Imperial College London

PERENCO

Supervised Seismic Section Classification with Convolutional Neural Networks (CNNs)

Zongpeng Chen

Supervisors: Olivier Dubrule, Lukas Mosser
Jeremy Fortun, Kurt Rattansingh

Introduction

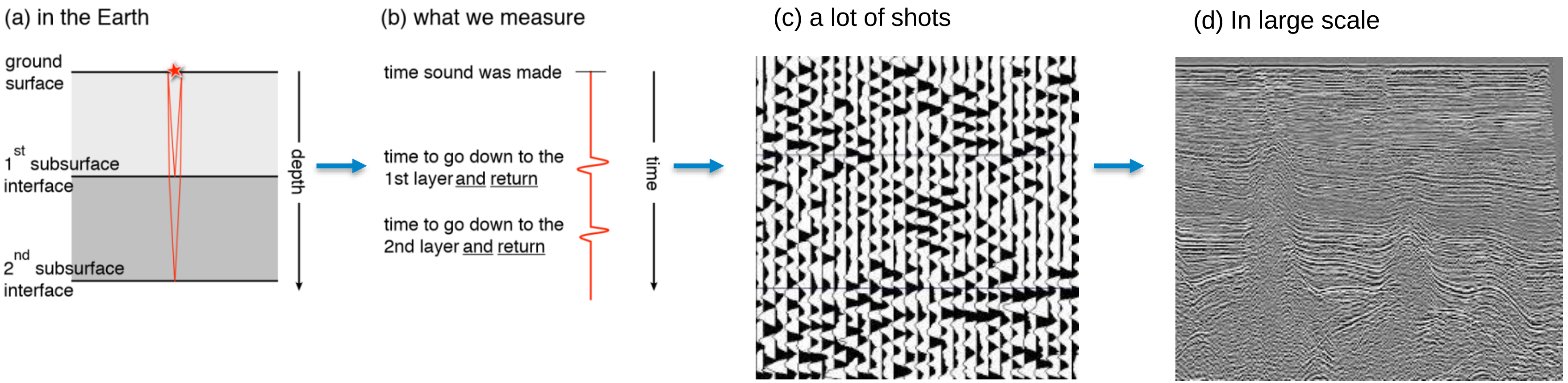
1. General Introduction
   - Seismic Section
   - Objective
   - Workflow

2. Training and Results
   - Dataset
   - Test Section
   - Feed-Forward DNN Model and Result
   - Convolutional Neural Network (CNN) Models and Results
   - Final Ensemble Model and Result
   - Summary

3. After Training
   - Discussion
   - Conclusion

Q&A

Seismic Section



(a) in the Earth

(b) what we measure

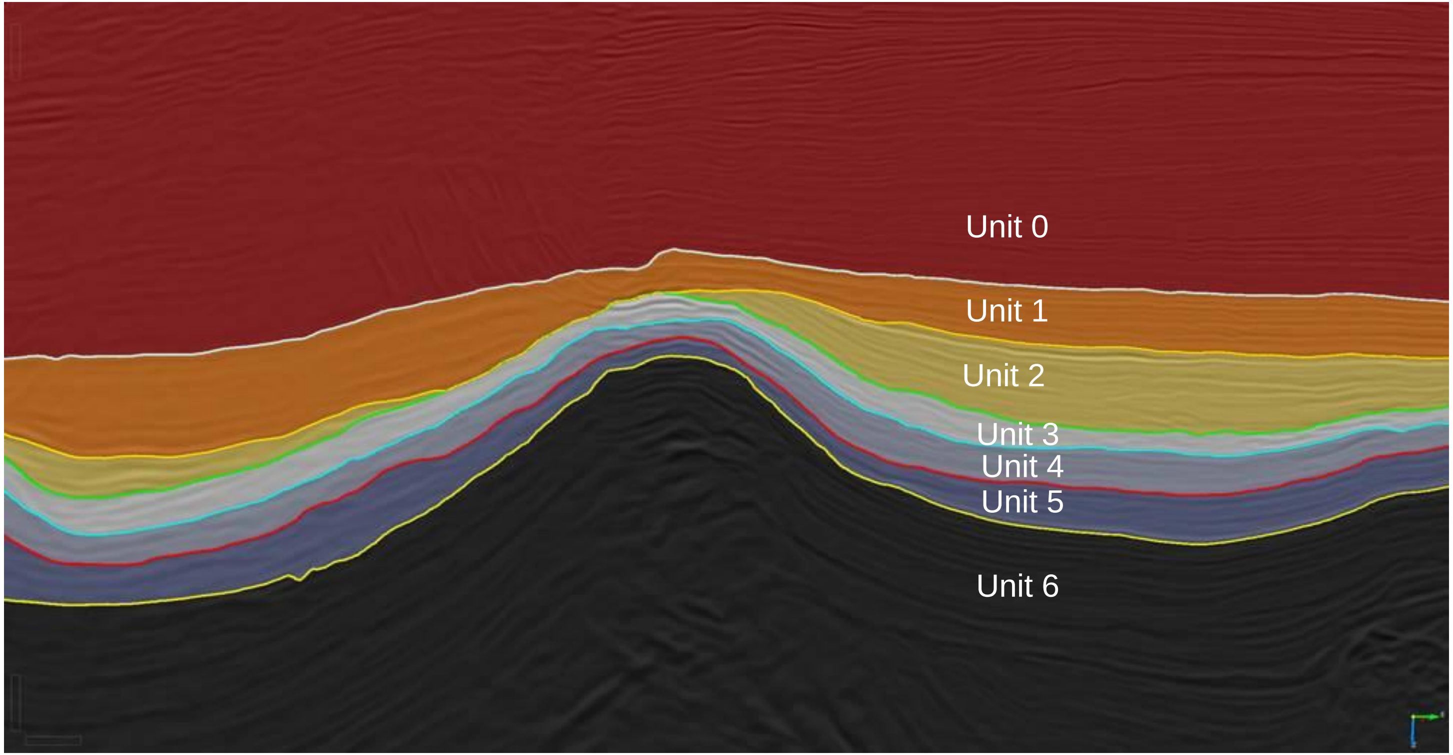
(c) a lot of shots

(d) In large scale

Imperial College London

Supervised Seismic Section Classification with Convolutional Neural Networks

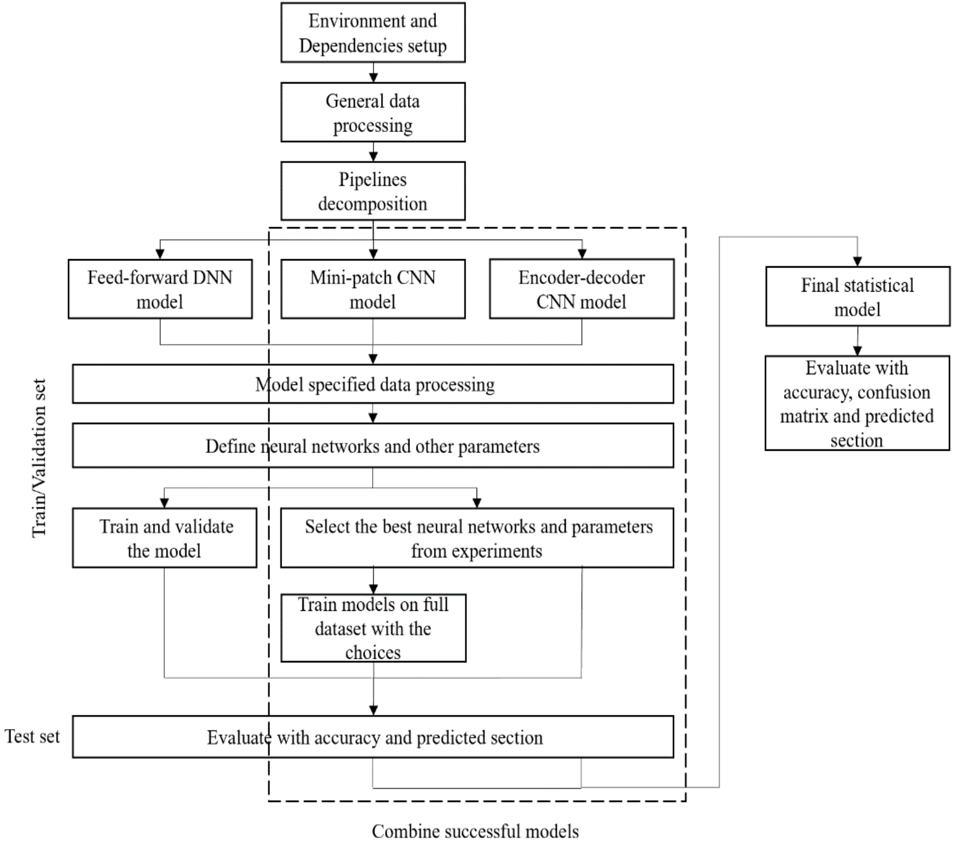
Objective



Test field in Gabon.

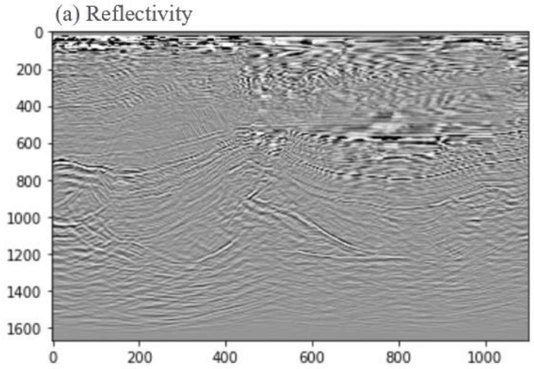
7 different units.

Workflow

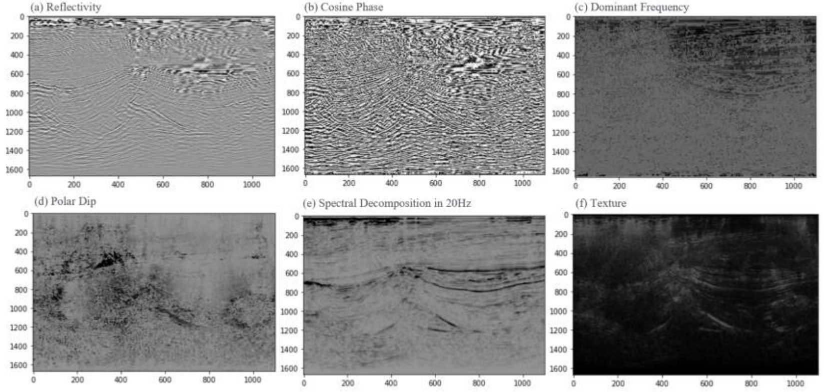

Environment and Dependencies setup
General data processing
Pipelines decomposition
Feed-forward DNN model
Mini-patch CNN model
Encoder-decoder CNN model
Model specified data processing
Define neural networks and other parameters
Train/Validation set
Train and validate the model
Select the best neural networks and parameters from experiments
Train models on full dataset with the choices
Test set
Evaluate with accuracy and predicted section
Combine successful models
Final statistical model
Evaluate with accuracy, confusion matrix and predicted section

Dataset

5 interpreted Seismic sections are manually chosen to train the Machine (**0.1% of total seismic dataset**).
Split in 3: 1: 1 (**3 Training sections, 1 Validation section and 1 Test section**).



VS

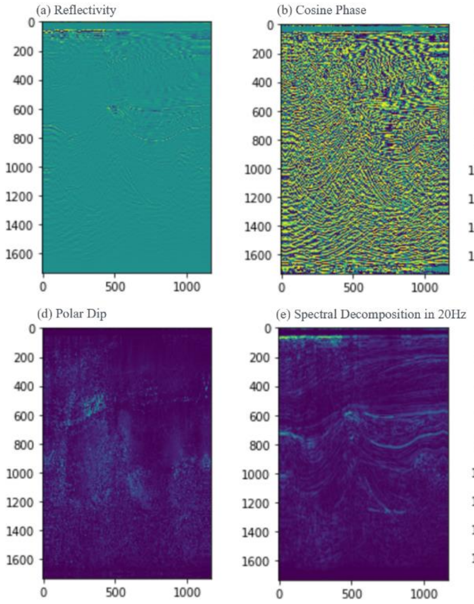


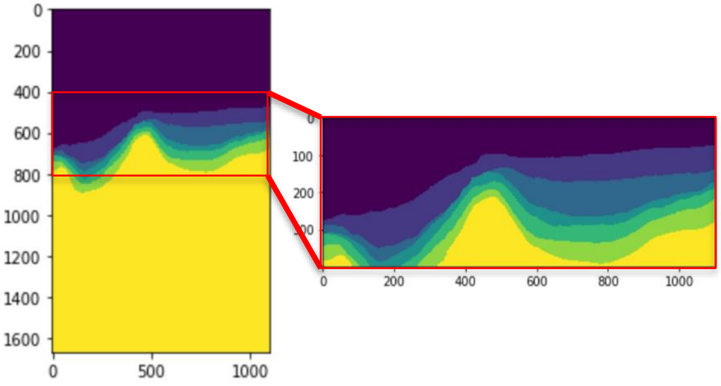
‘1-feature’

‘6-feature’

Imperial College London

Test Section



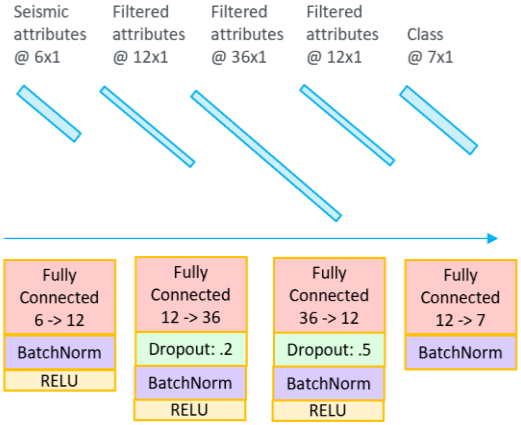


Class label map

Feed-Forward DNN Model

Feed-Forward DNN Model and Result

Network Architecture



Seismic attributes @ 6x1

Filtered attributes @ 12x1

Filtered attributes @ 36x1

Filtered attributes @ 12x1

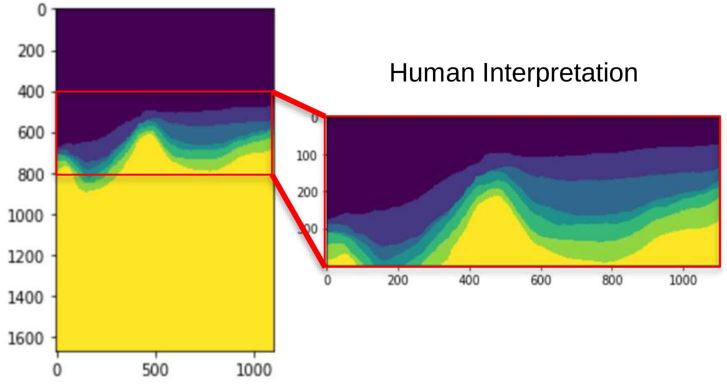
Class @ 7x1

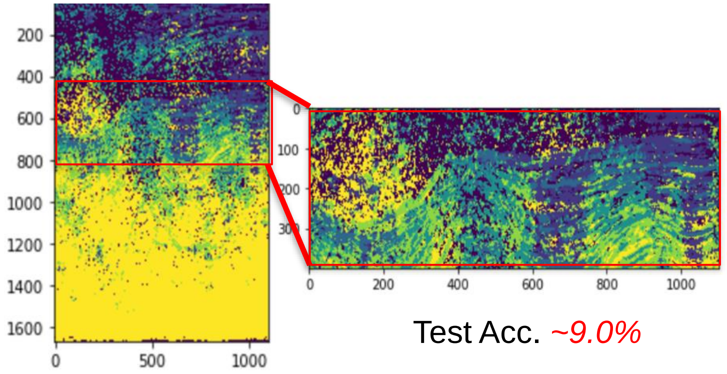
Fully Connected 6 -> 12	Fully Connected 12 -> 36	Fully Connected 36 -> 12	Fully Connected 12 -> 7
BatchNorm	Dropout: .2	Dropout: .5	BatchNorm
RELU	BatchNorm	BatchNorm	
	RELU	RELU	

Only on ‘6-feature’ dataset.

Bad accuracy and geological continuity.

Reference only.

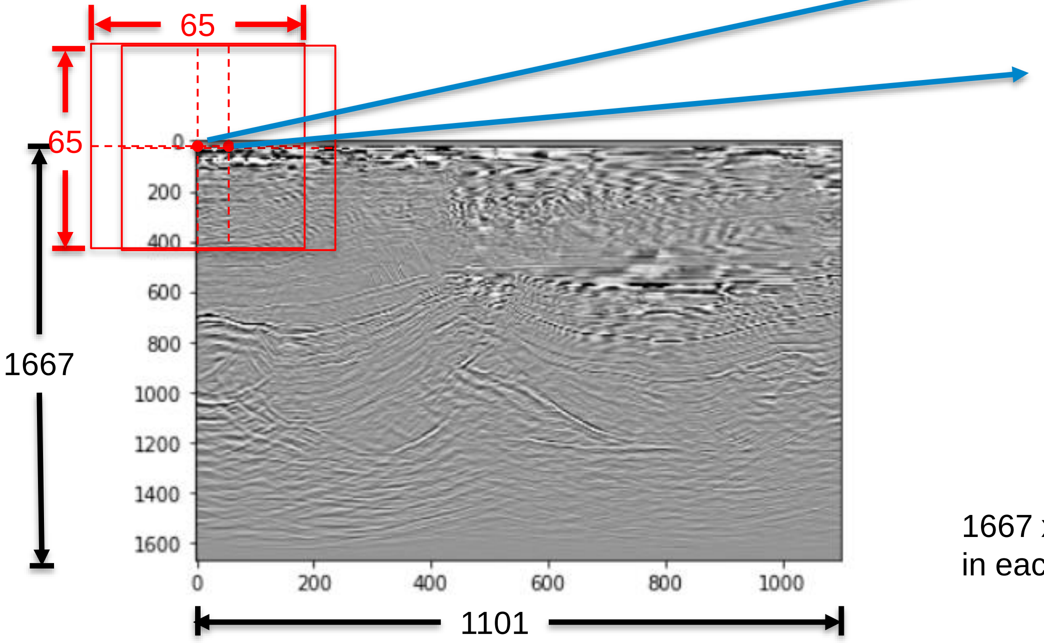




Test Acc. ~9.0%

Mini-patch CNN Model

Mini-patch CNN Model



1st pixel

2nd pixel

…

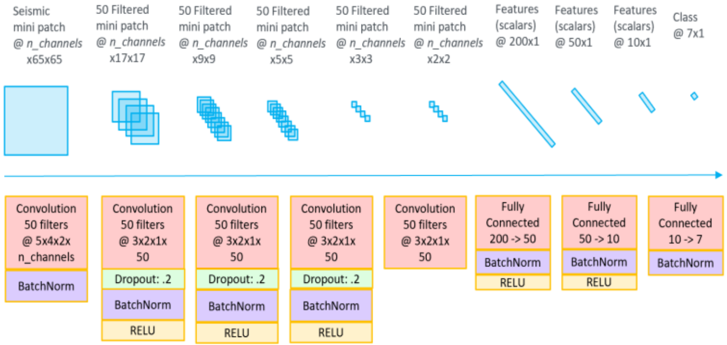
1667 x 1101 training squares in each seismic section

Imperial College London

Convolutional Neural Network (CNN) Models and Results

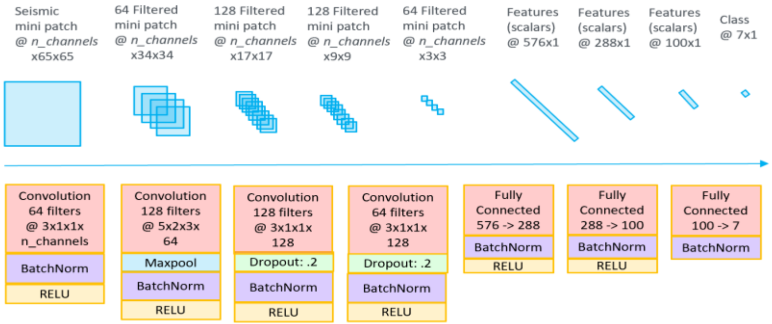
Mini-patch CNN Model

Network Architectures



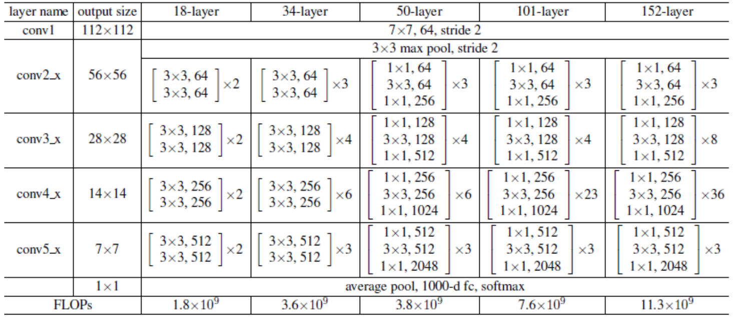
Seismic mini patch @ *n_channels* x65x65
50 Filtered mini patch @ *n_channels* x17x17
50 Filtered mini patch @ *n_channels* x9x9
50 Filtered mini patch @ *n_channels* x5x5
50 Filtered mini patch @ *n_channels* x3x3
50 Filtered mini patch @ *n_channels* x2x2
Features (scalars) @ 200x1
Features (scalars) @ 50x1
Features (scalars) @ 10x1
Class @ 7x1
Convolution 50 filters @ 5x4x2x n_channels
BatchNorm
Convolution 50 filters @ 3x2x1x 50
Dropout: .2
BatchNorm
RELU
Convolution 50 filters @ 3x2x1x 50
Dropout: .2
BatchNorm
RELU
Convolution 50 filters @ 3x2x1x 50
Dropout: .2
BatchNorm
RELU
Convolution 50 filters @ 3x2x1x 50
Fully Connected 200 -> 50
BatchNorm
RELU
Fully Connected 50 -> 10
BatchNorm
RELU
Fully Connected 10 -> 7
BatchNorm

BasicNet



Seismic mini patch @ *n_channels* x65x65
64 Filtered mini patch @ *n_channels* x34x34
128 Filtered mini patch @ *n_channels* x17x17
128 Filtered mini patch @ *n_channels* x9x9
64 Filtered mini patch @ *n_channels* x3x3
Features (scalars) @ 576x1
Features (scalars) @ 288x1
Features (scalars) @ 100x1
Class @ 7x1
Convolution 64 filters @ 3x1x1x n_channels
BatchNorm
RELU
Convolution 128 filters @ 5x2x3x 64
Maxpool
BatchNorm
RELU
Convolution 128 filters @ 3x1x1x 128
Dropout: .2
BatchNorm
RELU
Convolution 64 filters @ 3x1x1x 128
Dropout: .2
BatchNorm
RELU
Fully Connected 576 -> 288
BatchNorm
RELU
Fully Connected 288 -> 100
BatchNorm
RELU
Fully Connected 100 -> 7
BatchNorm

OptimizedNet



layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\left[\begin{array}{c}3\times3, 64\\3\times3, 64\end{array}\right]\times 2$	$\left[\begin{array}{c}3\times3, 64\\3\times3, 64\end{array}\right]\times 3$	$\left[\begin{array}{c}1\times1, 64\\3\times3, 64\\1\times1, 256\end{array}\right]\times 3$	$\left[\begin{array}{c}1\times1, 64\\3\times3, 64\\1\times1, 256\end{array}\right]\times 3$	$\left[\begin{array}{c}1\times1, 64\\3\times3, 64\\1\times1, 256\end{array}\right]\times 3$
conv3_x	28×28	$\left[\begin{array}{c}3\times3, 128\\3\times3, 128\end{array}\right]\times 2$	$\left[\begin{array}{c}3\times3, 128\\3\times3, 128\end{array}\right]\times 4$	$\left[\begin{array}{c}1\times1, 128\\3\times3, 128\\1\times1, 512\end{array}\right]\times 4$	$\left[\begin{array}{c}1\times1, 128\\3\times3, 128\\1\times1, 512\end{array}\right]\times 4$	$\left[\begin{array}{c}1\times1, 128\\3\times3, 128\\1\times1, 512\end{array}\right]\times 8$
conv4_x	14×14	$\left[\begin{array}{c}3\times3, 256\\3\times3, 256\end{array}\right]\times 2$	$\left[\begin{array}{c}3\times3, 256\\3\times3, 256\end{array}\right]\times 6$	$\left[\begin{array}{c}1\times1, 256\\3\times3, 256\\1\times1, 1024\end{array}\right]\times 6$	$\left[\begin{array}{c}1\times1, 256\\3\times3, 256\\1\times1, 1024\end{array}\right]\times 23$	$\left[\begin{array}{c}1\times1, 256\\3\times3, 256\\1\times1, 1024\end{array}\right]\times 36$
conv5_x	7×7	$\left[\begin{array}{c}3\times3, 512\\3\times3, 512\end{array}\right]\times 2$	$\left[\begin{array}{c}3\times3, 512\\3\times3, 512\end{array}\right]\times 3$	$\left[\begin{array}{c}1\times1, 512\\3\times3, 512\\1\times1, 2048\end{array}\right]\times 3$	$\left[\begin{array}{c}1\times1, 512\\3\times3, 512\\1\times1, 2048\end{array}\right]\times 3$	$\left[\begin{array}{c}1\times1, 512\\3\times3, 512\\1\times1, 2048\end{array}\right]\times 3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

ResNet

Convolutional Neural Network (CNN) Models and Results

Mini-patch CNN Model

Training VS Validation

On ‘1-feature’ dataset:

Learning rate

		$2*10^{-2}$		$1*10^{-2}$		$5*10^{-3}$		$1*10^{-3}$	
		Training	Validation	Training	Validation	Training	Validation	Training	Validation
Neural Network	**BasicNet**	88.9%	88.6%	91.1%	88.3%	92.0%	89.8%	92.5%	89.9%
	OptimizedNet	97.6%	92.3%	98.4%	**92.4%**	98.8%	92.4%	99.0%	92.4%
	ResNet 18	98.7%	91.6%	99.3%	91.1%	99.5%	91.0%	99.7%	91.3%
	ResNet 34	98.6%	91.8%	99.3%	**91.9%**	99.5%	91.6%	99.7%	91.7%
	ResNet 50	98.3%	90.8%	99,1%	90.4%	99.4%	90.5%	99.7%	90.8%

On ‘6-feature’ dataset:

Learning rate

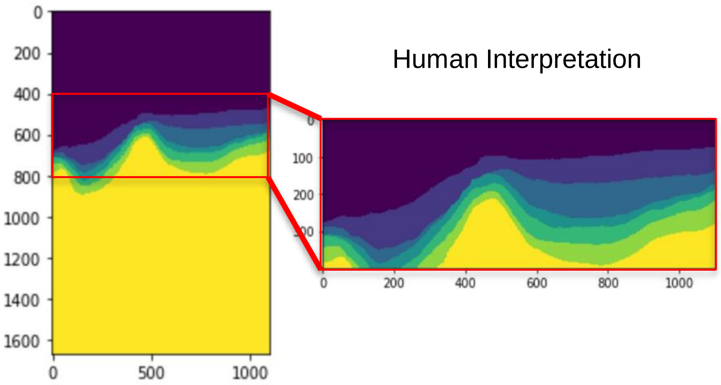
		$2*10^{-2}$		$1*10^{-2}$		$5*10^{-3}$		$1*10^{-3}$	
		Training	Validation	Training	Validation	Training	Validation	Training	Validation
Neural Network	**BasicNet**	88.5%	90.5%	90.9%	90.9%	92.0%	91.2%	92.5%	91.1%
	OptimizedNet	97.1%	92.4%	98.2%	**92.8%**	98.6%	93.1%	98.8%	93.0%
	ResNet 18	98.4%	92.2%	99.2%	92.0%	99.5%	92.2%	99.7%	92.5%
	ResNet 34	98.2%	92.2%	99.1%	**92.1%**	99.5%	92.1%	99.7%	92.3%
	ResNet 50	98.1%	91.8%	99.1%	91.8%	99.5%	92.0%	99.7%	92.0%

Imperial College London

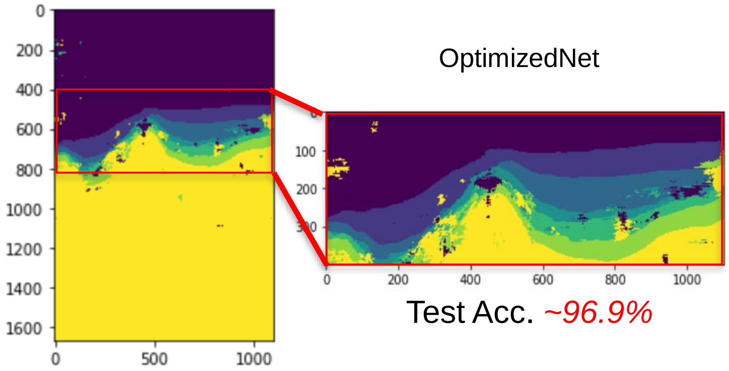
Convolutional Neural Network (CNN) Models and Results

Mini-patch CNN Model

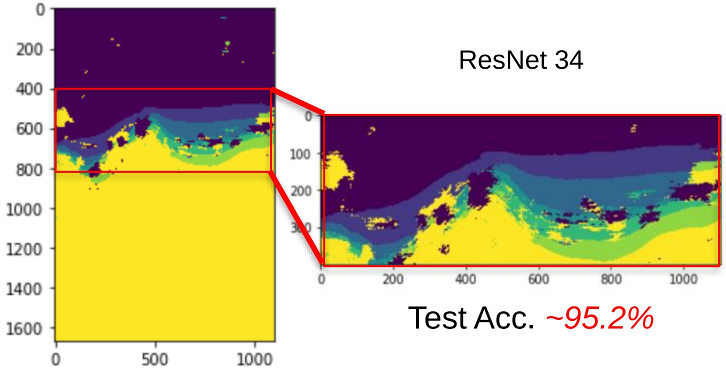
Full dataset training on ‘1-feature’ dataset


Human Interpretation

Good accuracy but low geological continuity.


OptimizedNet

Test Acc. *~96.9%*


ResNet 34

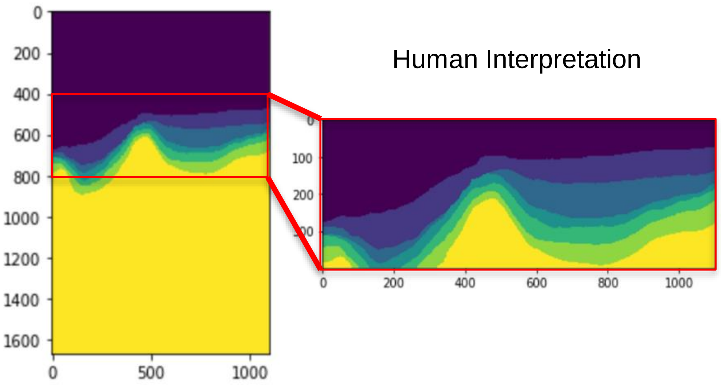
Test Acc. *~95.2%*

Imperial College London

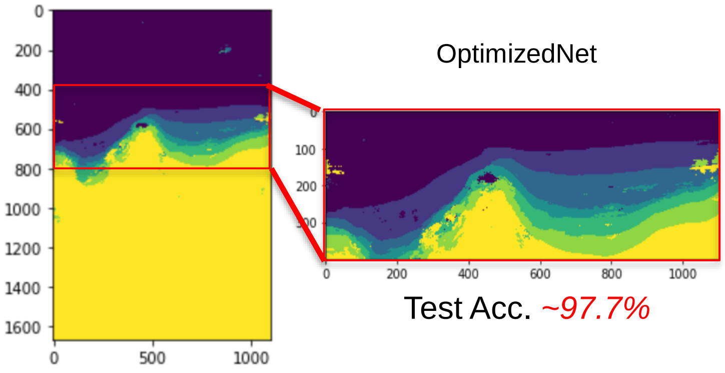
Convolutional Neural Network (CNN) Models and Results

Mini-patch CNN Model

Full dataset training on ‘6-feature’ dataset

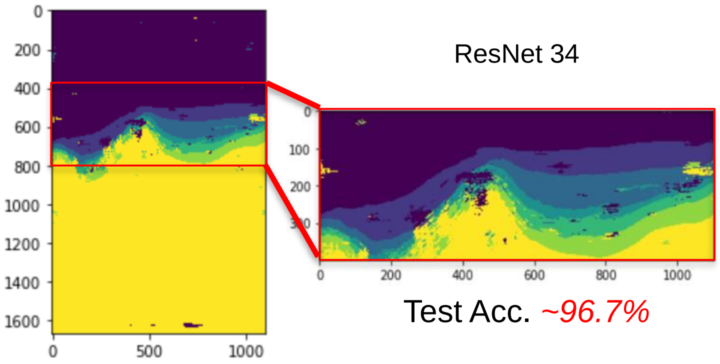


Human Interpretation



OptimizedNet

Test Acc. ~97.7%



ResNet 34

Test Acc. ~96.7%

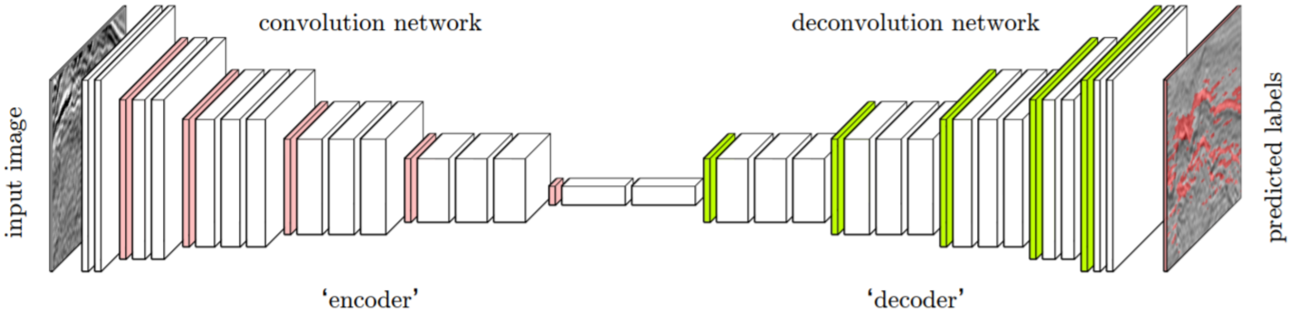
Better accuracy and better geological continuity.

Encoder-decoder CNN Model

Imperial College London

Convolutional Neural Network (CNN) Models and Results

Encoder-decoder CNN Model

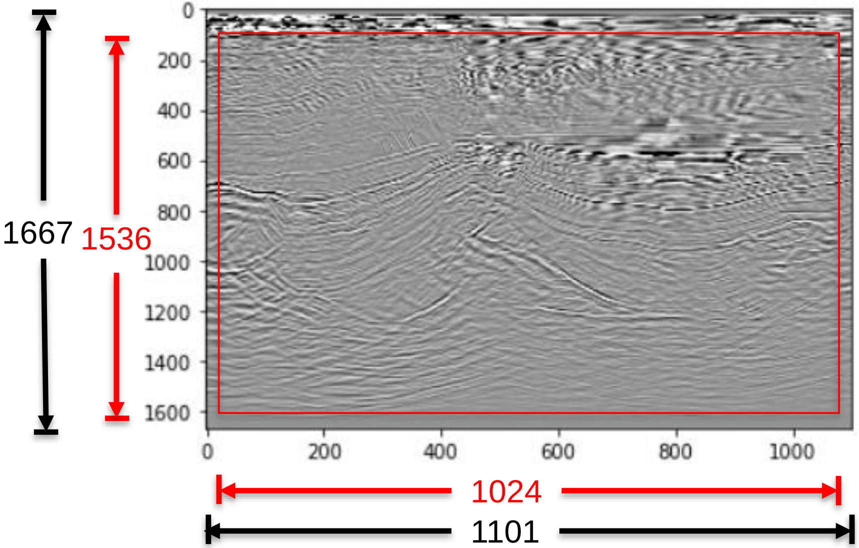


Whole image as a training object

Use 'encoder' to extract features and classify different facies

Use 'decoder' to accurately localize the different facies

Encoder-decoder CNN Model

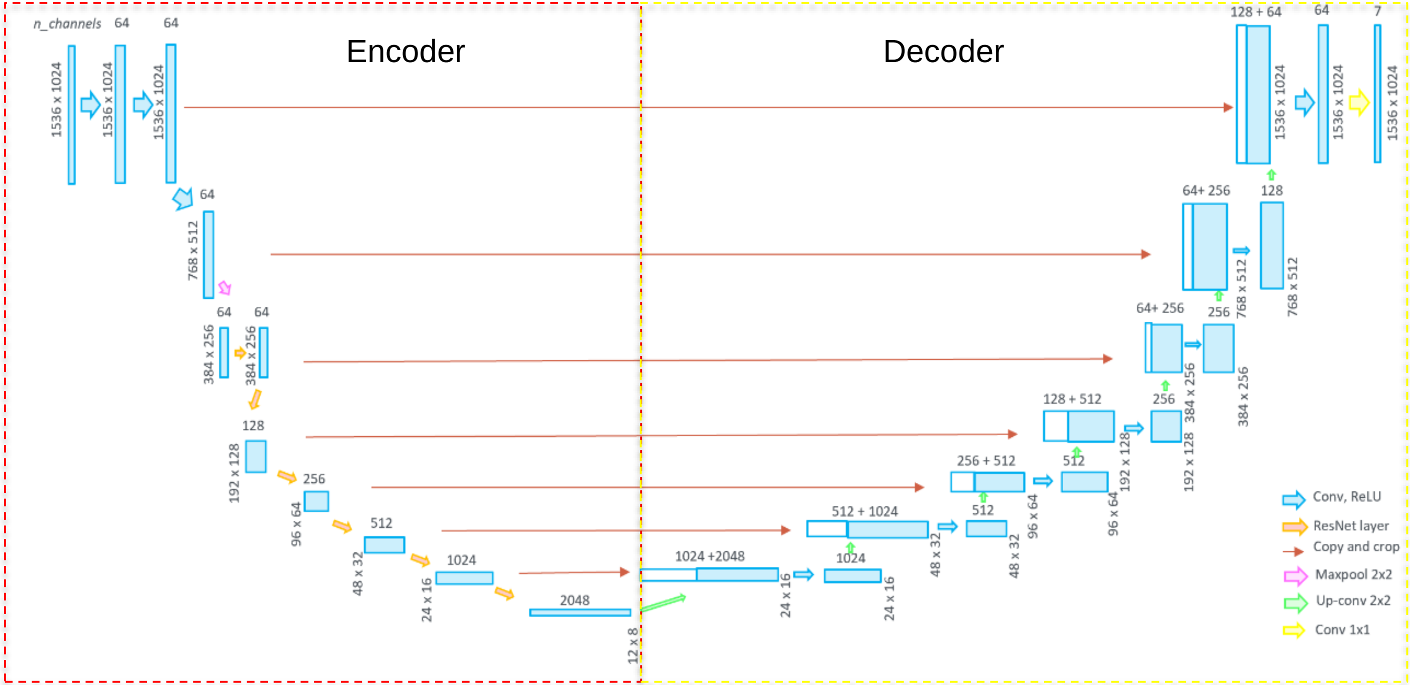


1 training squares in each seismic section

Convolutional Neural Network (CNN) Models and Results

Encoder-decoder CNN Model

Network Architecture

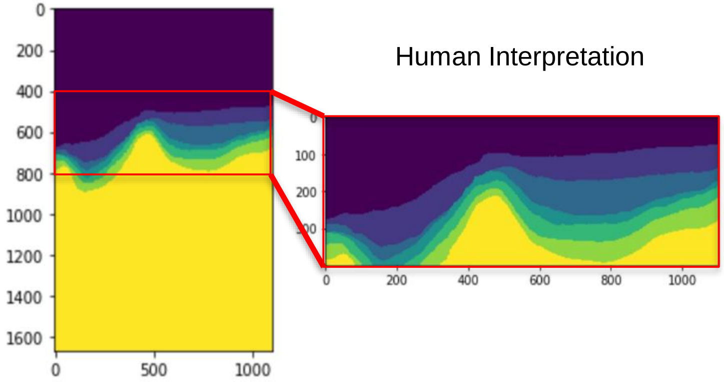


ResNetUNet

Imperial College London

Convolutional Neural Network (CNN) Models and Results

Encoder-decoder CNN Model

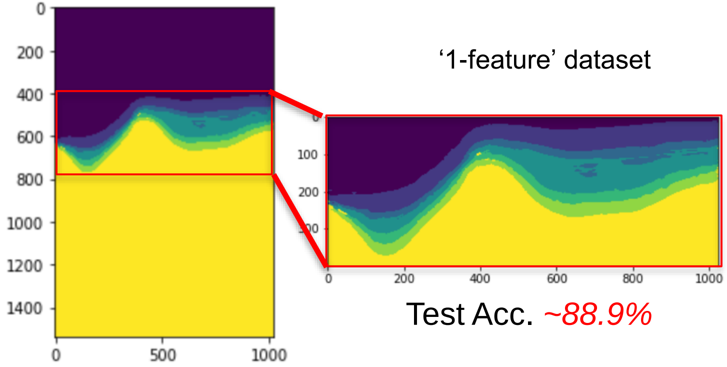


Human Interpretation

Good Geological continuity but low accuracy.

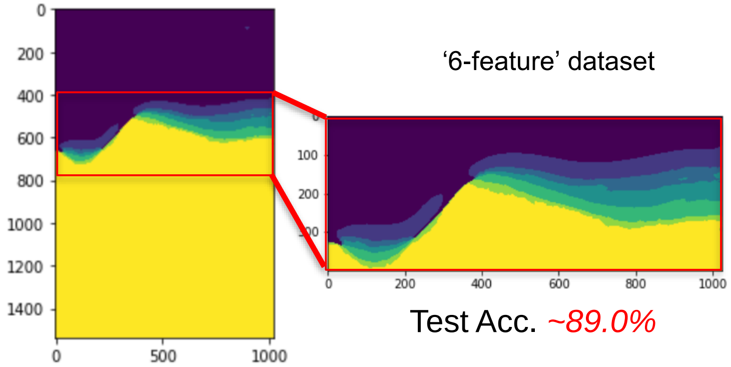
Hard to converge with a small number of datasets.

‘6-feature’ still wins on accuracy.



‘1-feature’ dataset

Test Acc. ~88.9%



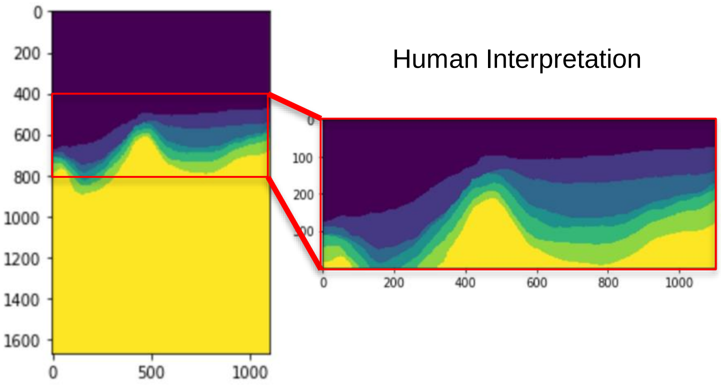
‘6-feature’ dataset

Test Acc. ~89.0%

Final Ensemble Model

Final Ensemble Model and Result

Final Ensemble Model

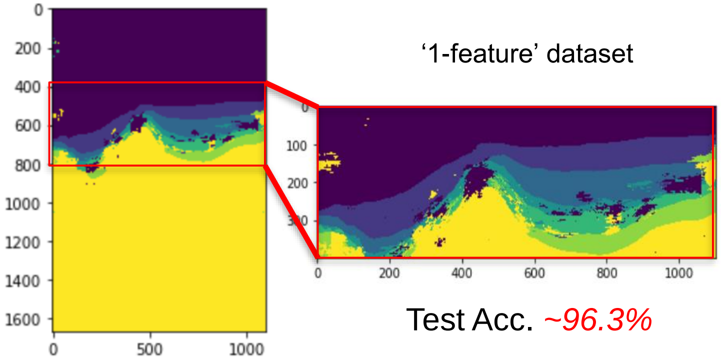


Human Interpretation

No Training.

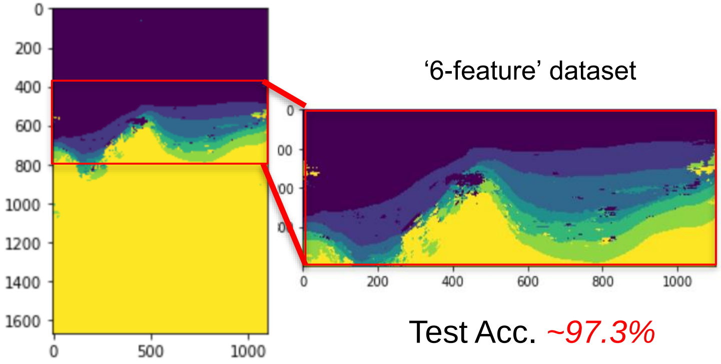
Combine the CNN models.

Combine strengths partly but not good enough .



‘1-feature’ dataset

Test Acc. ~96.3%



‘6-feature’ dataset

Test Acc. ~97.3%

Summary

On ‘1-feature’ dataset

	Training	Test
Feed-forward DNN model	N/A	N/A
Mini-patch CNN model (OptimizedNet)	99.4%	96.9%
Mini-patch CNN model (ResNet 34)	99.7%	95.2%
Encoder-decoder model	N/A	88.9%
Final ensemble model	N/A	96.3%

On ‘6-feature’ dataset

	Training	Test
Feed-forward DNN model	35.2%	9.0%
Mini-patch CNN model (OptimizedNet)	99.3%	97.7%
Mini-patch CNN model (ResNet 34)	99.6%	96.7%
Encoder-decoder model	N/A	89.0%
Final ensemble model	N/A	97.3%

After Training

Discussion

- Misclassified pixels:
  - Data Quality
  - Geological Complexity
  - Neural Network Design

- Final Ensemble Model has lower accuracy than some of its member models:
  - Number of member models
  - Quality of member models

Conclusion

- High accuracy and geological continuity achieved.

- ‘6-feature’ > ‘1-feature’.

- Better score could certainly be achieved

Imperial College London

Thanks !