Illinois Institute of Technology

CS-584 Machine Learning

Identification of Salt Regions in Seismic Images using Deep Learning

Naveen Raju Sreerama Raju Govinda Raju(A20516868)
nsreeramarajugovinda@hawk.iit.edu

Sai Manohar Vemuri (A20514848)
svemuri6@hawk.iit.edu

Sesha Shai Datta Kolli(A20516330)
skolli2@hawk.iit.edu

Dr. Yan Yan

TABLE OF CONTENTS

LIST OF FIGURES

1. Introduction

Many companies Extracting oil and gas reserves, several places of the Earth that are rich in oil and natural gas Finding subsurface salt deposits using Seismic waves Knowing precise locations of large salt deposits is extremely important to companies involved in oil and gas exploration. Large amounts of salt are present beneath the surface of many oceans and seas around the world. Seismic images have historically been used to locate salt deposits underground. Misinterpretation of the salt boundaries typically results in higher expenses. Seismic image interpretation by humans can be difficult at times and is not always accurate. Salt density is usually about 2.14 g/cm3, which is less than most nearby rocks. Salt seismic velocity of around 4.5 km/s is usually bigger than the surrounding rocks

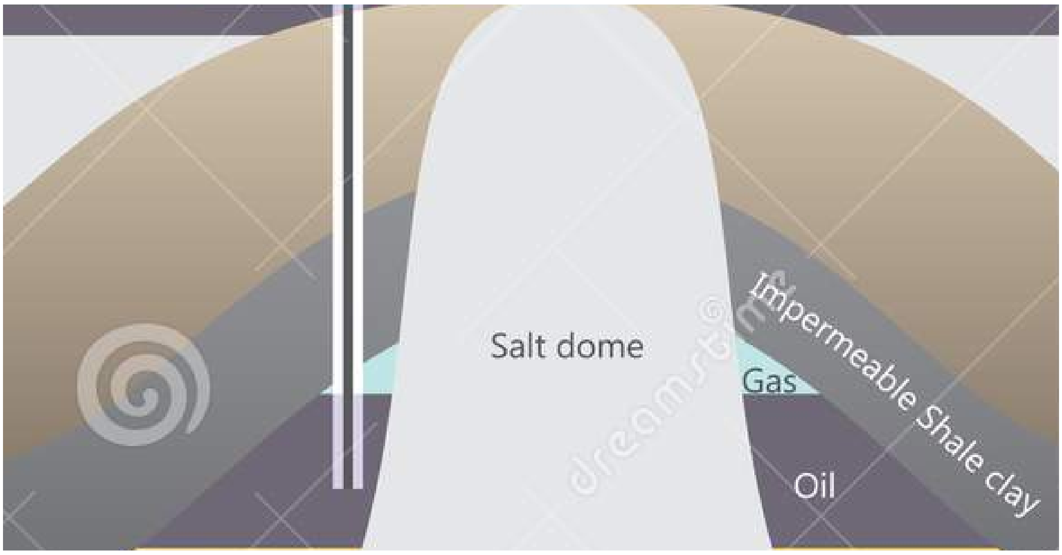


Fig.1.1. Gas and oil regions in earths subsurface

Technology called seismic imagery is used to describe the subsurface geology. It is based on the emission of sound waves into the ocean, where they are reflected off of underwater object structures and detected on the surface by hydrophone receivers.Seismic imaging enables the 3D visualization of underground structures

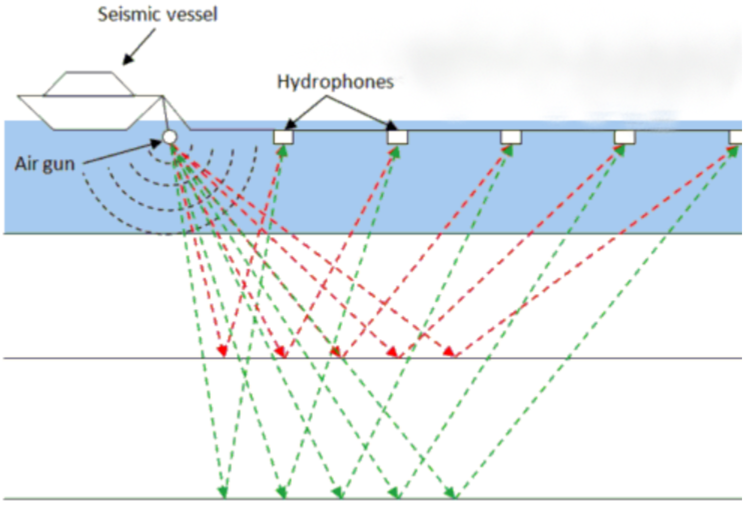


Fig.1.2. seismic image creation process

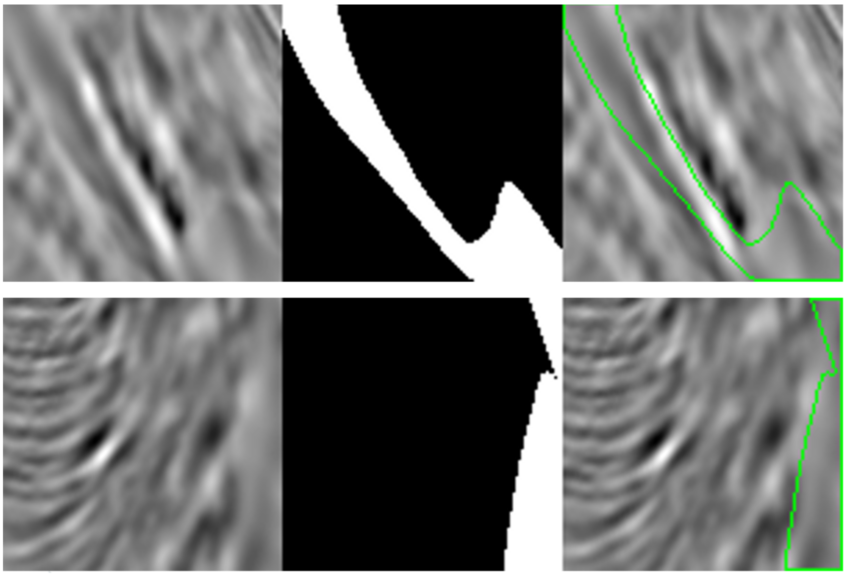


Fig.1.3. sample training data

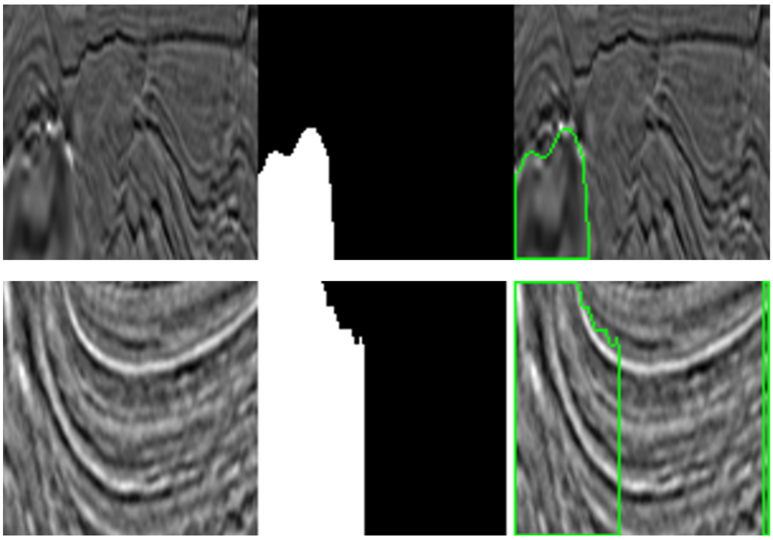


Fig.1.4. sample training data

In the two images above, the seismic image is on the left of the image stack, the salt region's binary mask is in the middle, and the right most image has binary mask's contours are drawn on it.

2. Problem description

Large amounts of salt are found in the subsurface of oceans and seas in several parts of the world. Seismic images have traditionally been used to identify the subsurface with salt, which is technically difficult because the seismic velocity of salt is very fast compared to other substances in the ocean. This is because salt is an amorphous rock with little structure.

internal organization Whereas regular rocks have some internal structure, seismic imaging technology can distinguish between different rocks but not salt. As a result, we employ Deep Learning algorithms to segment the salt regions in seismic images.

2.1. U-Net

The UNet architecture, which consists of a contraction path and an extraction path made up of CNN blocks, is typically used for semantic segmentation. Skip connections between the corresponding CNN blocks between the contraction and extraction path go along with this. As it descends into the lower layers of the encoding section during the encoding process in the contraction path, spatial information is reduced in a specific pattern while also picking up various significant features at various stages. Features from the contraction path are expanded at various stages during the expansive path by adding features from the corresponding paths of the contraction path

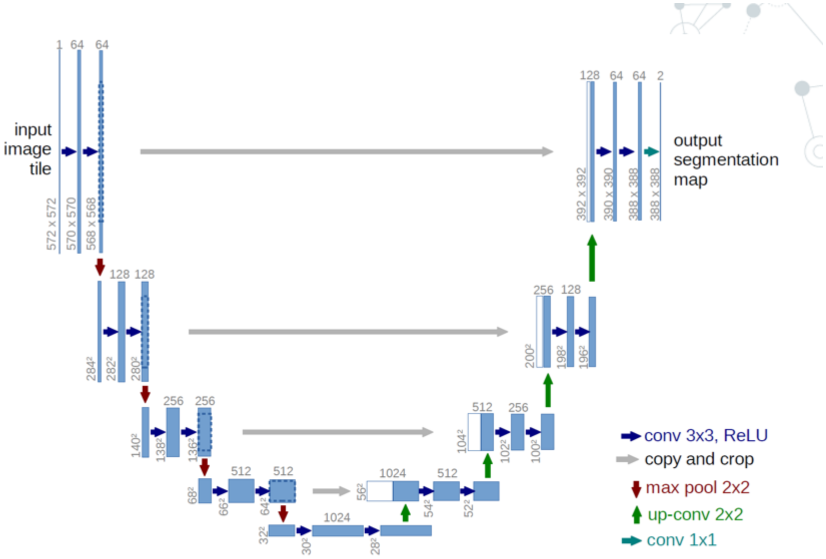


Fig.2.1. Unet

2.2. VGG 16

VGG, a team of Oxford researchers who created this architecture, goes by the name of Visual Geometry Group. We used VGG 16 is used as a backbone for UNet in encoder as well as decoder. The network receives a dimensioned image as input (224, 224, 3). The same padding and 64 channels with a 3*3 filter size are present in the first two layers. Then, two layers have convolution layers of 128 filter size and filter size after a max pool layer of stride (2, 2) (3, 3). The next layer is a max-pooling stride (2, 2) layer that is identical to the layer before it. There are then 256 filters spread across 2 convolution layers with filter sizes of 3 and 3. There are then two sets of three convolution layers, followed by a max pool layer. Each filter has the same padding and has 512 filters of size (3, 3) size with the same padding. The stack of two convolution layers then receives this image. The stack of two convolution layers then receives this image. The filters we use in these convolution and max-pooling layers are 3*3 in size. It also uses 1*1 pixels in some of the layers to adjust the number of input channels. To prevent the spatial feature of the image, 1-pixel padding is applied after each convolution layer. We obtained a (7, 7, 512) feature map after adding a convolution and max-pooling layer to the stack.

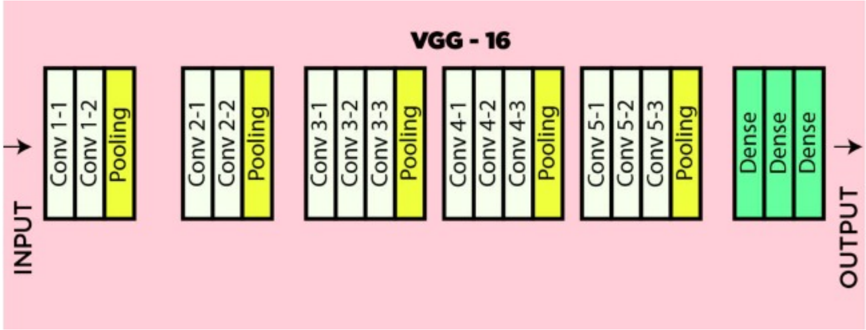


Fig.2.2. Vgg

2.3. ResNeT

In general, CNN architectures with numerous layers are required to capture fine-grained details. The Vanishing/Exploding gradient is a well-known deep learning issue that arises when the number of layers is increased. This results in the gradient becoming zero or being too large. Therefore, the training and test error rate also increases as the number of layers is increased. A new architecture called Residual Network was introduced by Microsoft Research researchers in 2015 with the proposal of ResNet. The Residual Blocks concept was introduced by this architecture to address the issue of the vanishing/exploding gradient. We apply a technique known as skip connections in this network. The skip connection bypasses some layers in between to link layer activations to subsequent layers. This creates a leftover block. These residual blocks are stacked to create resnets. The strategy behind this network is to let the network fit the residual mapping rather than have layers learn the underlying mapping. The benefit of including this kind of skip connection is that regularization will skip any layer that degrades architecture performance. As a result, training a very deep neural network is possible without encountering issues with vanishing or exploding gradients.

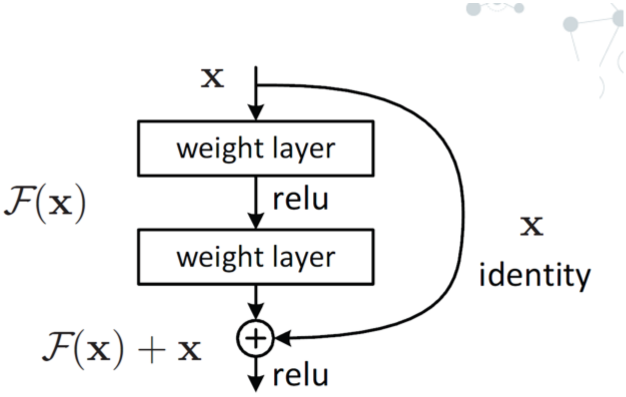
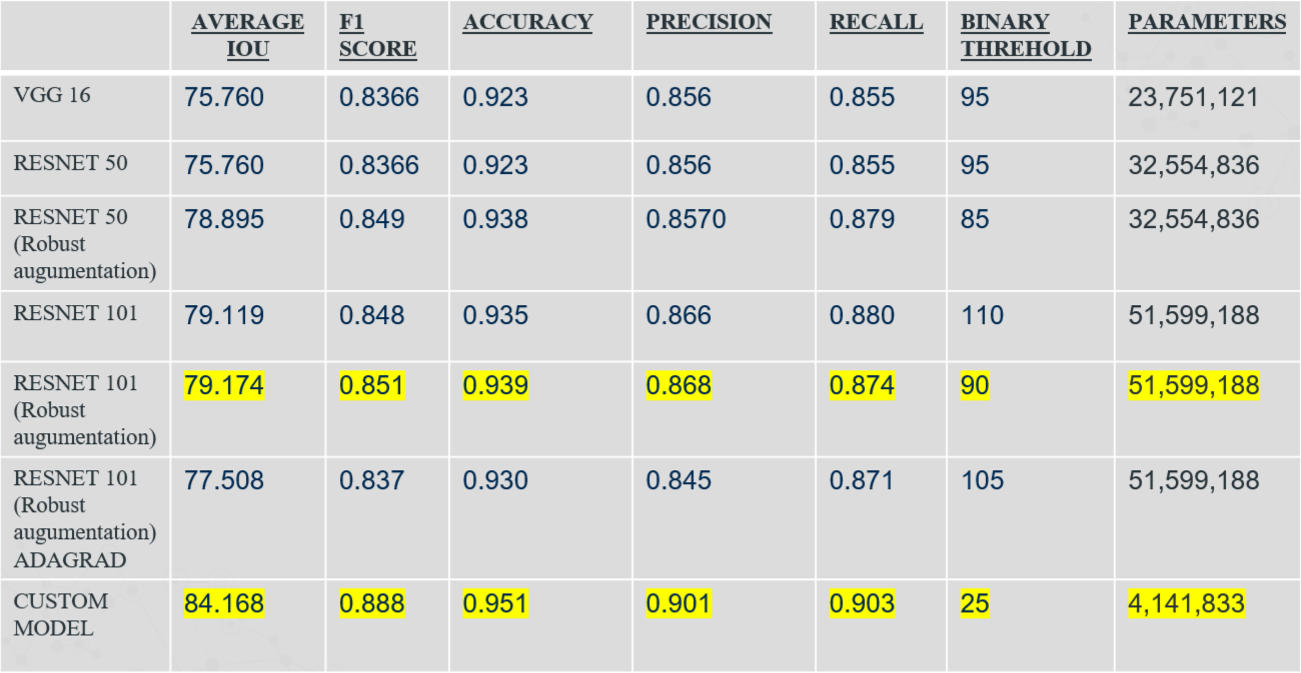

x
weight layer
$\mathcal{F}(\mathbf{x})$
relu
weight layer
x
identity
$\mathcal{F}(\mathbf{x}) + \mathbf{x}$
relu

Fig.2.3. ResNet

2.4. Training procedure

The training dataset comprises of 1151 seismic images and its co responding ground truth binary masks. The testing dataset comprises of 302 seismic images and its co responding ground truth binary masks. We used an initial learning rate of 0.01 with a reducing learning rate on a plateau with a patience rate of 4 and a factor of 0.1, based on validation loss. We experimented with Adam and Ada Grad optimizers. Various augmentation techniques used are: vertical flip, horizontal flip, width-shift, height-shift, shear, zoom, rotation, contrast-limited adaptive histogram equalization, image sharpening, and gamma correction (Power Law Transform). We rained for 100–200 epochs on average with a batch size of 6. Metrics used were the IOU score to evaluate how well the model is trained, and loss functions used were binary cross entropy and a combination of focal loss and dice loss.

2.5. Evaluation metrics



	AVERAGE IOU	F1 SCORE	ACCURACY	PRECISION	RECALL	BINARY THREHOLD	PARAMETERS
VGG 16	75.760	0.8366	0.923	0.856	0.855	95	23,751,121
RESNET 50	75.760	0.8366	0.923	0.856	0.855	95	32,554,836
RESNET 50 (Robust augumentation)	78.895	0.849	0.938	0.8570	0.879	85	32,554,836
RESNET 101	79.119	0.848	0.935	0.866	0.880	110	51,599,188
RESNET 101 (Robust augumentation)	79.174	0.851	0.939	0.868	0.874	90	51,599,188
RESNET 101 (Robust augumentation) ADAGRAD	77.508	0.837	0.930	0.845	0.871	105	51,599,188
CUSTOM MODEL	84.168	0.888	0.951	0.901	0.903	25	4,141,833

Fig.2.5. Evalution Metrics

For the following architectures as backbone to Unet that is VGG16, RESNET 50, RESNET101. Training parameters used were:
a)Augmentation - vertical flip, horizontal flip, width-shift, height-shift, shear, zoom, rotation
b)Metrics - IOU score
c)loss function - Binary cross entropy
d)Epochs - 200
e)Batch size - 6
f)Input dimension - 224*224*1
g)Optimiser - Adam
h)initial learning rate of 0.01 with a reducing learning rate on plateau with a patience rate of 4 and a factor of 0.1

For the following architectures as backbone to Unet that is RESNET 50(Robust augmentation), RESNET101(Robust augmentation) and Custom architecture.
For RESNET variants, pre-trained weights from the above mentioned approach were used. Training parameters used were:
a)Augmentation - vertical flip, horizontal flip, width-shift, height-shift, shear, zoom, rotation, contrast-limited adaptive histogram equalization, image sharpening, and gamma correction (Power Law Transform)
b)Metrics - IOU score
c)loss function - Focal loss+Dice loss
d)Epochs - 200
e)Batch size - 6
f)Input dimension - 224*224*1 , for custom model input dimension was 128*128*1
g)Optimiser - Adam
h)initial learning rate of 0.01 with a reducing learning rate on plateau with a patience rate of 4 and a factor of 0.1
Only for RESNET 101 Ada grad optimiser was also used.
As we can see from the table, the best performing model was Custom model and second best performing model was RESNET 101.

2.6. Graphs

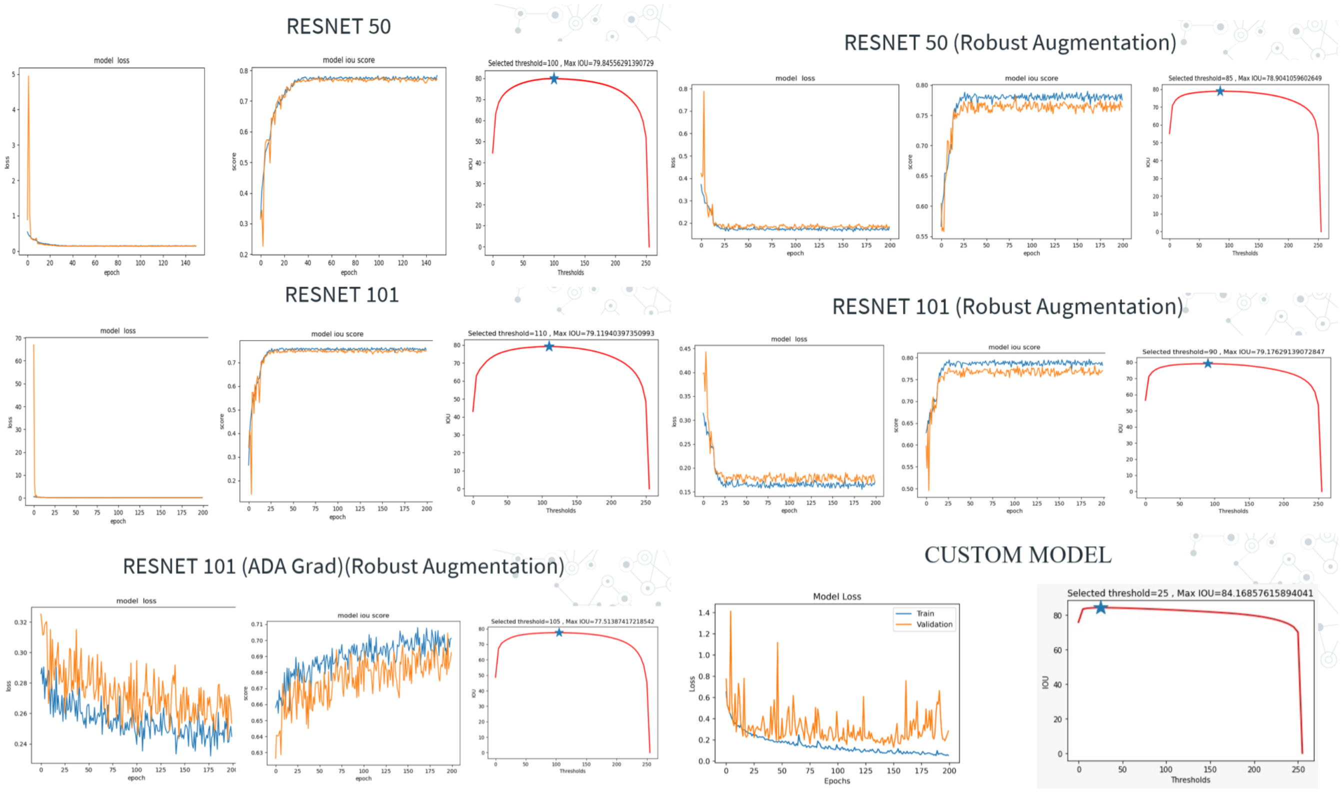


Fig.2.6. Training Graphs

Blue lines in the aforementioned graphs represent training metrics, while orange lines represent validation metrics.

2.7. Loss Function

The justification for why Balance cross entropy manages class imbalance better than Binary cross entropy is given mathematically below. Then it shows how focal loss overcomes the drawbacks of balanced cross entropy by giving more weight to classes that are difficult to classify and less weight to classes that are easy to classify. Furthermore, how focal loss deals with the issue of class inequality .
1. Binary Cross Entropy:
CE = -log(p)*y - log(1-p)(1-y)
where,p is the predicted probability, y is the ground truth
When y=1,
p=0.9 CE= 0.045
P=0.1 CE=1

2. Balanced Binary Cross Entropy:
Positive samples - 100
Negative samples - 80
Class weight for positive sample = 0.8
Class weight for negative sample = 1
CE = -0.8*log(p)*y - 1*log(1-p)(1-y)
When y=1, p = 0.9 CE = 0.0366 p = 0.1 CE = 0.8

3. Focal loss

Focal loss(p) when y=1 is $-\propto * \, (1-p)^{\gamma}$ * log(p)

Focal loss(p) when y=0 is -(1-∝) $* \, (p)^{\gamma}$ * log(1-p)

∝ - Balances importance between poistive and negative examples, but it does not differentiate between easy and hard examples.

$(1-p)^{\gamma}$ is a modulating factor and γ is tunable parameter which smoothly adjusts the rate at which easy examples is down-weighted.

Case 1:

When, y=1, p=0.9, Focal loss(p) = 0.000125

$$\frac{CE}{FL} = \frac{0.045}{0.000125} = 360$$

$$\frac{BCE}{FL} = \frac{0.0366}{0.000125} = 292.8$$

Case 2:

When, y=1, p=0.1, Focal loss(p) = 0.2025

$$\frac{CE}{FL} = \frac{1}{0.2025} = 4.938$$

$$\frac{BCE}{FL} = \frac{0.8}{0.2025} = 3.9506$$

.

Case 1:

For correctly classifying samples we can see that focal loss has 360 times less loss than Binary cross entropy and 292.8 times less loss than Balanced binary cross entropy.

Case2:

For wrongly classified samples we can see that focal loss is just 4.938 times less loss than Binary cross entropy and 3.9506 times less loss than Balanced binary cross entropy.

Thus we can see how strongly focal loss places emphasis on wrongly classified samples.

4) Dice loss

true positive(tp) = ground truth * prediction
false positive(fp) = prediction - true positive
false negative(fn) = ground truth - ground truth

IOU score $= \frac{(1+\beta^2)*tp+smooth}{(1+\beta^2)*tp+\beta^2*fn+fp+smooth}$

Loss = 1 - IOU score

The Dice loss is a popular loss in the computer vision community for determining how similar the ground truth and a prediction made using intersection over union logic.

3. Results

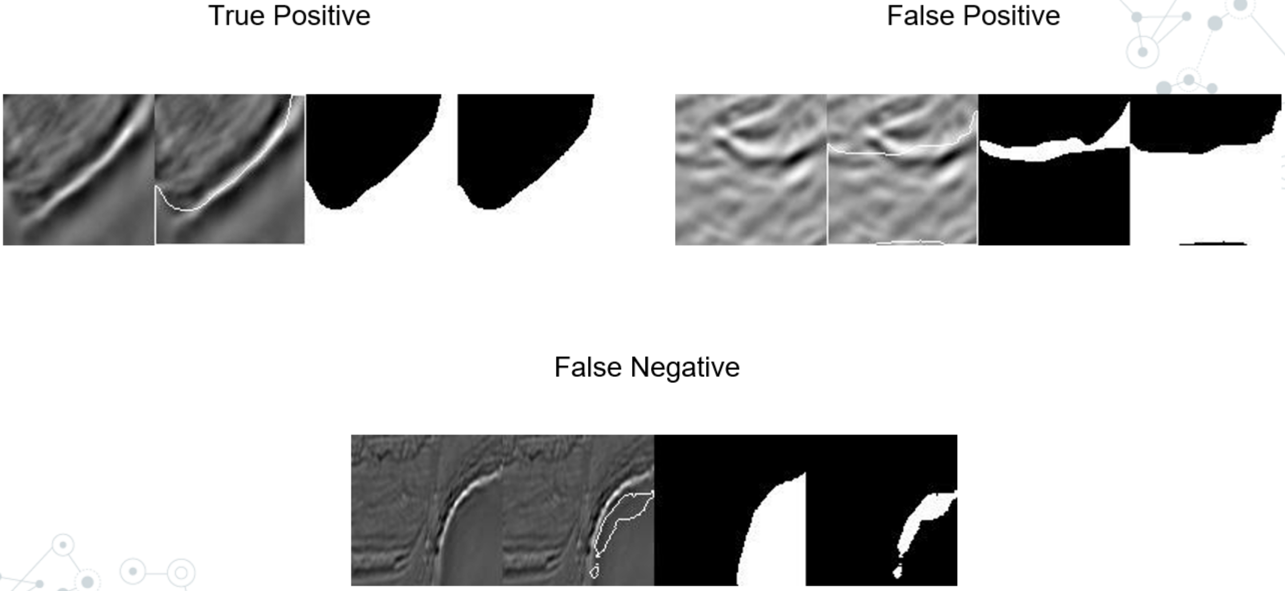


Fig.3.1. Predictions

The first image in each of the image stacks above is the input image, the second image is the binary mask prediction contours drawn on the input image, the third image is the ground truth binary mask, and the final image is the model prediction.

4. Conclusion and future scope

Conclusion:
After several experiments of using different backbone architecture for UNet we were able to get best results on our custom built backbone architecture for UNet that surpassed the evaluation metrics of big architectures like ResNet 50, ResNet 101 and VGG16. Compared to ResNet 50 our architecture had 7.86 times less total parameters,compared to ResNet 101 our architecture had 12.45 times less total parameters and compared to VGG 16 our architecture had 5.73 times less total parameters. The results were as follows: IOU: 84.168

Future Scope:
1. More research can be done in the future to improve the performance of the proposed hybrid design and overcome its current drawbacks. The following are the primary research directions that can be tested. Despite
being practical in real-time applications due to today's computing capabilities, the number of parameters of our model can be reduced by reducing model complexity by taking into account the design principles of some other state-of-the-art models with a lower number of parameters.
2. Better performance on the IoU metric can be a topic of future research by taking into account some modern design approaches of CNNs as well as model design choices that give a high IoU.
3. The data set can be sampled into various groups based on the visibility of the foreground features, and several custom efficient models can be trained so we can ensemble those models to produce better results.

.

5. Contributions

Naveen Raju Sreerama Raju Govinda Raju
- Image processing augmentation exploration
- UNet architecture with different backbones coding
- Training and testing

Sai Manohar Vemuri
- Unet Architecture research
- Custom architecture design and exploration
- Training and testing

Sesha Shai Datta Kolli
- Data set exploration and domain knowledge research
- Data set cleaning
- Analysis of predictions, U Net architecture exploration

6. Project Repository

https://github.com/mvemuri6642/Identification_of_Salt_Regions_in_Seismic_Images_using_DeepLearning

7. Reference

[1] Daniel Civitarese1, Daniela Szwarcman1, Emilio Vital Brazil1, Bianca Zadrozny,"Semantic Segmentation of Seismic Images"

[2] Olaf Ronneberger, Philipp Fischer, Thomas Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation"

[3] Karen Simonyan, Andrew Zisserman, "VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION"

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, "Deep Residual Learning for Image Recognition"

[5] Christopher M. Bishop,"Pattern Recognition and Machine Learning"

[6] Quentin Jodeleta,b, Xin Liub, Tsuyoshi Murataa, "Balanced softmax cross-entropy for incremental learning with and without memory"

[7] Tsung-Yi Lin, Priya Goyal, Ross Girshic,k Kaiming He, Piotr Dollar,"Focal Loss for Dense Object Detection"

[8] Carole H. Sudre, Wenqi Li, Tom Vercauteren, Sebastien Ourselin,M. Jorge Cardoso,"Generalised Dice overlap as a deep learning lossfunction for highly unbalanced segmentations"

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

[10] Reference Repository: https://github.com/zhixuhao/unet

[11] Reference Repository: https://github.com/qubvel/segmentation_models

[12] https://www.geeksforgeeks.org/vgg-16-cnn-model/

[13] https://www.geeksforgeeks.org/residual-networks-resnet-deep-learning/