step size, it still performs very well on the test set, an outcome which suggests that geologists only need to manually annotate less than 1% of a new dataset to train a deep learning model that will perform well.

The performance of the proposed workflow on Beatrice dataset is shown in Fig. 6, in which Fig. 6(a) displays the input seismic image. Surprisingly, all four models in Fig. 6(b) successfully identify most large-scale faults, together with a few small faults on the top right corner. As would be expected, the two segmentation networks (U-Net

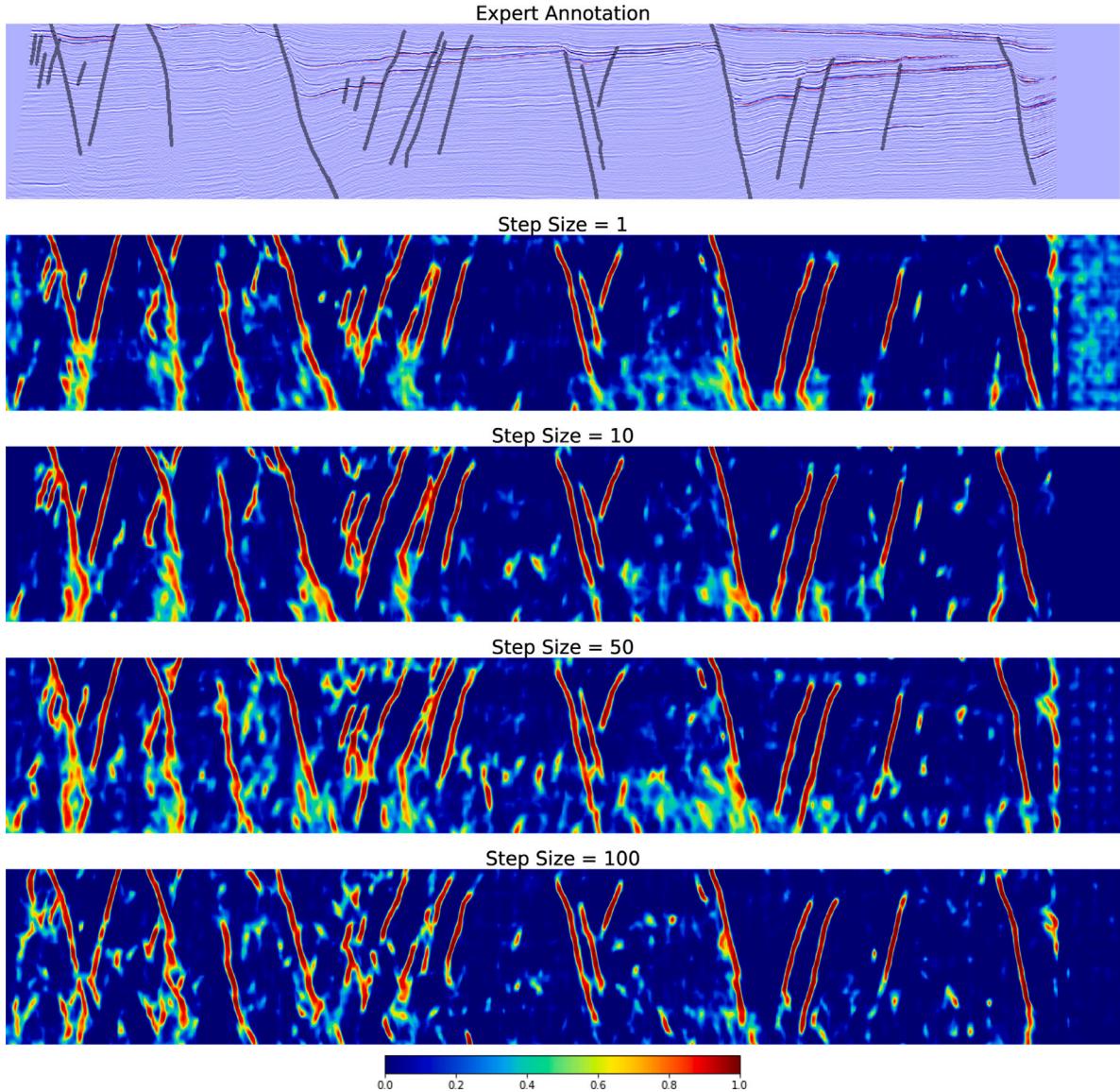


Fig. 5. Comparison of model HEDs using different step sizes on the 100th crossline of Thebe test set. The step size represents the selection interval when creating the training set and the validation set, where step size = 1 represents the best model HED in Fig. 4.

and Mobile DeepLabV3+) are sensitive to noise, leading to many small dots and line segments in the prediction map. All four models struggle with the migration noise within the deeper layers, which is not seen by the models during the training phase. Overall, the predictions from model HED and RCF are relatively less noisy than the other predictions, managing to highlight the main imageable faults.

For different augmentation methods, Fig. 6(c) shows that proper augmentation methods like ElasticTransform, VerticalFlip and Grid-Distortion, better define the two large-scale faults. Compared to the model trained with the original Thebe training data (step size = 1), the model trained with augmented data (step size = 10, 50, 100) in Fig. 6(d) performs better on the Beatrice dataset. This aspect proves that augmentation methods can improve the robustness of the model.

7. Comparison with 3D model and transfer-learning based models

In previous sections, we evaluated the performance of our workflow on different 2D models and demonstrated the effectiveness of data augmentation methods. In this section, we have devised two experiments from different perspectives to compare our workflow with two recently published approaches, a 3D model (Wu et al., 2019b) and a

transfer-learning based model (Cunha et al., 2020), to further validate its efficiency.

Wu et al. (2019b) proposed a 3D fault recognition solution, in which a 3D U-Net model is trained by synthetic seismic volumes. In our first experiment, we simply evaluate their original 3D model (trained by synthetic data, and named FaultSeg3D_synthetic in Table 5) on our Thebe dataset. As there are significant differences between synthetic data and the Thebe dataset, such as fault displacements and annotation thickness, “faultSeg3D_synthetic” predicted too many false positives (i.e. noisy dots), resulting in relatively low-performance scores. We then re-trained the FaultSeg3D model (named faultSeg3D_thebe in Table 5) with our Thebe dataset and performed a direct comparison by, for example, using the same 3% filter to filter out small blocks that have almost no faults and preparing patches/cubes from the same train, validate and test sets. The results show that faultSeg3D_thebe achieves slightly higher performance on ODS and OIS scores than our 2D methods without data augmentation (i.e. Table 2). This phenomenon might arise from FaultSeg3D embedding 3D data augmentation methods in its training process (to maintain the integrity of the faultSeg3D model, we do not remove its data augmentation function). However, comparison of their approach with our 2D method including data augmentation

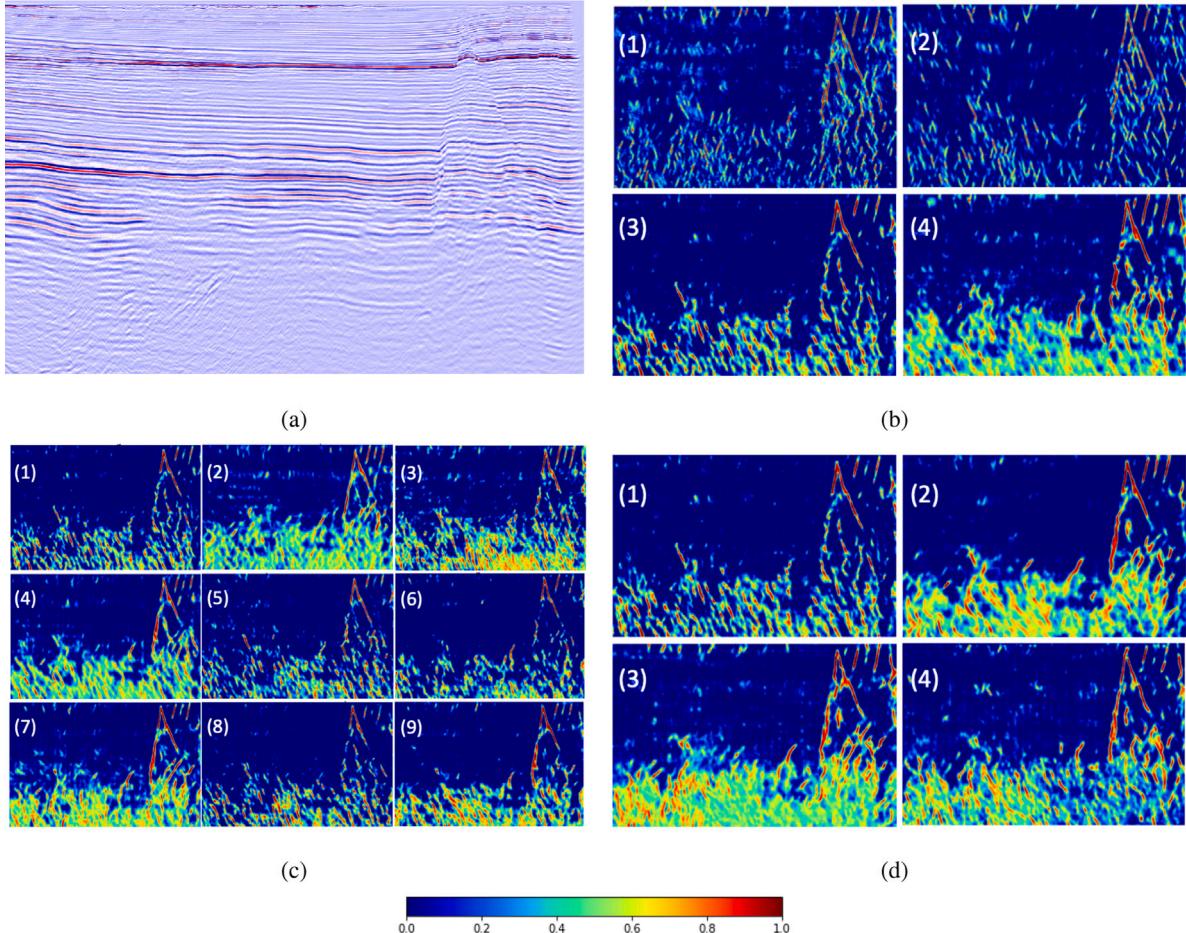


Fig. 6. Beatrice Crossline 800. (a) Input seismic Image (b) Model Performance Comparison (1) U-Net (2) Mobile DeepLabV3+ (3) HED (4) RCF. Models are trained using Thebe training set, which is same as Fig. 4(c) Augmentation Method vs Model Performance. Models are trained using augmented Thebe training set, which doubled the original thebe training set by a corresponding augmentation method. (1)-(9): noaug (Fig. 6(b)(3)), vertical flip, emboss, grid distortion, optical distortion, random brightness contrast, elastic transform, horizontal & vertical flip, shift scale rotate. (d) Step Size vs Model Performance. Models are trained using augmented Thebe training set, which selects a subset of original thebe training set by corresponding selection interval and fills it by augmented data. (1) Step size = 1 (Fig. 6(b)(3)) (2) Step size = 10 (3) Step size = 50 (4) Step size = 100.

Table 5
FaultSeg3D performance on the Thebe dataset.

ModelName	AP	ODS	OIS
faultSeg3D_synthetic	0.335	0.431	0.441
faultSeg3D_thebe	0.663	0.836	0.840

(i.e. ElasticTransform in Table 3, StepSize = 10 in Table 4), indicates that our workflow is able to achieve better performance on all three metrics.

We also evaluate the knowledge transfer capabilities of our work by comparison with the work of (Cunha et al., 2020), which harnesses transfer learning techniques to adapt their model to new seismic datasets. Cunha et al. (2020)'s workflow consists of two essential parts: 1, utilize Haralick texture features (Haralick et al., 1973) as coordinates to find the appropriate patch size with the feature attributes closest to the synthetic data; 2, use transfer learning techniques to adjust the pre-trained (using synthetic data) model, such as full fine-tuning (FFT), Feature extractor with MLP or SVM.

We use the slicing window method introduced above instead of complex clustering Haralick texture features to prepare data, and compare our method SlicingFFT (i.e. slicing window + full fine-tuning our tested models) with the best performance method HaralickFFT (i.e. Haralick + full fine-tuning their classification model) proposed by Cunha et al. (2020). For a fair comparison, we use the same

five sections of New Zealand Great South Basin (GSB) for training (i.e. crl2800) and testing (i.e. crl2600, crl3000, inl1791, inl2011). Instead of using precision, recall and AP metrics in this experiment, we use exactly the same evaluation metrics as Cunha et al. (2020)'s work, i.e. accuracy, sensitivity (also known as recall), specificity, F1, and area under receiver operating characteristic (AUROC). Specificity is calculated from Eq. (7), which focuses more on true negative (i.e. non-fault) pixels instead of true positive pixels in metrics precision (Eq. (5)). The y-axis of AUROC and AUPRC are the same (i.e. recall/sensitivity), but the x-axis is different. They are false-positive rate (i.e. 1-specificity) and precision respectively.

$$\text{Specificity} = \frac{TN}{TN + FP} \quad (7)$$

The evaluation results are presented in Table 6. We note that our workflow can greatly outperform the HaralickFFT method in many aspects, especially in sensitivity and F1 score. There is an exception in terms of specificity, with our models slightly lower than the HaralickFFT method. This may be caused by our models tending to predict more continuous fault interpretations, instead of a few line segments, and tend to predict potential faults. One visual comparison of crl2600 is shown in Fig. 7, while other sections can be seen in the supplementary material.

Due to their limited training data (only one crossline with a size of 76×484 pixels), it is not surprising that the U-Net model achieves better performance as it is designed for small data volumes. Among

Table 6

Comparison with Cunha et al. (2020) on GSB dataset. The number suffix (i.e. 9 or 99) of the model name indicates the number of times the augmented data is used.

Metrics	Cunha's	U-Net	DeepLab	HED	RCF	U-Net9	DeepLab9	HED99	RCF9
Accuracy	88.9%	+2.3%	+0.8%	-5.5%	-2.1%	+3.2%	+2.4%	+1.6%	+1.3%
Sensitivity	30.9%	+19.6%	+11.7%	+42.0%	+47.6%	+25.5%	+22.1%	+44.3%	+51.5%
Specificity	98.9%	-0.5%	-0.7%	-14.0%	-10.8%	-0.4%	-0.6%	-5.7%	-7.4%
F1	44.1%	+18.6%	+10.5%	+12.6%	+18.5%	+23.2%	+19.5%	+24.7%	+26.5%
AUROC	88.0%	-0.5%	-4.9%	-2.3%	+0.9%	+5.2%	+2.6%	+2.6%	+4.1%

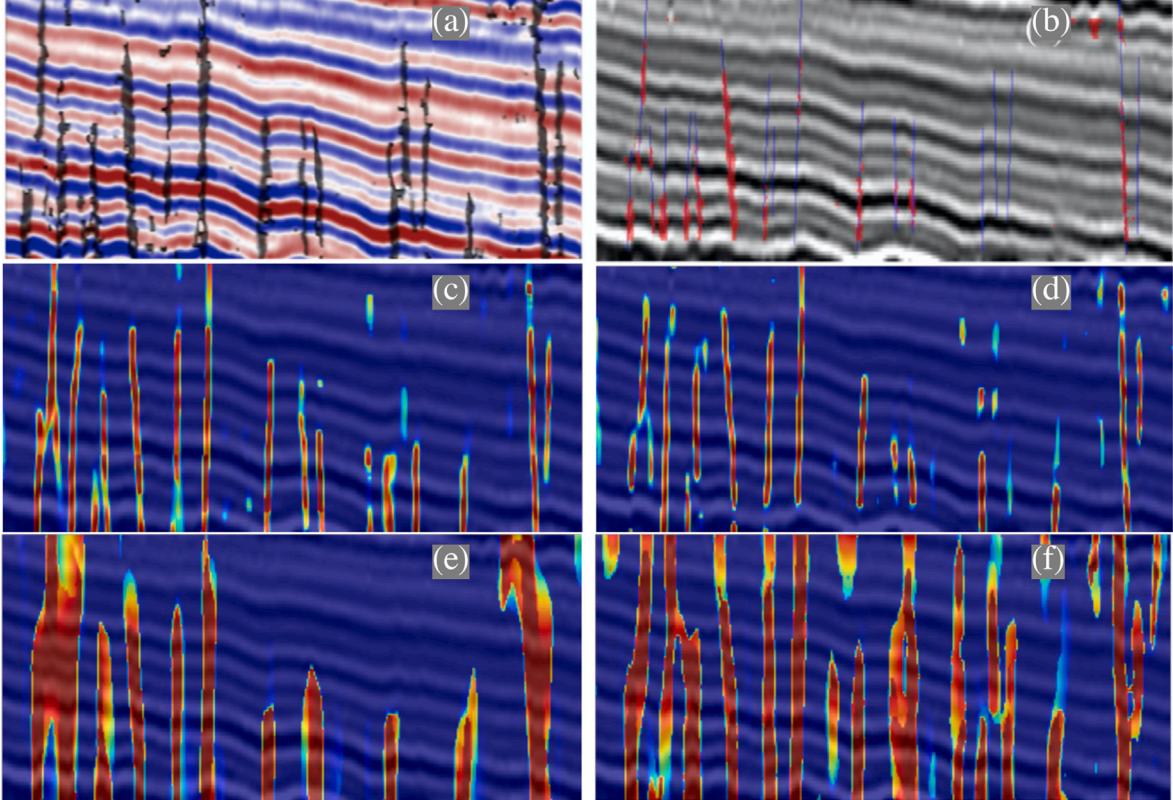


Fig. 7. Comparison in GSB croselines 2600, the blue lines are a thinner version of manual interpretation provided by Di et al. (2018a), Cunha et al. (2020) (a) best results of Di et al. (2018a) (b) best results of Cunha et al. (2020) and the predictions of our four tested models: U-Net9 (c), DeepLab9 (d), HED99 (e), RCF9 (f).

them, the HED method has the highest requirements for training data volume, but by using data augmentation methods, it can still achieve better performance than the HaralickFFT method in 4 out of 5 metrics. The results in Table 6 also validate the value of augmentation methods, indicating that the addition of augmentation provides improvements for all models.

8. Conclusion

Our work on the application of DCNN to the recognition of faults from seismic data has provided advances in three principal ways: 1. We open-sourced the Thebe dataset and its corresponding fault labels, which will promote cooperation between the fields of geology and computer science and help verify the performance of different algorithms. We also propose a method to convert geological project files to a data format suitable for deep learning. 2. We first introduced the edge detection networks for fault recognition. Compared with the commonly used image segmentation networks, the edge detection networks can predict more continuous faults instead of line segments, however, requiring more training data. We also experimentally demonstrated that proper augmentation could provide a more robust model and reduce the size of the required manually labelled training data. The automated recognition workflow suggested above outputs narrow zones of high

fault probability that accurately replicate manual fault interpretations and outperform the two state-of-the-art related works. If used with a fault discretization algorithm (Hale, 2013) or the newest version of PaleoScan, this automated mapping has the potential to rapidly generate accurate automated fault interpretations for input to standard industry workflows. 3. We propose a numerical evaluation method to automatically obtain objective and comprehensive test set performance that allows optimum workflows to be designed that maximize efficiency and performance. The workflow proposed uses image processing methods to perform 2D fault recognition. Prediction of each crossline can be concatenated together to form a 3D fault network for later comparison using voxel-based imaging techniques. In the future, we will investigate the following: 1) Improved model generalization by investigating proper transfer learning techniques; 2) 3D fault interpretation using 3D neural networks (e.g. 3D U-Net) or CNN in conjunction with time-series neural networks; 3) Noise reduction using super-resolution deep learning networks (e.g. Yuan et al., 2019; Lu et al., 2018) 4) Unbiased dataset collection by inviting multi interpreters (experts/students) to annotate the same dataset. With the proposed workflow, we plan to build a library providing annotated datasets for areas with different geological and fault system characteristics. In this way, a more geologically robust fault network (of a new dataset) can be built using deep learning models pre-trained on existing datasets.

9. Computer code availability

Name: CNNforFaultInterpretation
 Developer: Yu An, Ruihai Dong
 Telephone: +353 83 379 7994
 E-mail: yu.an@insight-centre.org
 Year first available: 2020
 hardware required: GPU
 software required: none
 program language: Python
 program size: 582M
 Source code: <https://github.com/anyuzoey/CNNforFaultInterpretation>

CRediT authorship contribution statement

Yu An: Conceptualization, Implementation, Methodology, Data acquisition, Analysis, Writing. **Jiulin Guo:** Conceptualization, Data acquisition, Analysis, Writing. **Qing Ye:** Data acquisition, Analysis, Dataset interpretation, Writing. **Conrad Childs:** Data acquisition, Analysis, Writing. **John Walsh:** Conceptualization, Data acquisition, Analysis, Writing. **Ruihai Dong:** Supervision, Conceptualization, Implementation, Methodology, Data acquisition, Analysis, Writing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

Thanks to Australia National Offshore Petroleum Information Management System (NOPIMS) for providing the original Thebe dataset. Thanks to Niall Reenan, Dr Marcus Carneiro, Dr Eoin Dunlevy and Luo Luo from UCD for providing help on datasets and Petrel software. This work was supported by Science Foundation Ireland (SFI) [SFI/12/RC/2289_P2, SFI/13/RC/2092]; the European Regional Development Fund; Beijing-Dublin International College Fund.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at <https://doi.org/10.1016/j.cageo.2021.104776>.

References

- Alcalde, J., Bond, C., Johnson, G., Ellis, J., Butler, R., 2017. Impact of seismic image quality on fault interpretation uncertainty. *GSA Today* 27 (2), 4–10. <http://dx.doi.org/10.1130/GSATG282A.1>.
- Alcalde, J., Bond, C.E., Johnson, G., Kloppenburg, A., Ferrer, O., Bell, R., Ayarza, P., 2019. Fault interpretation in seismic reflection data: an experiment analysing the impact of conceptual model anchoring and vertical exaggeration. *Solid Earth* 10 (5), 1651–1662. <http://dx.doi.org/10.5194/se-10-1651-2019>, URL: <https://www.solid-earth.net/10/1651/2019/>.
- AlRegib, G., Deriche, M., Long, Z., Di, H., Wang, Z., Alaudah, Y., Shafiq, M.A., Alfarraj, M., 2018. Subsurface structure analysis using computational interpretation and learning: A visual signal processing perspective. *IEEE Signal Process. Mag.* 35 (2), 82–98. <http://dx.doi.org/10.1109/MSP.2017.2785979>.
- Araya-Polo, M., Dahlke, T., Frogner, C., Zhang, C., Poggio, T., Hohl, D., 2017. Automated fault detection without seismic processing. *Lead. Edge* 36 (3), 208–214. <http://dx.doi.org/10.1190/le36030208.1>.
- Arbeláez, P., Maire, M., Fowlkes, C., Malik, J., 2011. Contour detection and hierarchical image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* 33 (5), 898–916. <http://dx.doi.org/10.1109/TPAMI.2010.161>.
- Badley, M., Freeman, B., Roberts, A., Thatcher, J., Walsh, J.J., Watterson, J., Yielding, G., 1991. Fault interpretation during seismic interpretation and reservoir evaluation. In: *The Integration of Geology, Geophysics, Petrophysics and Petroleum Engineering in Reservoir Delineation, Description and Management*. Proc. Conference. Houston, 1990, AAPG, pp. 224–241.
- Biondi, B., Society of Exploration Geophysicists, European Association of Geoscientists and Engineers, 2007. *Concepts and Applications in 3D Seismic Imaging : 2007 Distinguished Instructor Short Course*. Society of Exploration Geophysicists, p. 243.
- Boult, P., Freeman, B., Yielding, G., 2016. Structural interpretation of seismic geologic reality, perspective, and 3-D thinking. In: AAPG Memoir, Vol. 111. pp. 75–89. <http://dx.doi.org/10.1306/13561987M1113157>.
- Buslaev, A., Parinov, A., Khvedchenya, E., Iglovikov, V.I., Kalinin, A.A., 2018. Albuminations: fast and flexible image augmentations. [arXiv:1809.06839](https://arxiv.org/abs/1809.06839).
- Camanni, G., Roche, V., Childs, C., Manzocchi, T., Walsh, J.J., Conneally, J., Saqab, M., Delogkos, S., 2019. The three-dimensional geometry of relay zones within segmented normal faults. *J. Struct. Geol.* 103895. <http://dx.doi.org/10.1016/j.jsg.2019.103895>.
- Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L., 2014. Semantic image segmentation with deep convolutional nets and fully connected CRFs. [arXiv:1412.7062](https://arxiv.org/abs/1412.7062).
- Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L., 2016. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. [arXiv:1606.00915](https://arxiv.org/abs/1606.00915).
- Chen, L.-C., Papandreou, G., Schroff, F., Adam, H., 2017. Rethinking atrous convolution for semantic image segmentation. [arXiv:1706.05587](https://arxiv.org/abs/1706.05587).
- Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H., 2018. Encoder-decoder with atrous separable convolution for semantic image segmentation. [arXiv:1802.02611](https://arxiv.org/abs/1802.02611).
- Chevalier, G., 2017. Smoothly-Blend-Image-Patches. GitHub, URL: <https://github.com/Vooban/Smoothly-Blend-Image-Patches>.
- Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O., 2016. 3D UNet: Learning dense volumetric segmentation from sparse annotation. [arXiv:1606.06650](https://arxiv.org/abs/1606.06650).
- Cohen, I., Coul, N., Vassiliou, A.A., 2006. Detection and extraction of fault surfaces in 3D seismic data. *Geophysics* 71 (4), P21–P27. <http://dx.doi.org/10.1190/1.2215357>.
- Csillag, F., Stogicza, Á., 1987. Application of satellite imagery in tectonic interpretation. *J. Geodyn.* 8 (2), 205–219. [http://dx.doi.org/10.1016/0264-3707\(87\)90038-X](http://dx.doi.org/10.1016/0264-3707(87)90038-X), URL: <http://www.sciencedirect.com/science/article/pii/026437078790038X>, Recent Crustal Movements In The Carpatho-Balkan Region.
- Cunha, A., Pochet, A., Lopes, H., Gattass, M., 2020. Seismic fault detection in real data using transfer learning from a convolutional neural network pre-trained with synthetic seismic data. *Comput. Geosci.* 135 (October 2019), 104344. <http://dx.doi.org/10.1016/j.cageo.2019.104344>, URL: <https://linkinghub.elsevier.com/retrieve/pii/S0098300418307040>.
- Di, H., 2018. Developing a seismic pattern interpretation network (SpiNet) for automated seismic interpretation. [arXiv:1810.08517](https://arxiv.org/abs/1810.08517).
- Di, H., Shafiq, M.A., AlRegib, G., 2017. Seismic-fault detection based on multiattribute support vector machine analysis. In: SEG Technical Program Expanded Abstracts 2017. Society of Exploration Geophysicists, pp. 2039–2044. <http://dx.doi.org/10.1190/segam2017-17748277.1>, URL: <http://library.seg.org/doi/10.1190/segam2017-17748277.1>.
- Di, H., Shafiq, M.A., Wang, Z., AlRegib, G., 2019. Improving seismic fault detection by super-attribute-based classification. *Interpretation* 7 (3), SE251–SE267. <http://dx.doi.org/10.1190/INT-2018-0188.1>, URL: <https://library.seg.org/doi/10.1190/INT-2018-0188.1>.
- Di, H., Wang, Z., AlRegib, G., 2018a. Seismic fault detection from post-stack amplitude by convolutional neural networks. In: 80th EAGE Conference and Exhibition 2018. European Association of Geoscientists & Engineers, pp. 1–5.
- Di, H., Wang, Z., AlRegib, G., et al., 2018b. Why using CNN for seismic interpretation? An investigation. In: 2018 SEG International Exposition and Annual Meeting. Society of Exploration Geophysicists, pp. 2216–2220.
- Fossen, H., 2010. *Structural Geology*. Cambridge University Press, <http://dx.doi.org/10.1017/CBO9780511777806>.
- Giba, M., Walsh, J.J., Nicol, A., 2012. Segmentation and growth of an obliquely reactivated normal fault. *J. Struct. Geol.* 39, 253–267. <http://dx.doi.org/10.1016/j.jsg.2012.01.004>, URL: <http://www.sciencedirect.com/science/article/pii/S0191814112000132>.
- Gibbs, A.D., 1984. Structural evolution of extensional basin margins. *J. Geol. Soc.* 141 (4), 609–620. <http://dx.doi.org/10.1144/gsjgs.141.4.0609>, arXiv:<https://jgs.lyellcollection.org/content/141/4/609.full.pdf>.
- Gibson, D., Spann, M., Turner, J., Wright, T., Hale, D., 2012. Fault surfaces and fault throws from 3D seismic images. In: SEG Technical Program Expanded Abstracts 2012, Vol. 43, No. 9. pp. 2094–2102, URL: <http://library.seg.org/doi/abs/10.1190/segam2012-0734.1>.
- Guillon, S., Joncour, F., Goutorbe, P., Castanié, L., 2019. Reducing training dataset bias for automatic fault detection. In: SEG Technical Program Expanded Abstracts 2019. Society of Exploration Geophysicists, pp. 2423–2427. <http://dx.doi.org/10.1190/segam2019-3216557.1>, URL: <https://library.seg.org/doi/10.1190/segam2019-3216557.1>.
- Guitton, A., Wang, H., Trainor-Guitton, W., 2017. Statistical imaging of faults in 3D seismic volumes using a machine learning approach. In: SEG Technical Program Expanded Abstracts 2017. Society of Exploration Geophysicists, pp. 2045–2049. <http://dx.doi.org/10.1190/segam2017-17589633.1>, URL: <http://library.seg.org/doi/10.1190/segam2017-17589633.1>.
- Hale, D., 2013. Methods to compute fault images, extract fault surfaces, and estimate fault throws from 3D seismic images. *Geophysics* 78, 33. <http://dx.doi.org/10.1190/geo2012-0331.1>.

- Haralick, R., Shanmugam, K., Dinstein, I., 1973. Textural features for image classification. *IEEE Trans. Syst. Man Cybern. SMC-3* (6), 610–621. <http://dx.doi.org/10.1109/TSMC.1973.4309314>.
- Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H., 2017. MobileNets: Efficient convolutional neural networks for mobile vision applications. [arXiv:1704.04861](https://arxiv.org/abs/1704.04861).
- Iacopini, D., Butler, R.W., Purves, S., McArdle, N., De Freslon, N., 2016. Exploring the seismic expression of fault zones in 3D seismic volumes. *J. Struct. Geol.* 89, 54–73. <http://dx.doi.org/10.1016/j.jsg.2016.05.005>.
- Iglovikov, V., Mushinsky, S., Osin, V., 2017. Satellite imagery feature detection using deep convolutional neural network: A kaggle competition. [arXiv:1706.06169](https://arxiv.org/abs/1706.06169).
- James, H., Telles, M., Schaetzlein, G., Stark, T., 1994. Geophysical interpretation - from bits and bytes to the big picture. *Oilfield Rev.* 6 (3), 23–31.
- Kortström, J., Uski, M., Tiira, T., 2016. Automatic classification of seismic events within a regional seismograph network. *Comput. Geosci.* 87, 22–30. <http://dx.doi.org/10.1016/j.cageo.2015.11.006>, URL: [http://doi.org/10.1016/j.cageo.2015.11.006](https://doi.org/10.1016/j.cageo.2015.11.006)<https://linkinghub.elsevier.com/retrieve/pii/S0098300415300832>.
- Lisle, R., 2004. Geological Structures and Maps: A Practical Guide. p. 106.
- Liu, Y., Cheng, M.-M., Hu, X., Wang, K., Bai, X., 2016. Richer convolutional features for edge detection. [arXiv:1612.02103](https://arxiv.org/abs/1612.02103).
- Lu, P., Morris, M., Brazell, S., Comiskey, C., Xiao, Y., 2018. Using generative adversarial networks to improve deep-learning fault interpretation networks. *Lead. Edge* 37, 578–583. <http://dx.doi.org/10.1190/tle37080578.1>.
- Marfurt, K.J., Kirlin, R.L., Farmer, S.L., Bahorich, M.S., 1998. 3-D seismic attributes using a semblance-based coherency algorithm. *Geophysics* 63 (4), 1150–1165. <http://dx.doi.org/10.1190/1.1444415>.
- Marfurt, K.J., Sudhaker, V., Gersztenkorn, A., Crawford, K.D., Nissen, S.E., 1999. Coherency calculations in the presence of structural dip. *Geophysics* 64 (1), 104–111. <http://dx.doi.org/10.1190/1.1444508>.
- Mohammadpoor, M., Torabi, F., 2018. Big data analytics in oil and gas industry: An emerging trend. *Petroleum* <http://dx.doi.org/10.1016/j.petlm.2018.11.001>, URL: <http://www.sciencedirect.com/science/article/pii/S2405656118301421>.
- Pochet, A., Diniz, P.H.B., Lopes, H., Gattass, M., 2019. Seismic fault detection using convolutional neural networks trained on synthetic poststacked amplitude maps. *IEEE Geosci. Remote Sens. Lett.* 16 (3), 352–356. <http://dx.doi.org/10.1109/LGRS.2018.2875836>, URL: <https://ieeexplore.ieee.org/document/8527536/>.
- Richards, F.L., Richardson, N.J., Bond, C.E., Cowgill, M., 2015. Interpretational variability of structural traps: implications for exploration risk and volume uncertainty. *Geol. Soc. Lond. Spec. Publ.* 421 (1), 7–27. <http://dx.doi.org/10.1144/SP421.13>, [arXiv:https://sp.lyellcollection.org/content/421/1/7.full.pdf](https://sp.lyellcollection.org/content/421/1/7.full.pdf).
- Robein, E., 2010. Seismic Imaging: A Review of the Techniques, their Principles, Merits and Limitations (EET 4). EAGE, p. 244, URL: [https://bookshop.eage.org/product/seismic-imaging-a-review-of-the-techniques-their-principles-merits-and-limitations-eet-4/](https://bookshop.eage.org/product/seismic-imaging-a-review-of-the-techniques-their-principles-merits-and-limitations-eet-4).
- Roberts, A., 2001. Curvature attributes and their application to 3D interpreted horizons. *First Break* 19 (2), 85–100. <http://dx.doi.org/10.1046/j.0263-5046.2001.00142.x>, URL: <https://onlinelibrary.wiley.com/doi/abs/10.1046/j.0263-5046.2001.00142.x>.
- Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation. [arXiv:1505.04597](https://arxiv.org/abs/1505.04597).
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L., 2014. ImageNet large scale visual recognition challenge. [arXiv:1409.0575](https://arxiv.org/abs/1409.0575).
- Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.-C., 2018. MobileNetV2: Inverted residuals and linear bottlenecks. [arXiv:1801.04381](https://arxiv.org/abs/1801.04381).
- Shengrong, L., Changchun, Y., Hui, S., Zhang, H., 2019. Seismic fault detection using an encoder-decoder convolutional neural network with a small training set. *J. Geophys. Eng.* 16, <http://dx.doi.org/10.1093/jge/gxy015>.
- Silva, C., Marcolino, C., Lima, F., 2005. Automatic fault extraction using ant tracking algorithm in the marlim south field, campos basin. In: Seg Technical Program Expanded Abstracts, Vol. 24. <http://dx.doi.org/10.1190/1.2148294>.
- Simonyan, K., Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. [arXiv:1409.1556](https://arxiv.org/abs/1409.1556).
- Souza, J., Santos, M., Magalhães, R., Neto, E., Oliveira, G., Roque, W., 2019. Automatic classification of hydrocarbon “leads” in seismic images through artificial and convolutional neural networks. *Comput. Geosci.* 132 (June), 23–32. <http://dx.doi.org/10.1016/j.cageo.2019.07.002>, URL: <https://doi.org/10.1016/j.cageo.2019.07.002><https://linkinghub.elsevier.com/retrieve/pii/S0098300419300263>.
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* 15, 1929–1958, URL: <http://jmlr.org/papers/v15/srivastava14a.html>.
- Walsh, J.J., Childs, C., Imber, J., Manzocchi, T., Watterson, J., Nell, P., 2003. Strain localisation and population changes during fault system growth within the Inner Moray Firth, Northern North Sea. *J. Struct. Geol.* 25, 307–315. [http://dx.doi.org/10.1016/S0191-8141\(02\)00028-7](http://dx.doi.org/10.1016/S0191-8141(02)00028-7).
- Walsh, J.J., Watterson, J., Bailey, W., Childs, C., 1999. Fault relays, bends and branch-lines. *J. Struct. Geol.* 21, 1019–1026. [http://dx.doi.org/10.1016/S0191-8141\(99\)00026-7](http://dx.doi.org/10.1016/S0191-8141(99)00026-7).
- Walsh, J.J., Watterson, J., Childs, C., Nicol, A., 1996. Ductile strain effects in the analysis of seismic interpretations of normal fault systems. *Geol. Soc. Lond. Spec. Publ.* 99, 27–40. <http://dx.doi.org/10.1144/GSL.SP.1996.099.01.04>.
- Wang, Z., AlRegib, G., 2014. Automatic fault surface detection by using 3D hough transform. In: SEG Technical Program Expanded Abstracts, Vol. 33. pp. 1439–1444. <http://dx.doi.org/10.1190/segam2014-1590.1>.
- Wu, X., Geng, Z., Shi, Y., Pham, N., Fomel, S., 2019a. Building realistic structure models to train convolutional neural networks for seismic structural interpretation. In: SEG Technical Program Expanded Abstracts 2019. Society of Exploration Geophysicists, pp. 4745–4750. <http://dx.doi.org/10.1190/segam2019-3214282.1>, URL: <https://library.seg.org/doi/10.1190/segam2019-3214282.1>.
- Wu, X., Hale, D., 2016. 3D seismic image processing for faults. *Geophysics* 81, IM1–IM11. <http://dx.doi.org/10.1190/geo2015-0380.1>.
- Wu, X., Liang, L., Shi, Y., Fomel, S., 2019b. FaultSeg3D: Using synthetic data sets to train an end-to-end convolutional neural network for 3D seismic fault segmentation. *Geophysics* 84, IM35–IM45. <http://dx.doi.org/10.1190/geo2018-0646.1>.
- Wu, X., Liang, L., Shi, Y., Geng, Z., Fomel, S., 2020. Deep learning for local seismic image processing: Fault detection, structure-oriented smoothing with edge-preserving, and slope estimation by using a single convolutional neural network. In: SEG International Exposition and Annual Meeting 2019, No. August. pp. 2222–2226. <http://dx.doi.org/10.1190/segam2019-3215251.1>.
- Wu, X., Shi, Y., Fomel, S., Liang, L., 2019c. Convolutional neural networks for fault interpretation in seismic images. In: 2018 SEG International Exposition and Annual Meeting, No. August. SEG 2018, pp. 1946–1950. <http://dx.doi.org/10.1190/segam2018-2995341.1>.
- Wu, X., Zhu, Z., 2017. Methods to enhance seismic faults and construct fault surfaces. *Comput. Geosci.* 107 (May), 37–48. <http://dx.doi.org/10.1016/j.cageo.2017.06.015>, URL: <https://linkinghub.elsevier.com/retrieve/pii/S0098300417301346>.
- Xie, S., Tu, Z., 2015. Holistically-nested edge detection. [arXiv:1504.06375](https://arxiv.org/abs/1504.06375).
- Yan, Z., Liu, S., Gu, H., 2019. Fault image enhancement using a forward and backward diffusion method. *Comput. Geosci.* 131 (June), 1–14. <http://dx.doi.org/10.1016/j.cageo.2019.06.004>, URL: <https://doi.org/10.1016/j.cageo.2019.06.004><https://linkinghub.elsevier.com/retrieve/pii/S0098300418308513>.
- Yuan, Z., Huang, H., Jiang, Y., Tang, J., Li, J., 2019. An enhanced fault-detection method based on adaptive spectral decomposition and super-resolution deep learning. *Interpretation* 7 (3), T713–T725. <http://dx.doi.org/10.1190/INT-2018-0180.1>, URL: <https://library.seg.org/doi/10.1190/INT-2018-0180.1>.
- Zhang, Q., Yusifov, A., Joy, C., Shi, Y., Wu, X., 2019. FaultNet: A deep CNN model for 3D automated fault picking. In: SEG Technical Program Expanded Abstracts 2019, No. August. Society of Exploration Geophysicists, pp. 2413–2417. <http://dx.doi.org/10.1190/segam2019-3215930.1>, URL: <https://library.seg.org/doi/10.1190/segam2019-3215930.1>.