Accelerating the Next Generation of Seismic Interpretation (S9479)

Yunzhi Shi, Graduate Research Assistant

The University of Texas at Austin

Bureau of Economic Geology

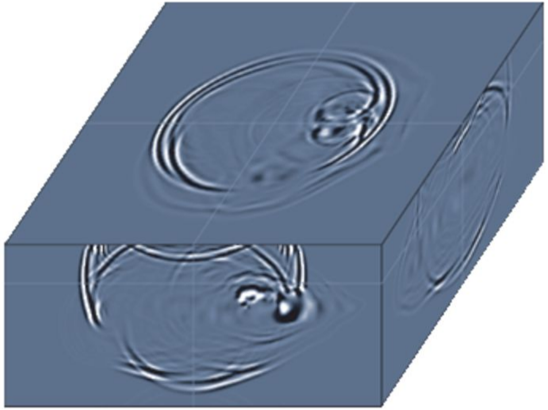
TCCS
Texas Consortium for Computational Seismology

About us - TCCS

Dr. Sergey B. Fomel

Texas Consortium for Computational Seismology

The **Texas Consortium for Computational Seismology** is a joint initiative of the Bureau of Economic Geology (BEG) and the Center for Numerical Analysis at the Institute for Computational Engineering and Science (ICES) at The University of Texas at Austin. Its mission is to address the most important and challenging research problems in computational geophysics as experienced by the energy industry while educating the next generation of research geophysicists and computational scientists.



Examples of research challenges include:

- Estimating seismic velocities by using full waveform information.
- Identifying the most accurate and the most efficient seismic imaging algorithms while controlling the trade-off between accuracy and efficiency.
- Assisting the seismic interpreter by automating common interpretation tasks.

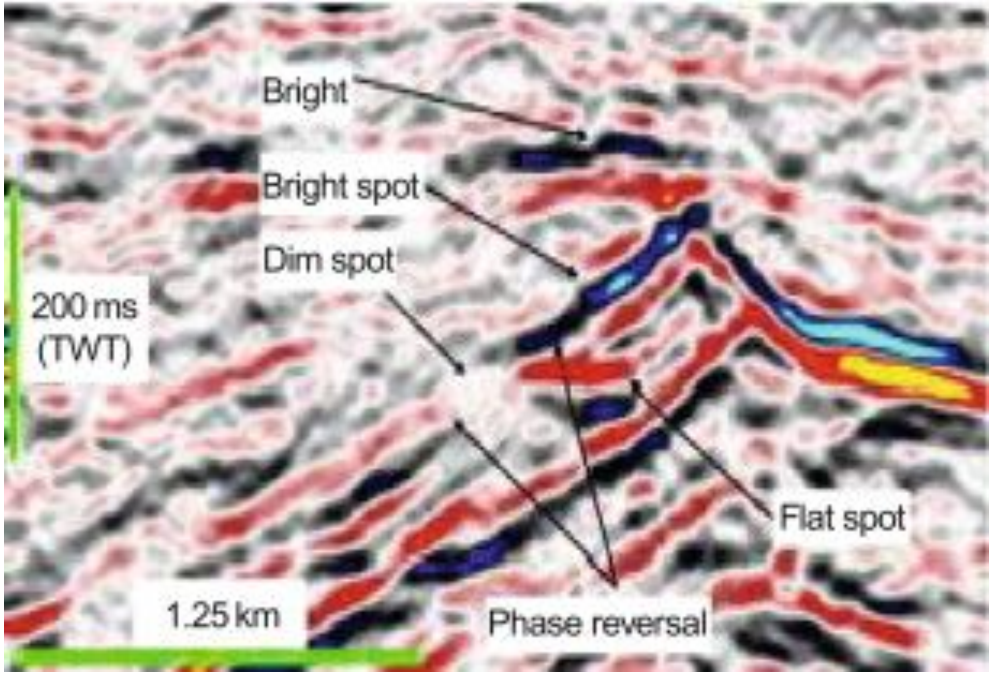
To learn more about joining the Texas Consortium for Computational Seismology, please click here.

About us - BEG

Bureau of Economic Geology

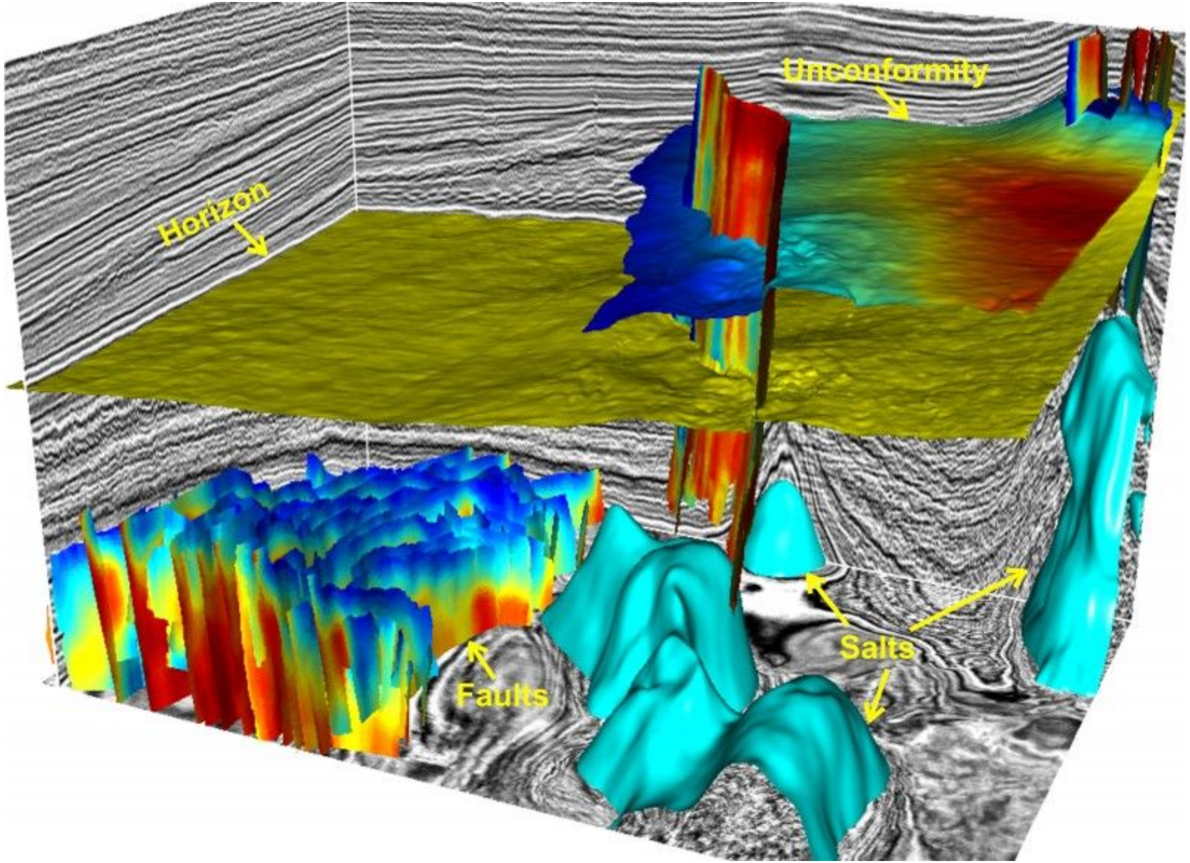
Established in 1909, the Bureau conducts research focusing on the intersection of energy, the environment, and the economy, where significant advances are being made tackling tough problems globally.

About seismic interpretation



Onajite E., 2014, *Understanding Seismic Interpretation Methodology*, in Seismic Data Analysis Techniques in Hydrocarbon Exploration

About seismic interpretation

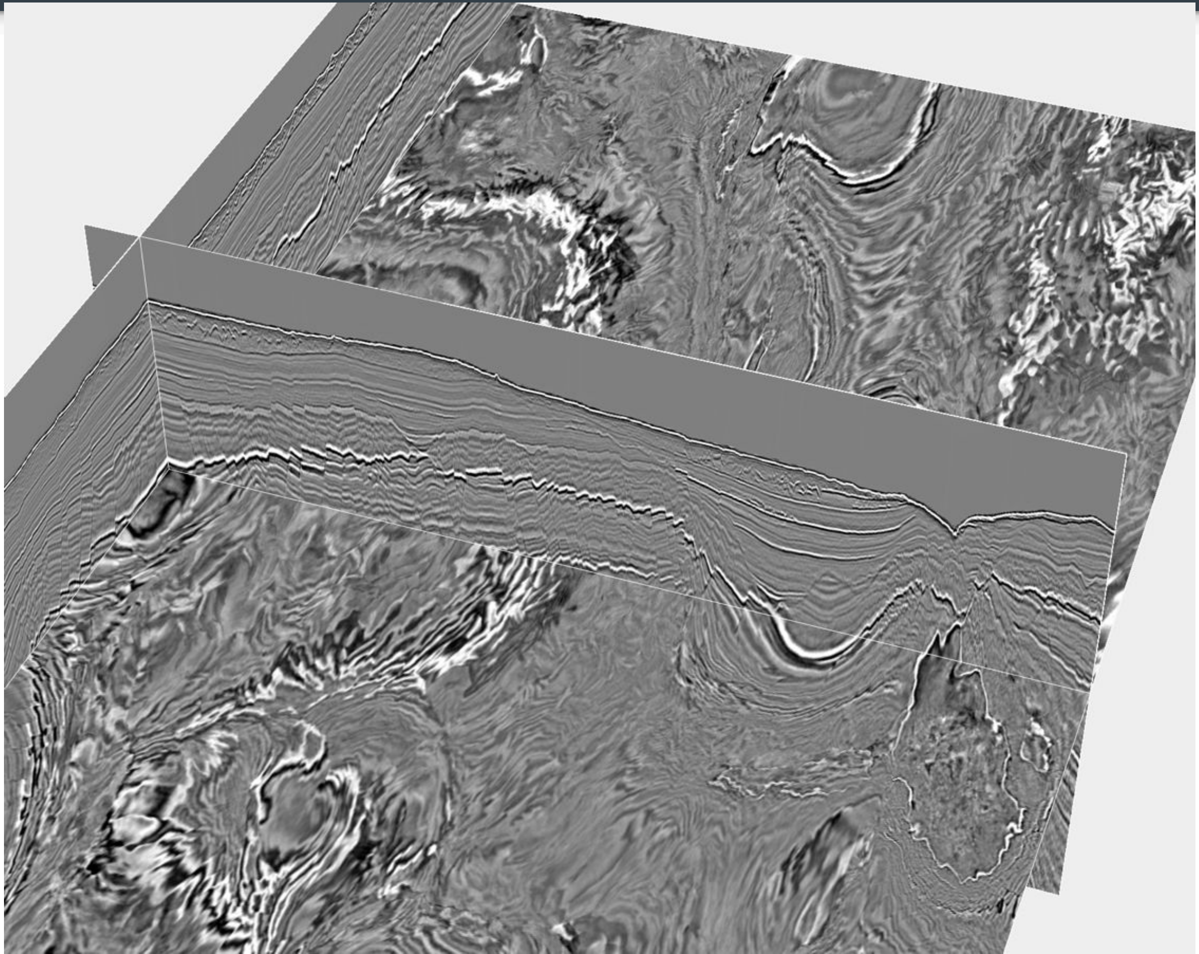


About seismic interpretation

OK …

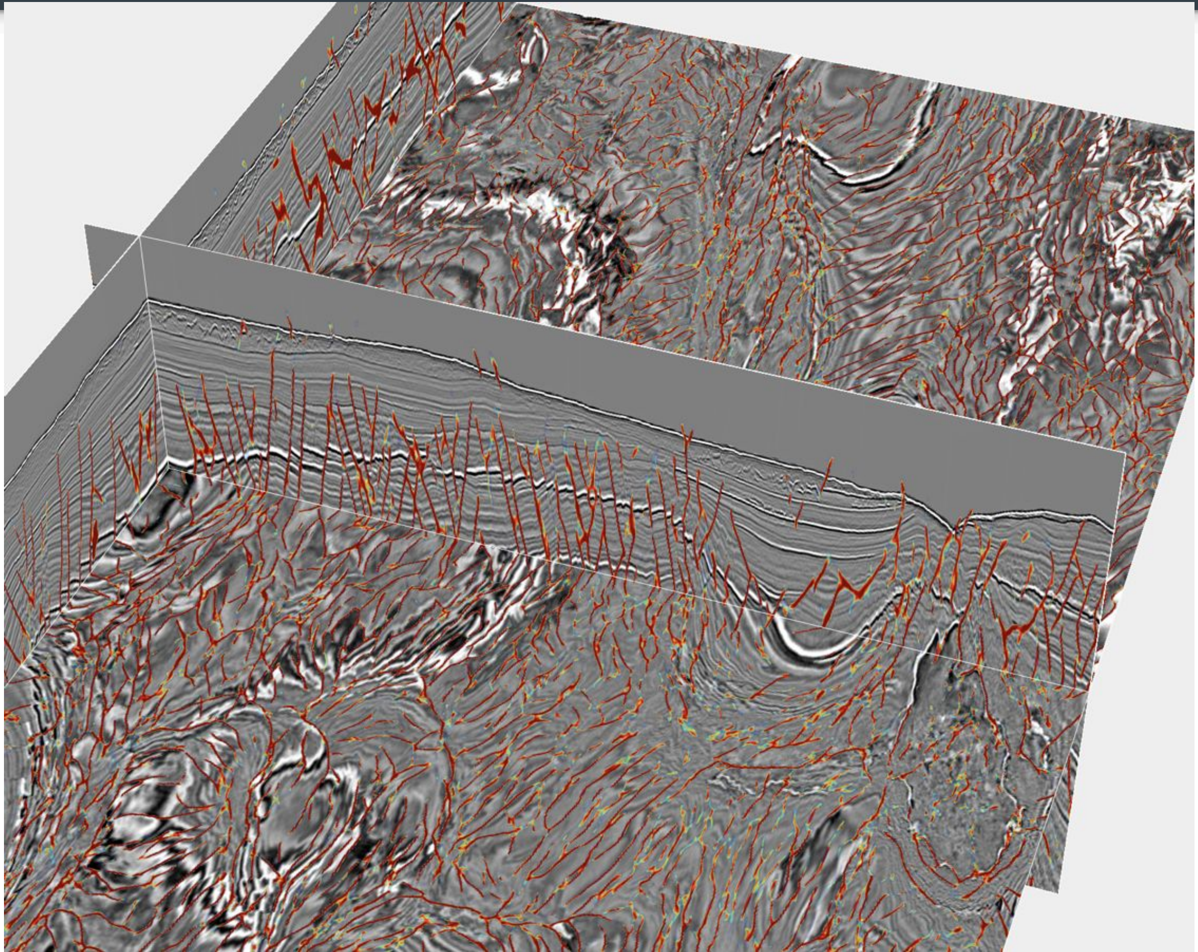
So what is THIS generation of seismic interpretation?

Example: Campos Basin, offshore Brazil



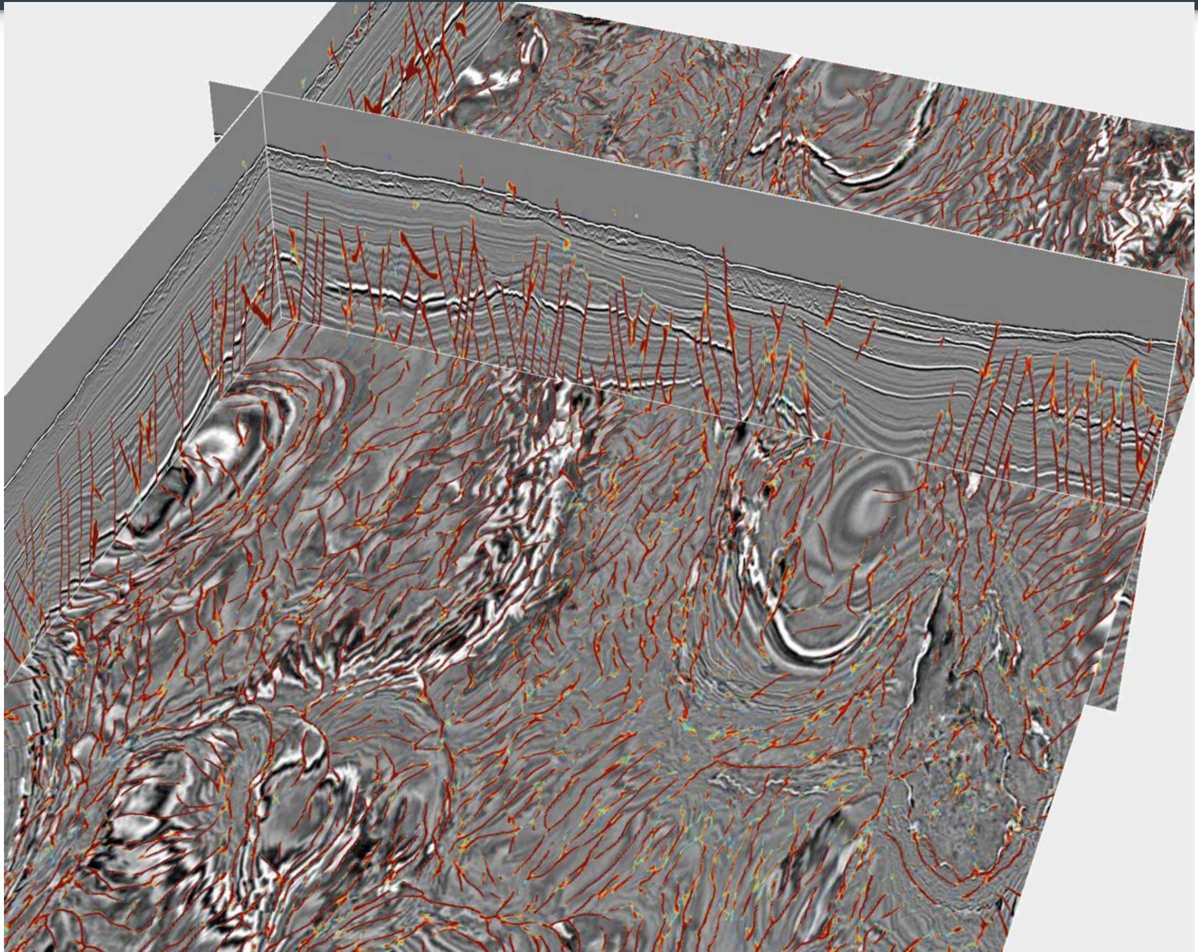
450 x 1950 x 1200 samples

Example: Campos Basin, offshore Brazil



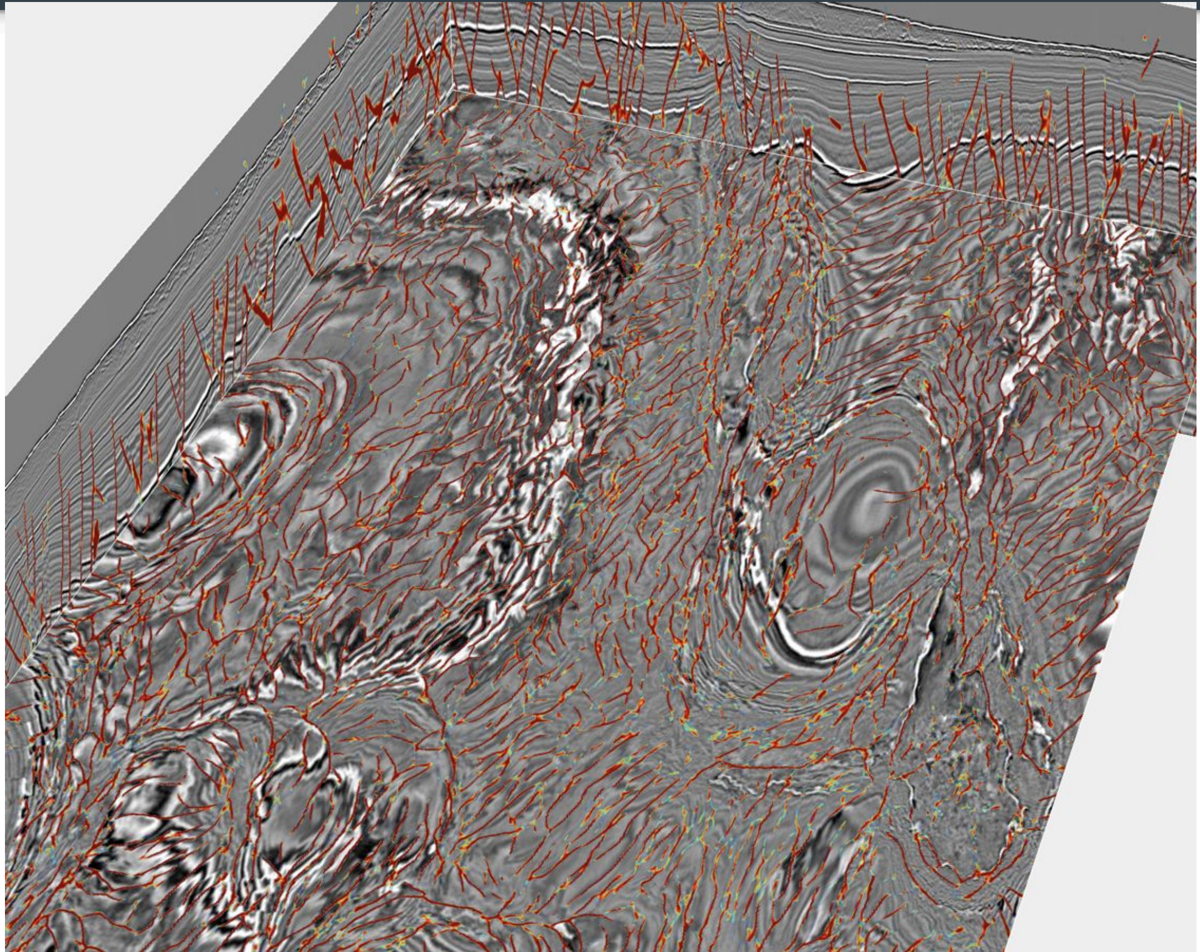
450 x 1950 x 1200 samples

Example: Campos Basin, offshore Brazil



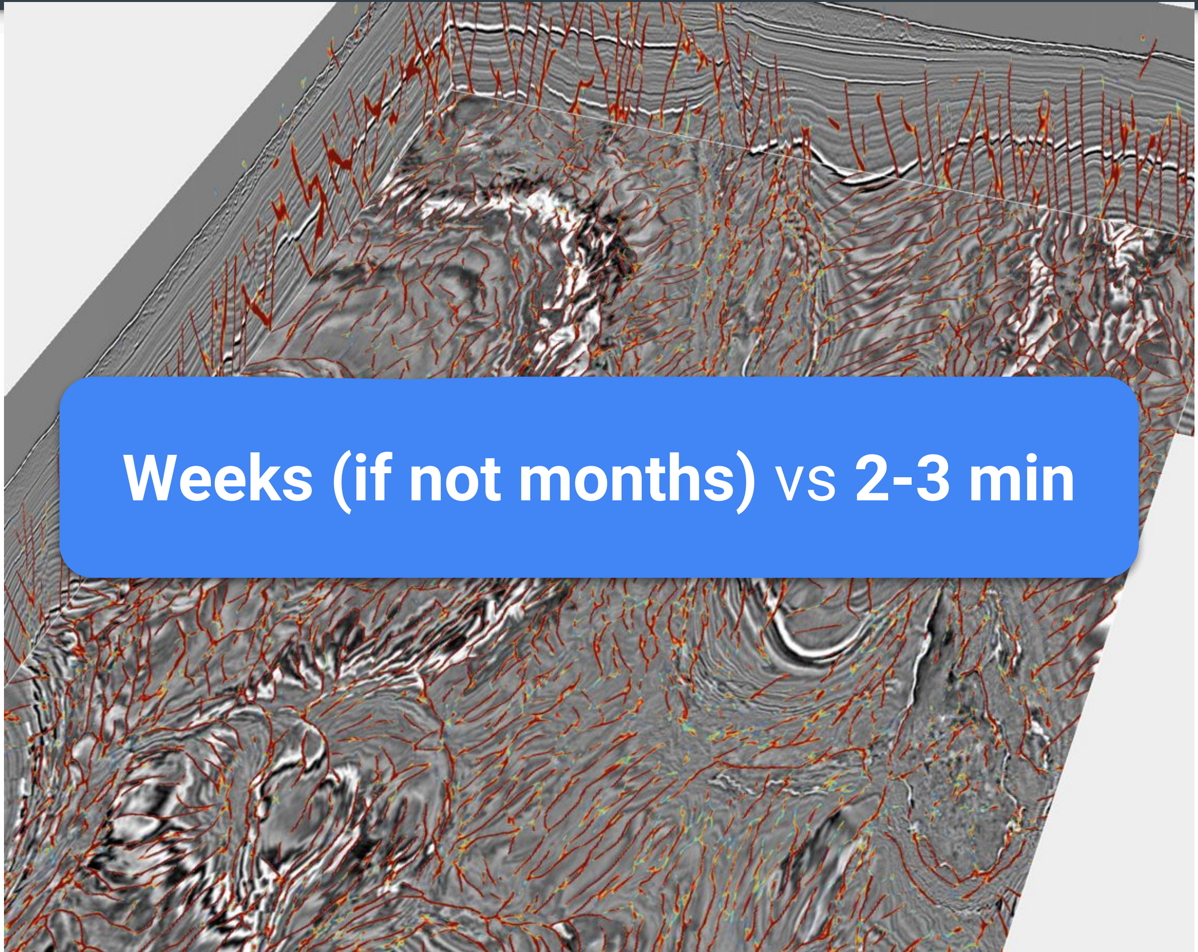
450 x 1950 x 1200 samples

Example: Campos Basin, offshore Brazil



450 x 1950 x 1200 samples

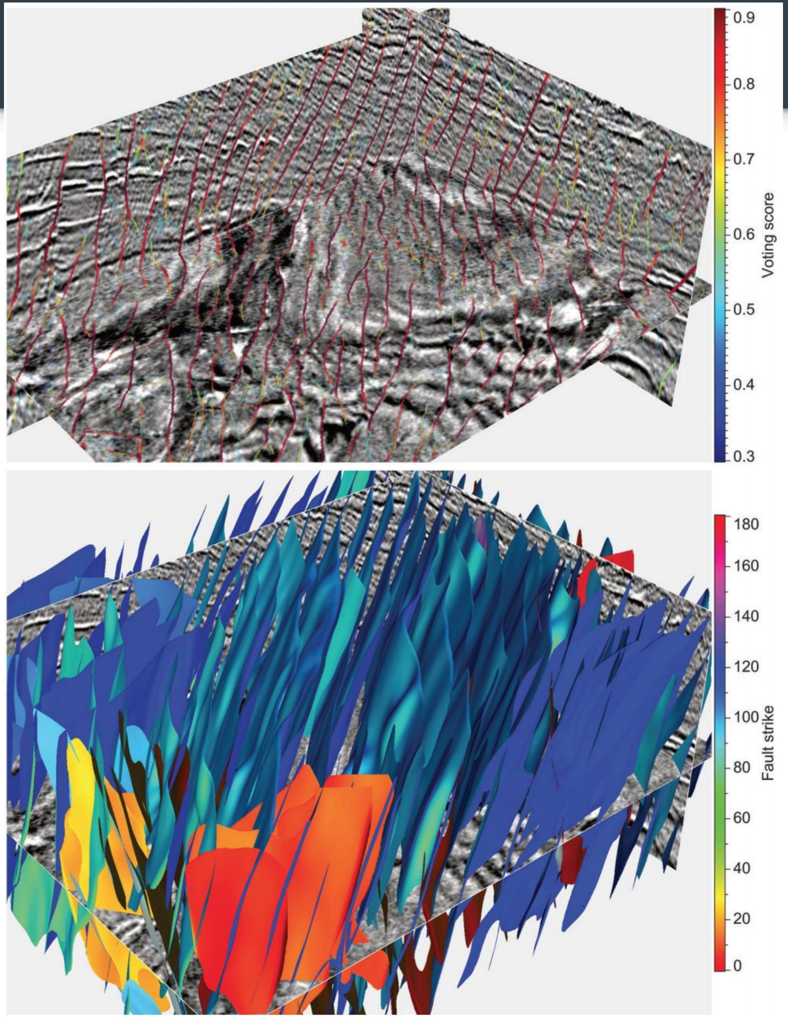
Example: Campos Basin, offshore Brazil



450 x 1950 x 1200 samples

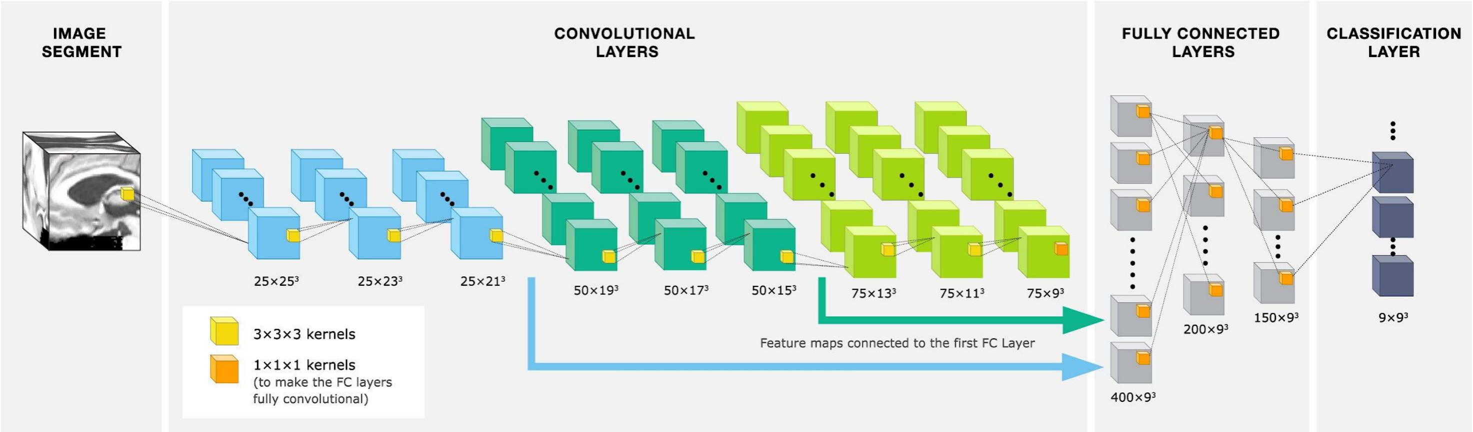
About seismic interpretation

Are we really doing this in a proper 3D way?



Why deep learning?

- Everybody knows deep learning is fast in application now …
- But it also brings a new 3D/4D/… perspective to seismic interpretation!



Dolz, J., et al., 2018, *3D fully convolutional networks for subcortical segmentation in MRI: A large-scale study*, NeuroImage

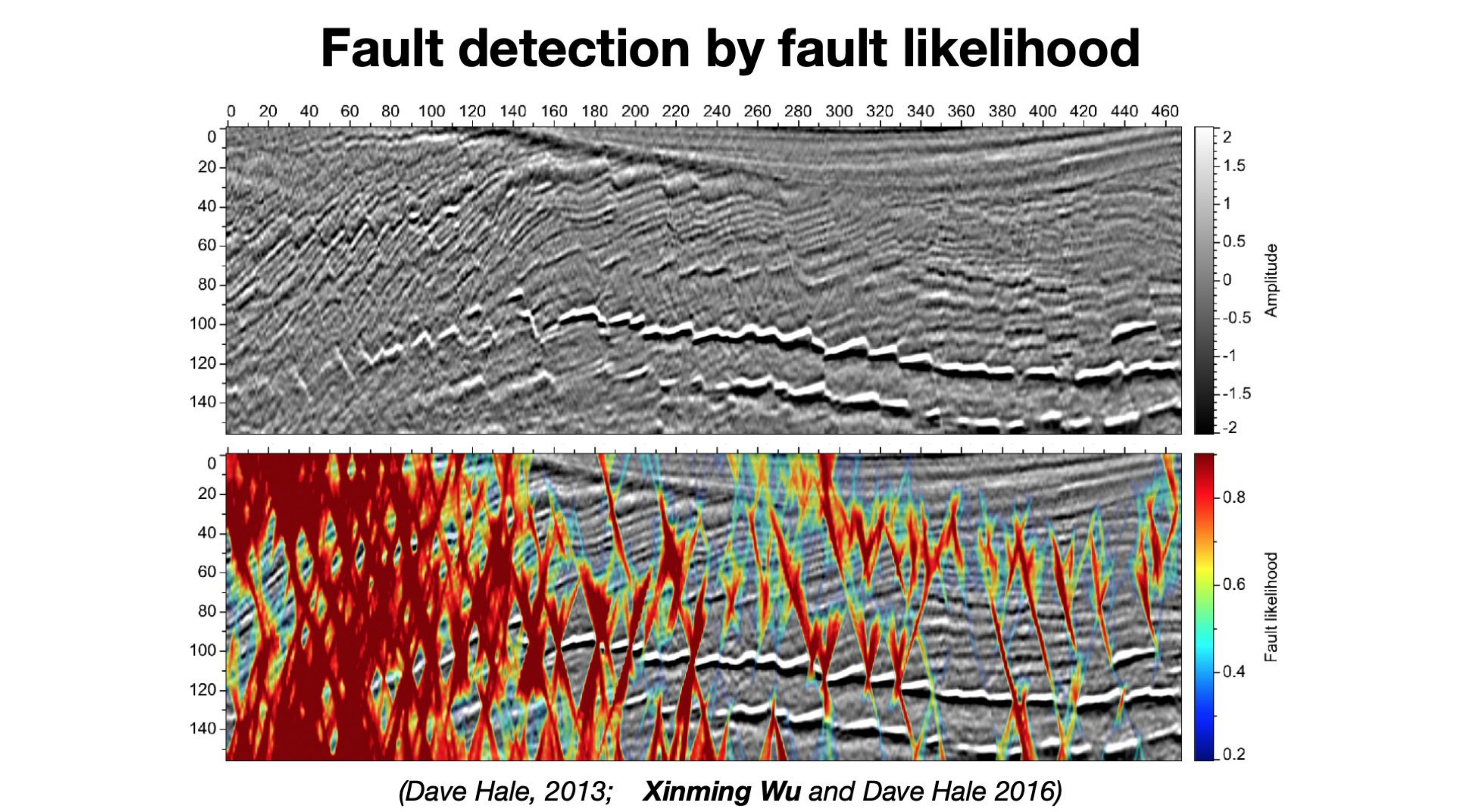
Outline

- Simple fault classification
- Generating geophysical synthetic training data
- From classification to segmentation:
  - Fault segmentation
  - Salt body segmentation
  - Channel segmentation
- Tracking geobody in a recurrent style
- Predicting relative geological time (RGT)

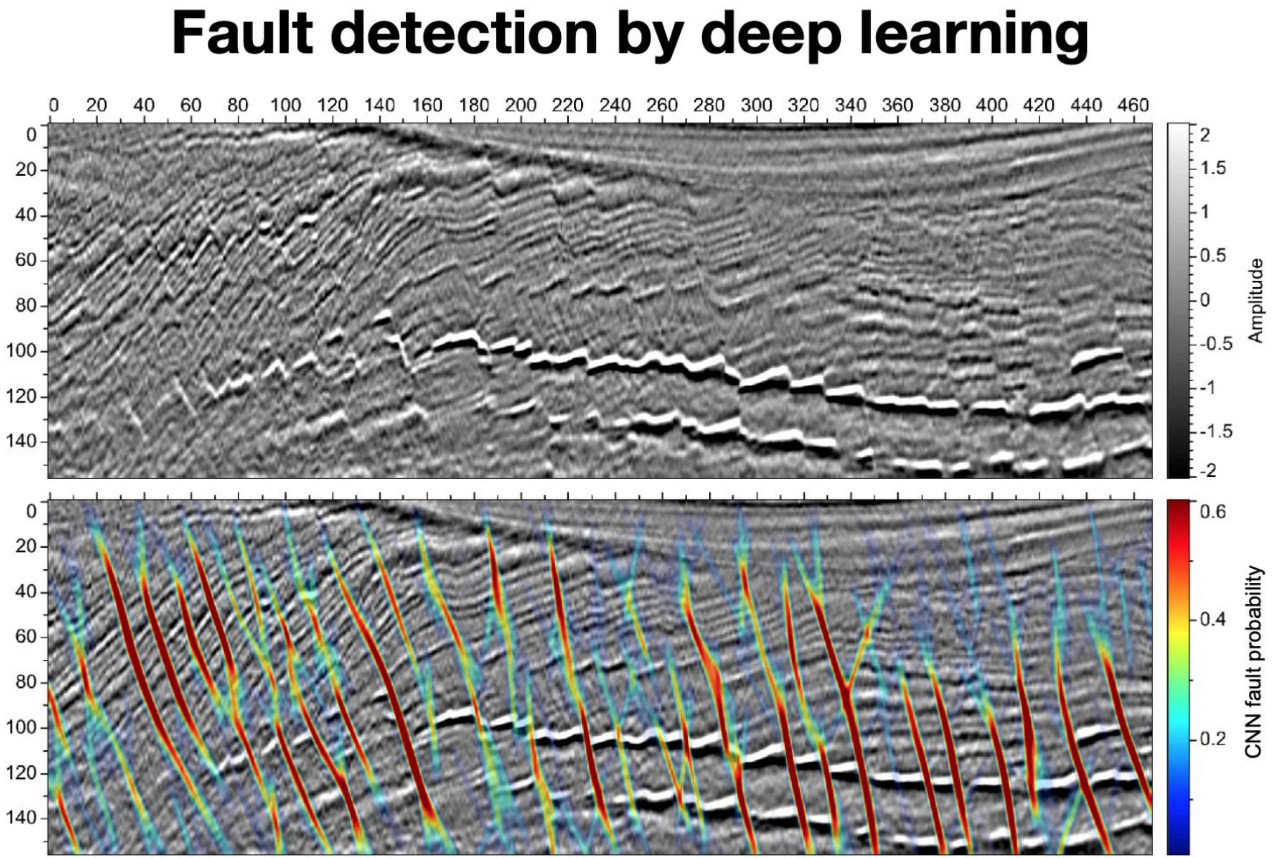
Outline

- **Simple fault classification**
- Generating geophysical synthetic training data
- From classification to segmentation:
  - Fault segmentation
  - Salt body segmentation
  - Channel segmentation
- Tracking geobody in a recurrent style
- Predicting relative geological time (RGT)

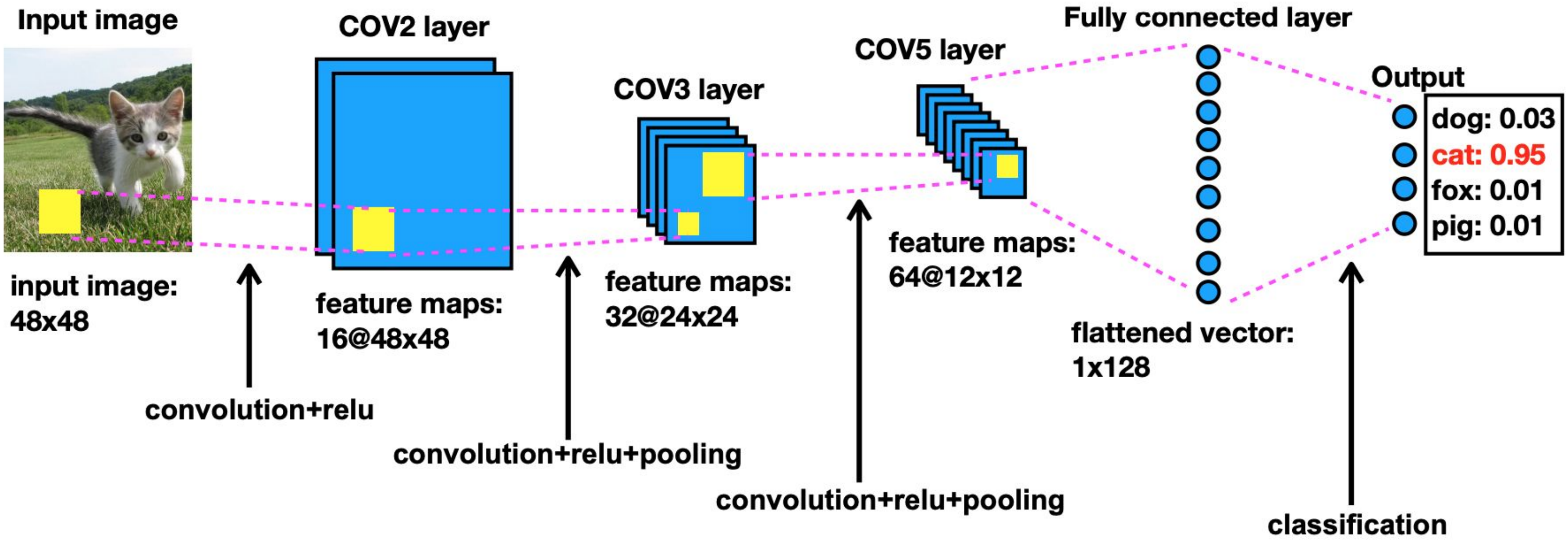
Simple fault classification



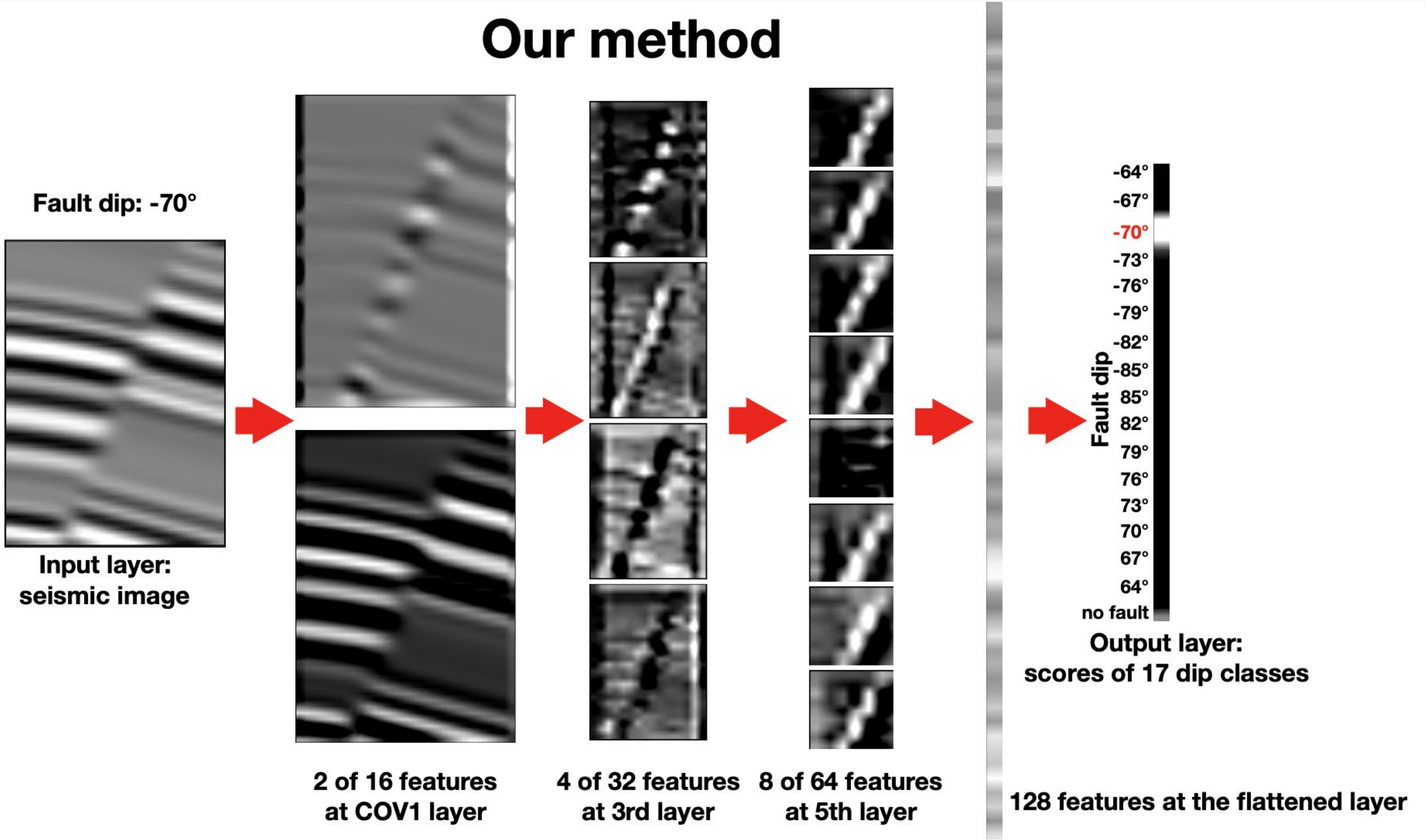
Simple fault classification



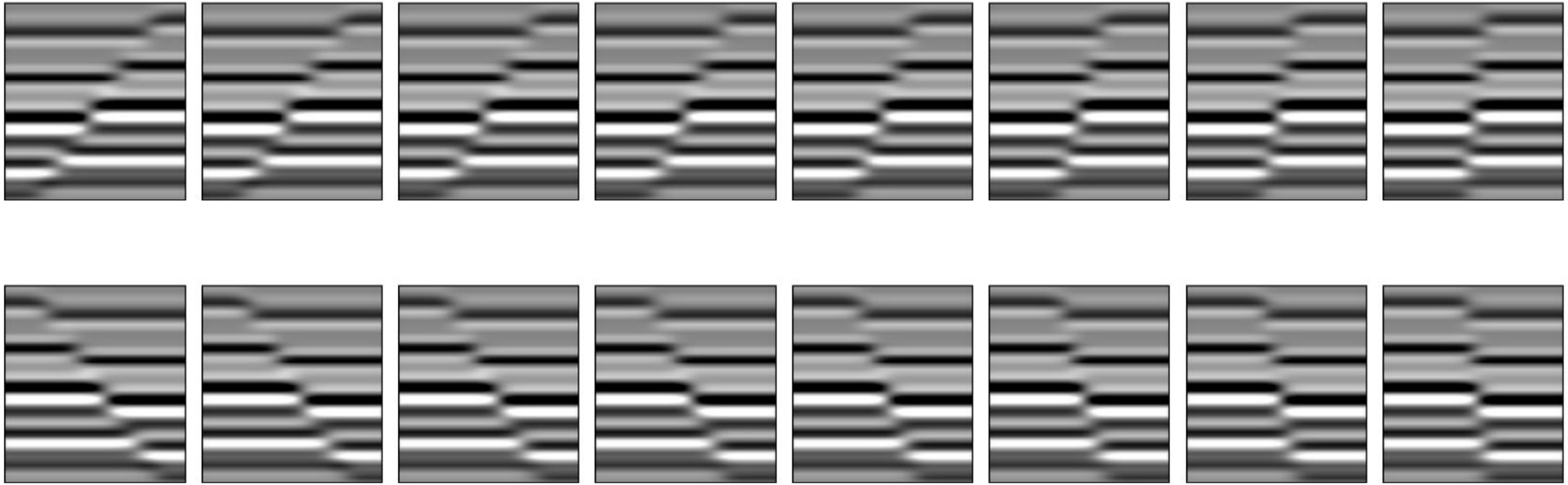
Simple fault classification

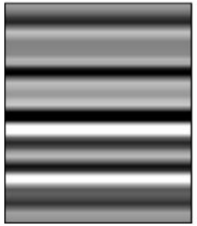


Simple fault classification



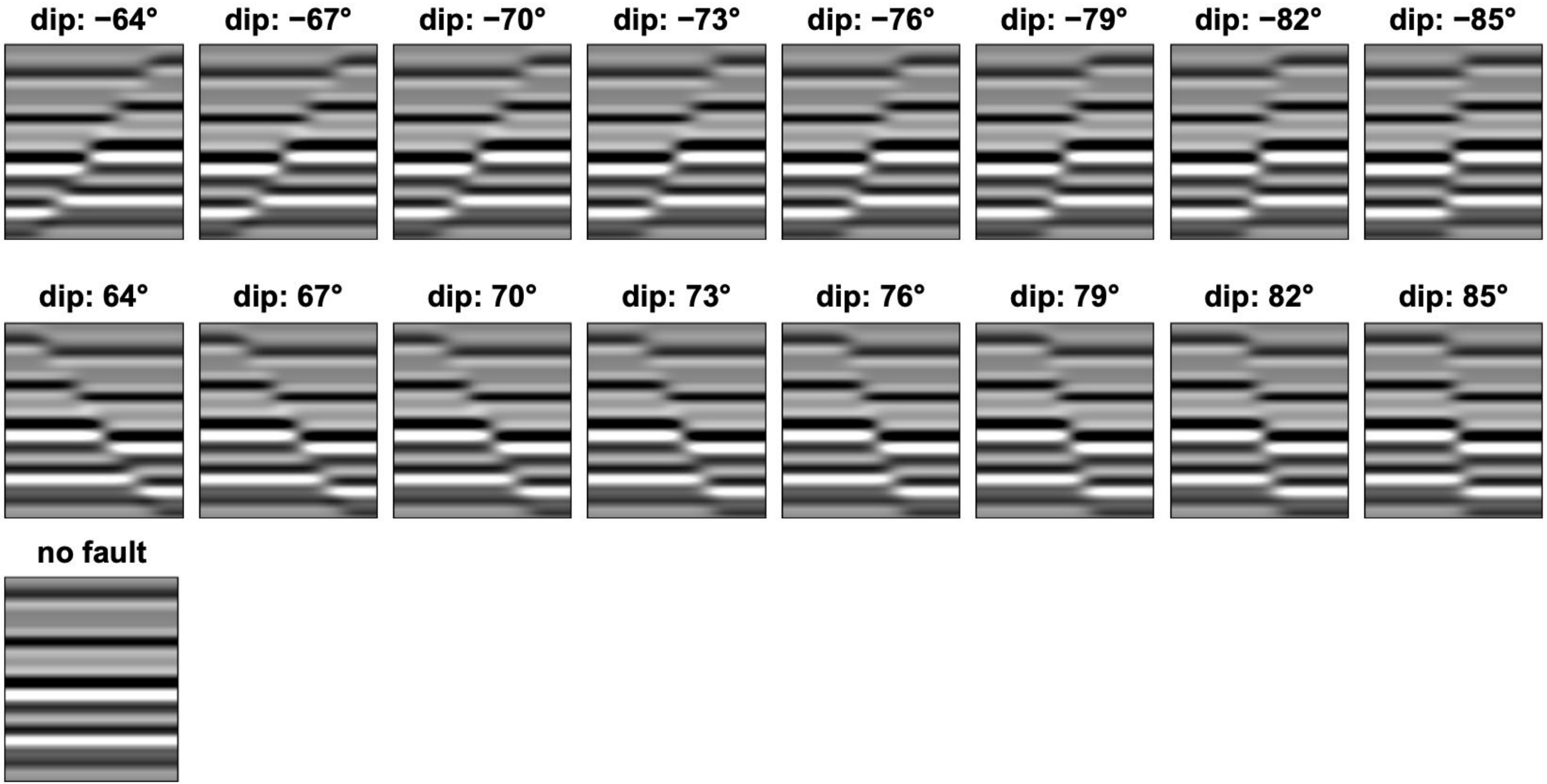
Simple fault classification



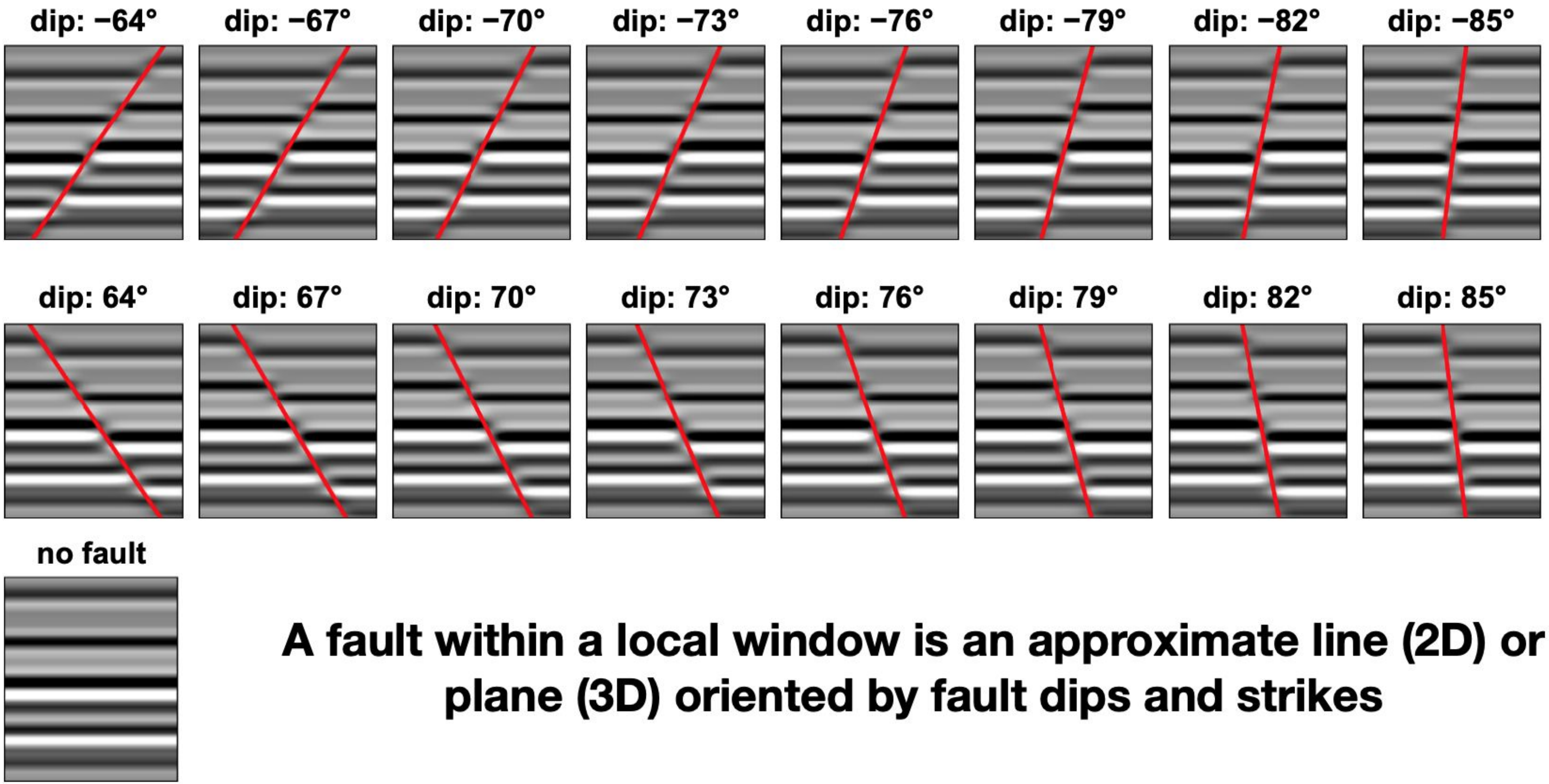


A fault within a local window is an approximate line (2D) or plane (3D) oriented by fault dips and strikes

Simple fault classification

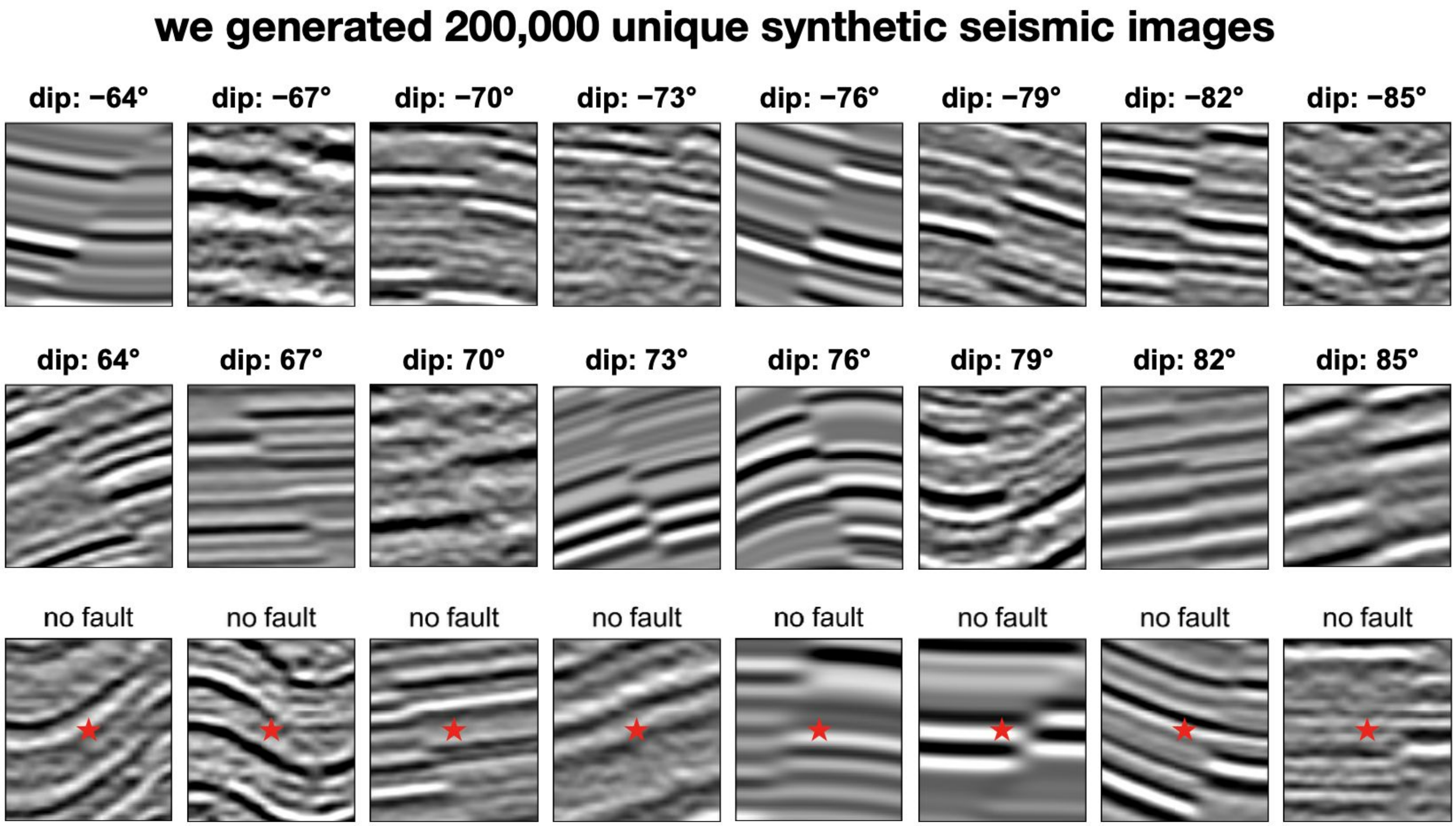


Simple fault classification

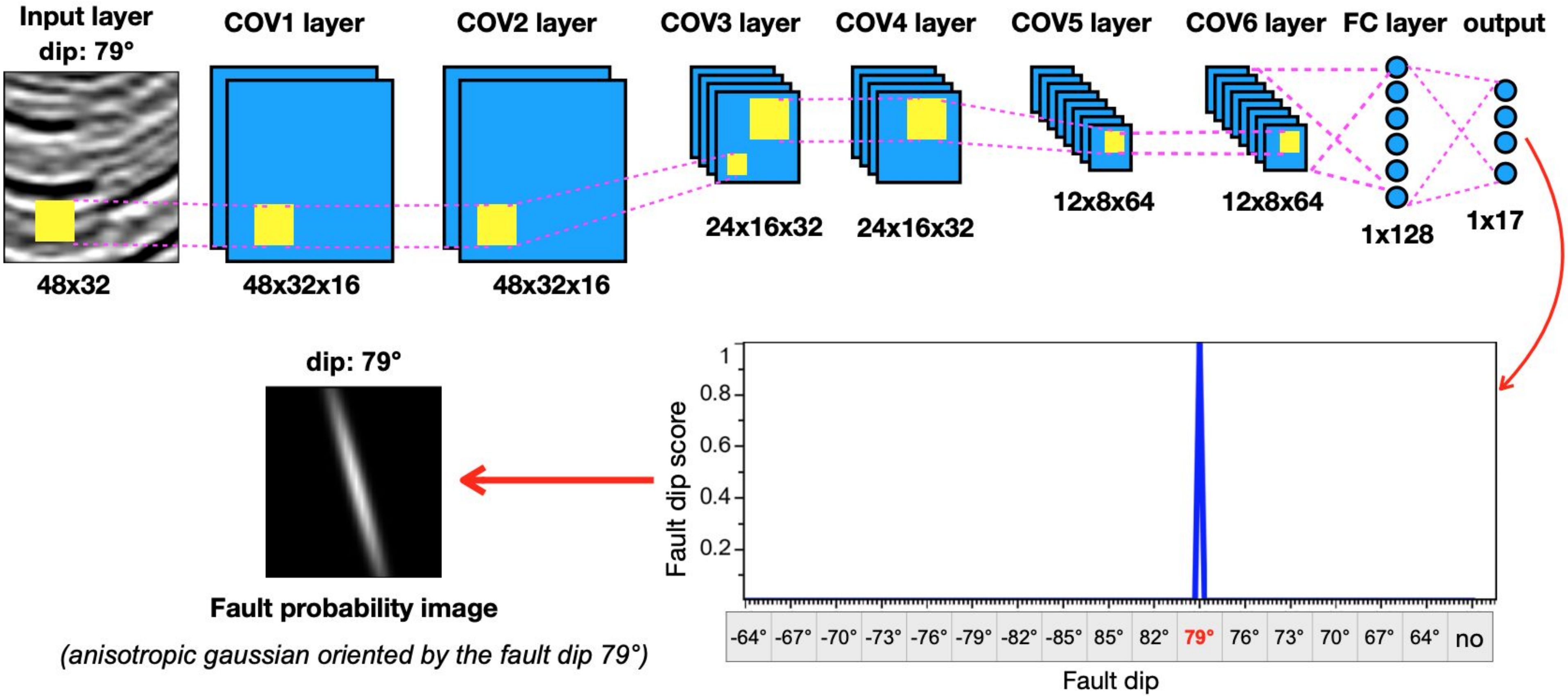


A fault within a local window is an approximate line (2D) or plane (3D) oriented by fault dips and strikes

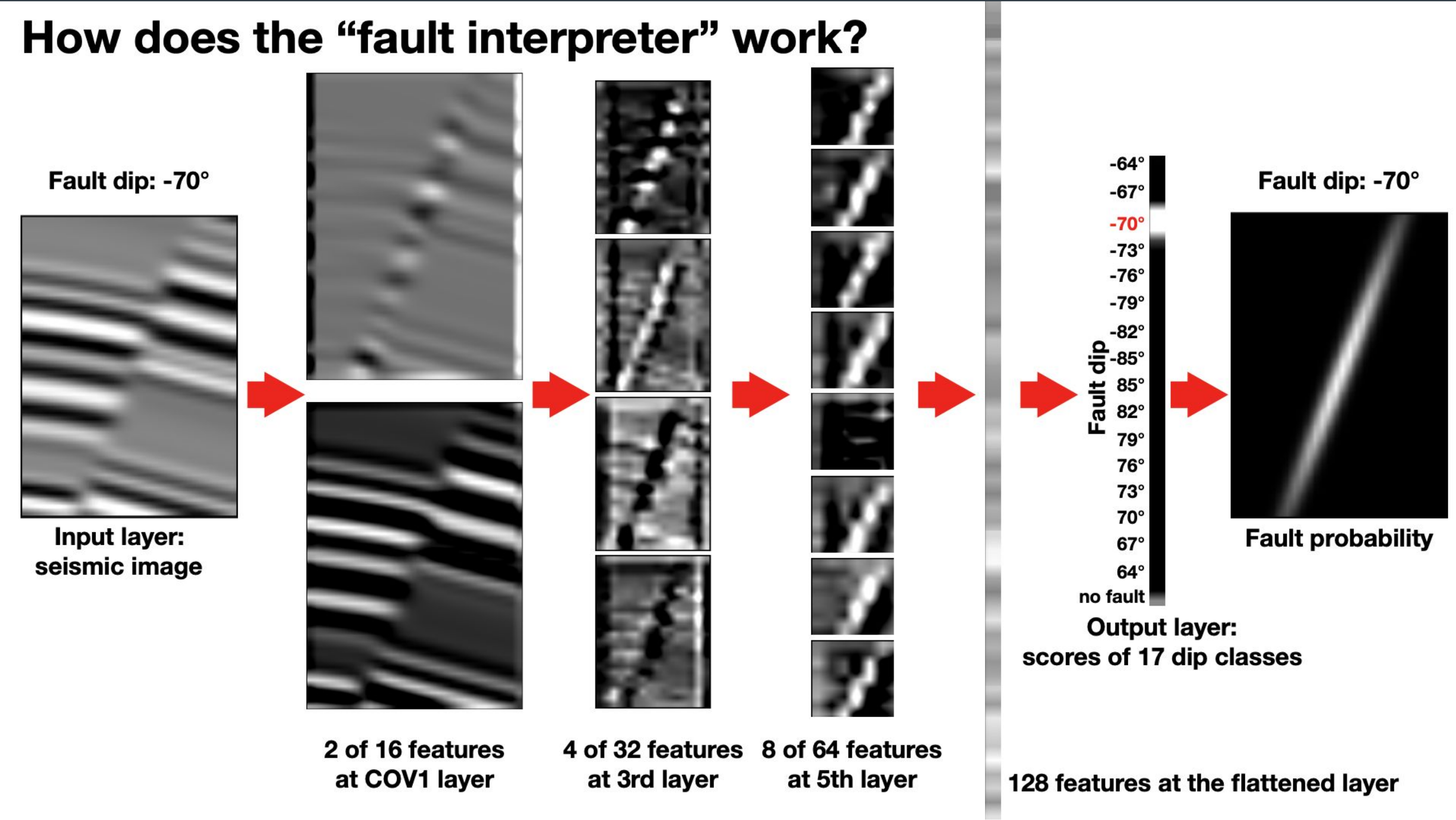
Simple fault classification



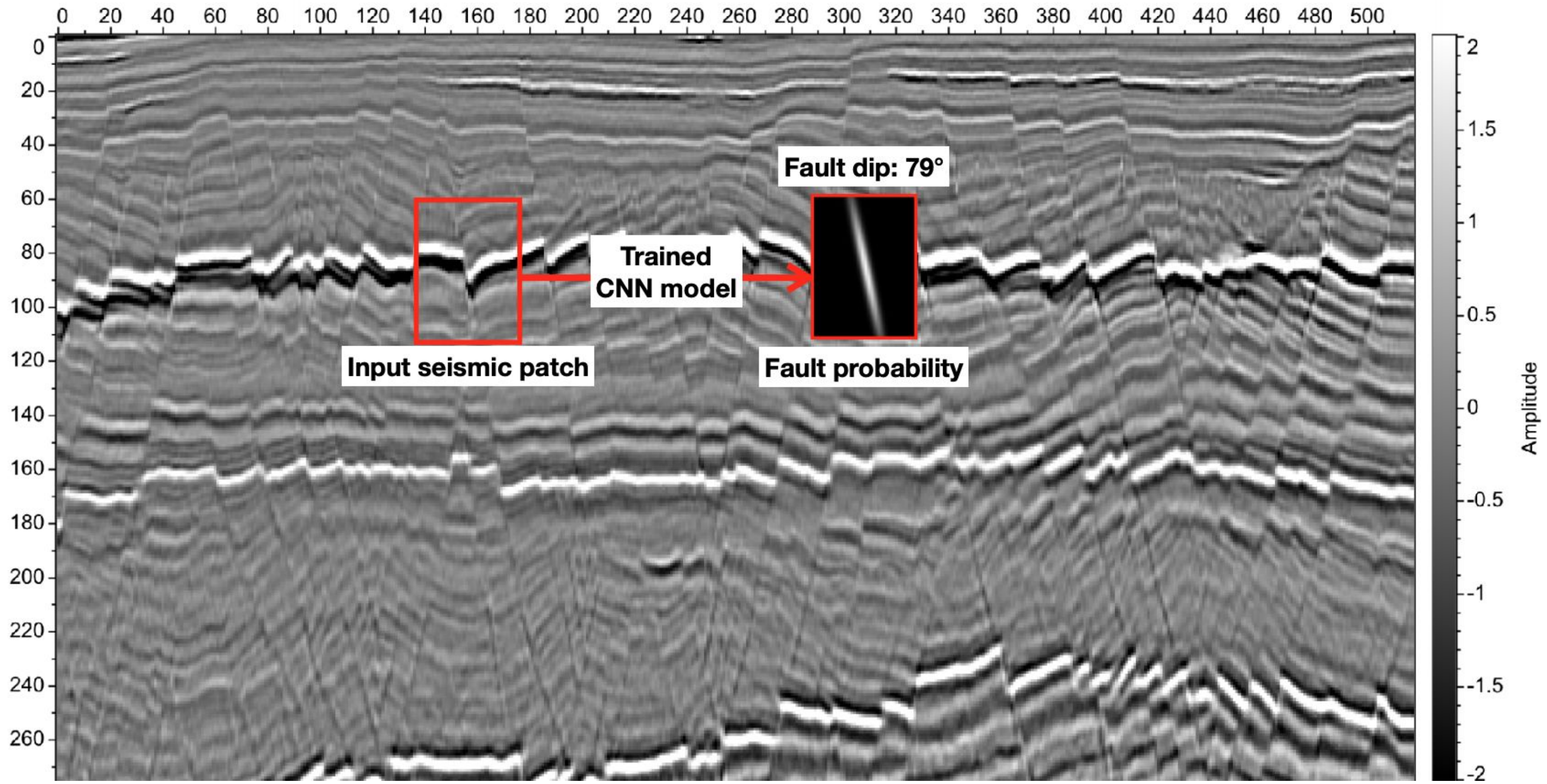
Simple fault classification



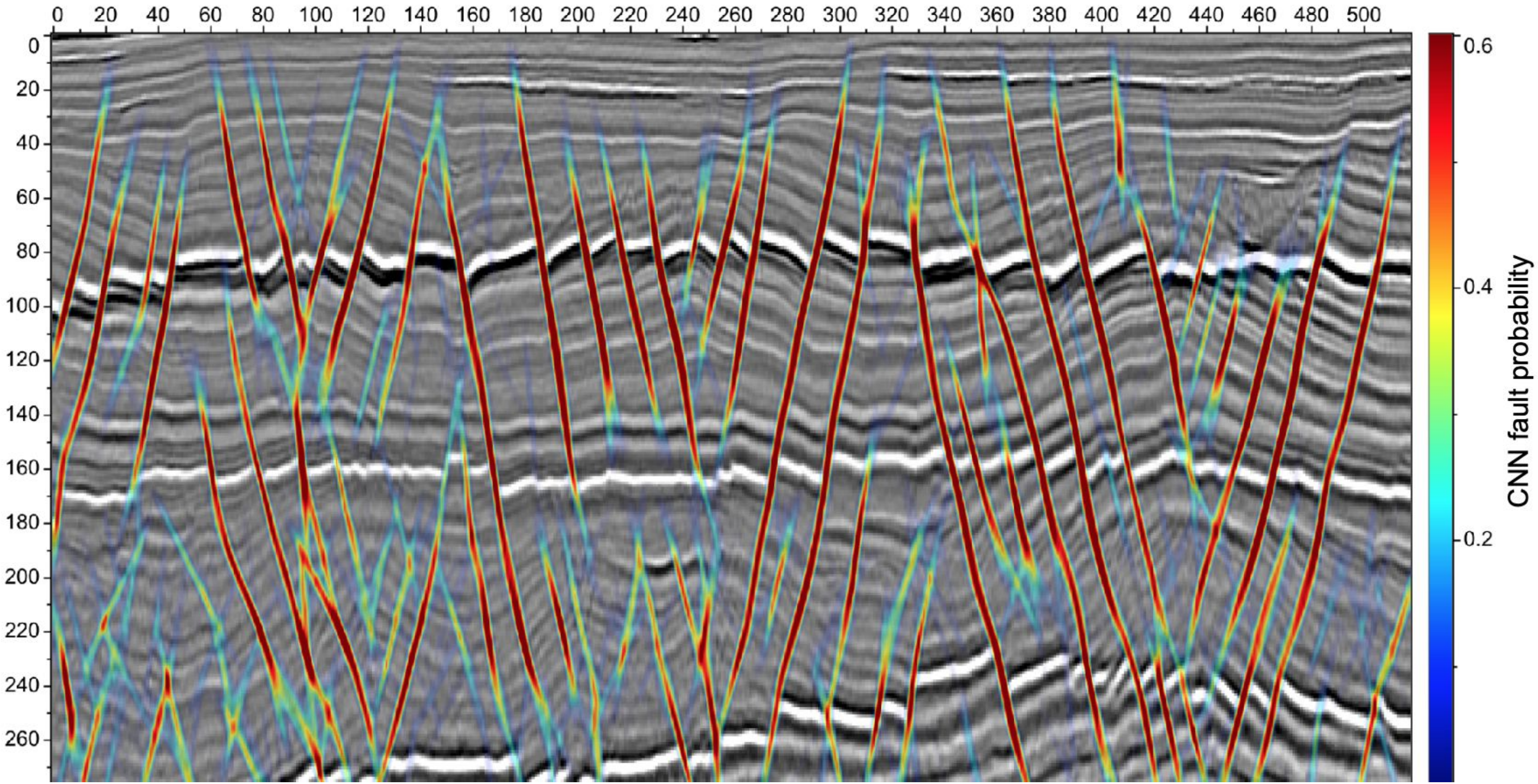
Simple fault classification



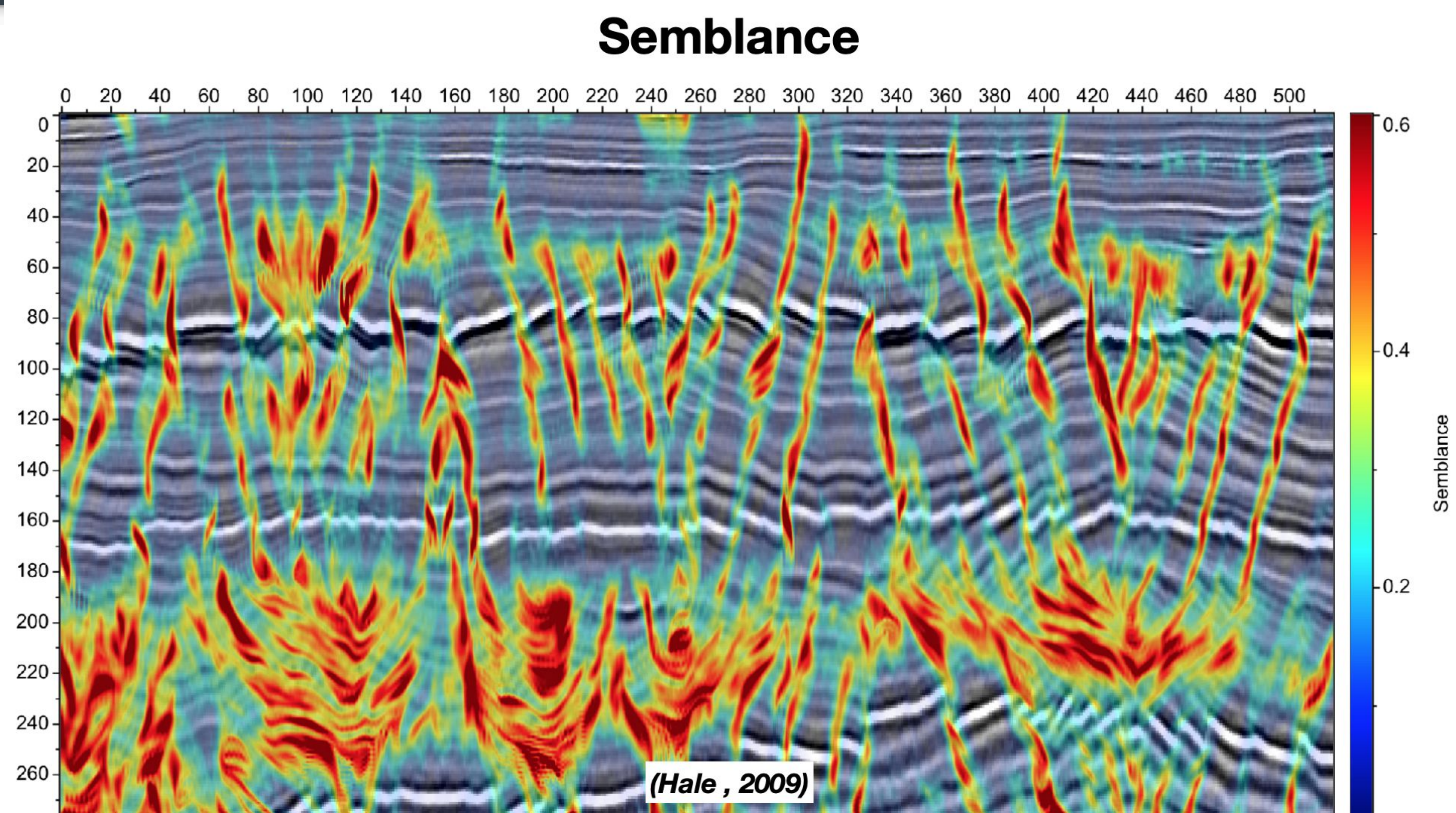
Simple fault classification



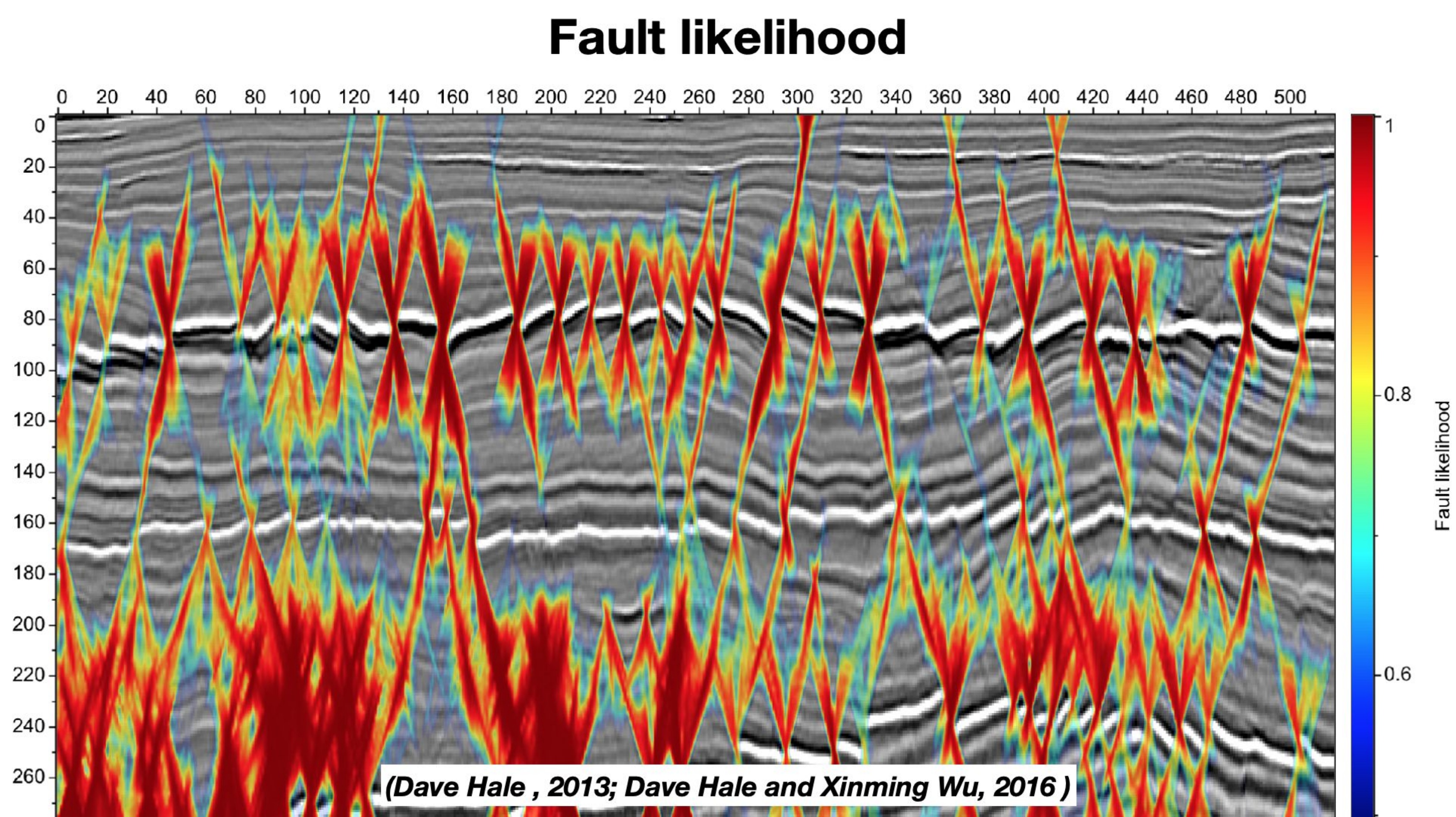
Simple fault classification



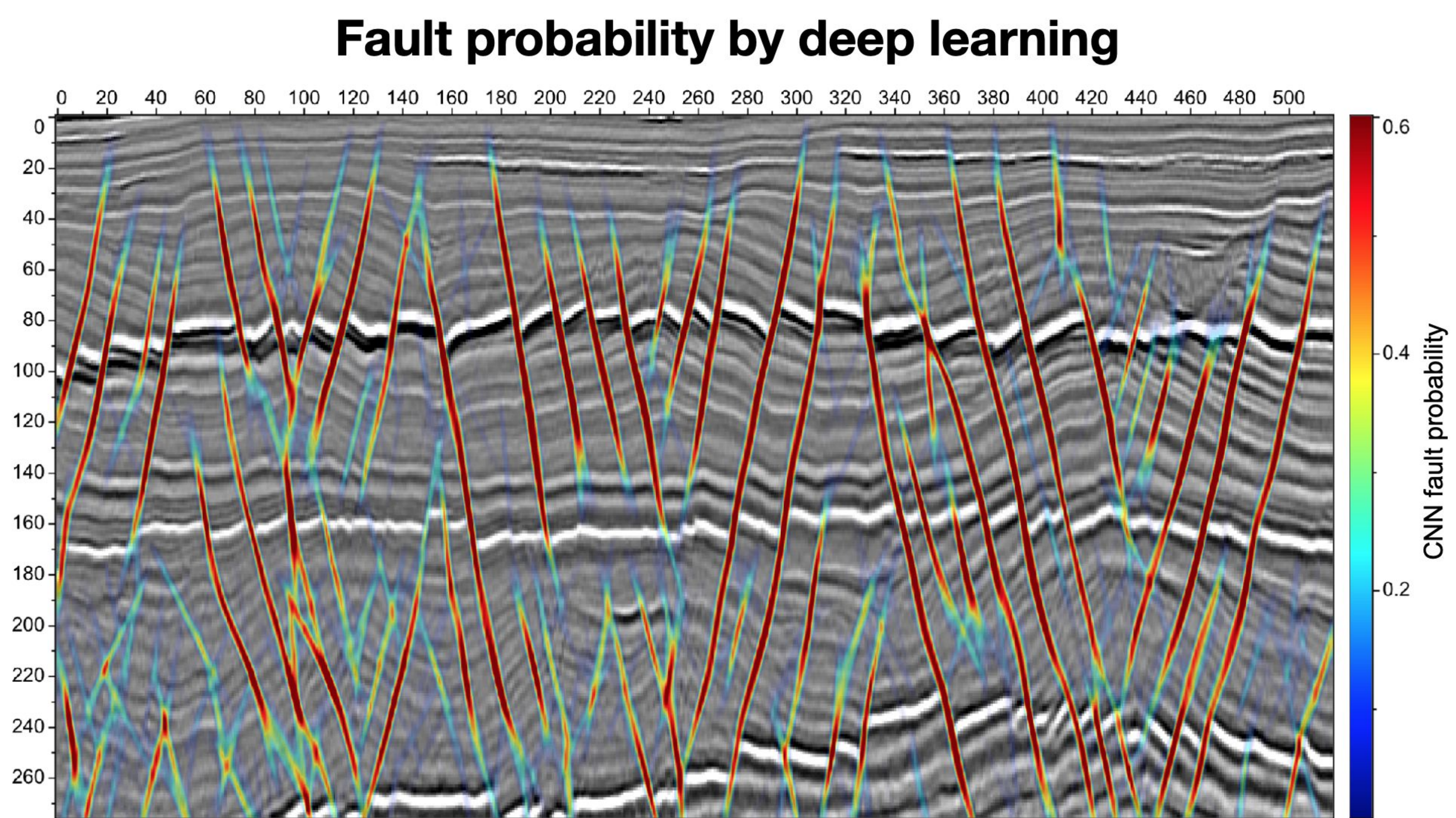
Simple fault classification



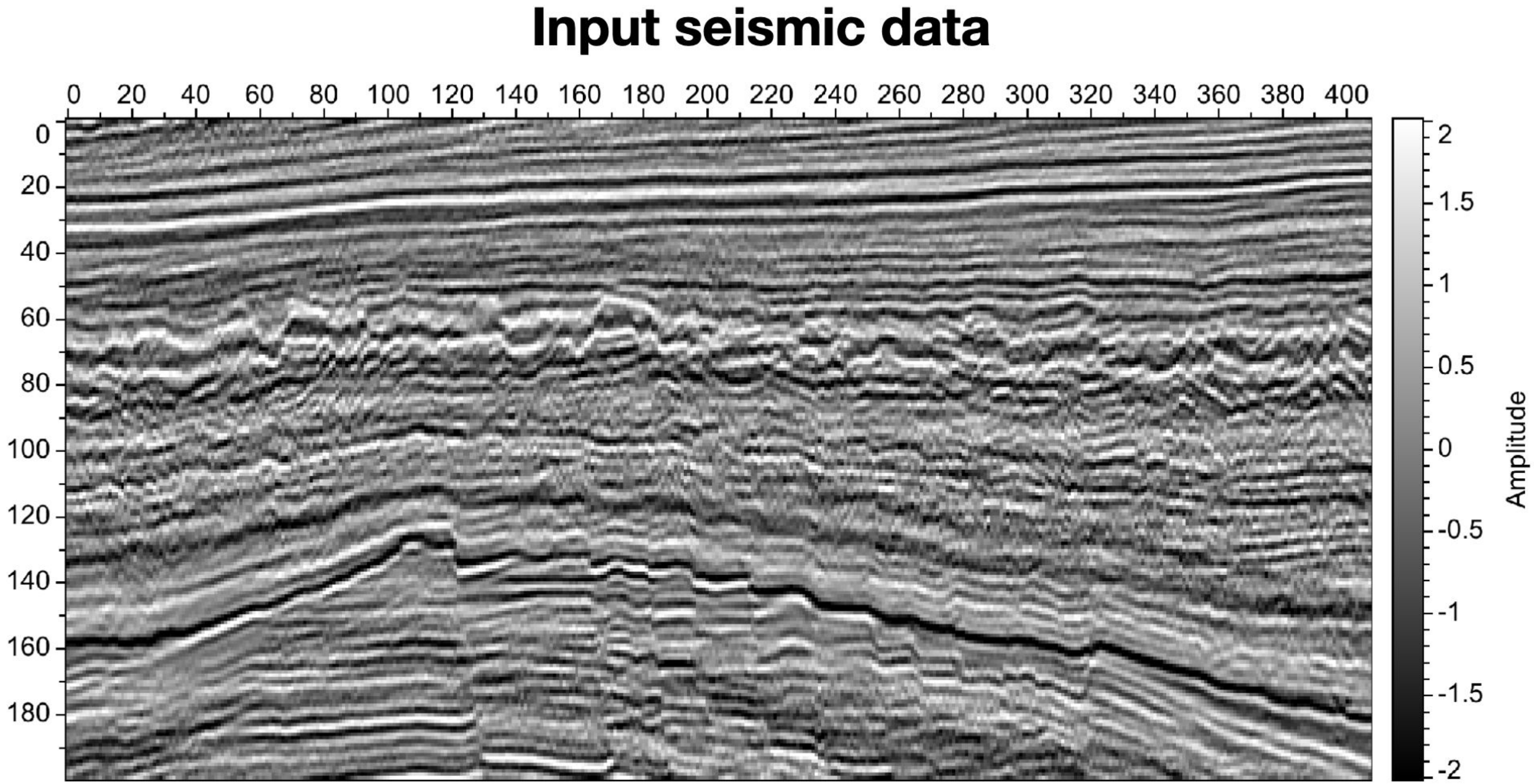
Simple fault classification



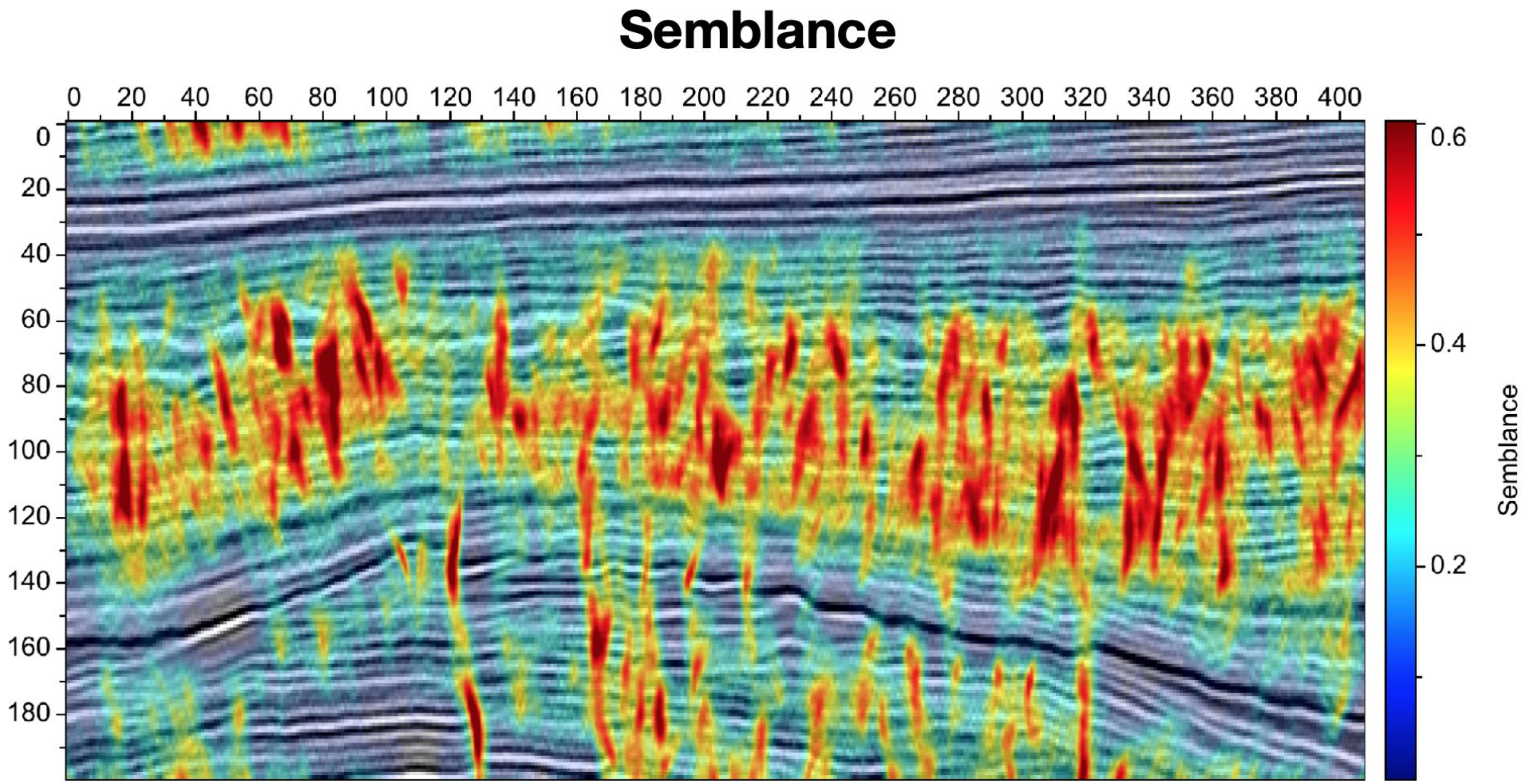
Simple fault classification



Simple fault classification

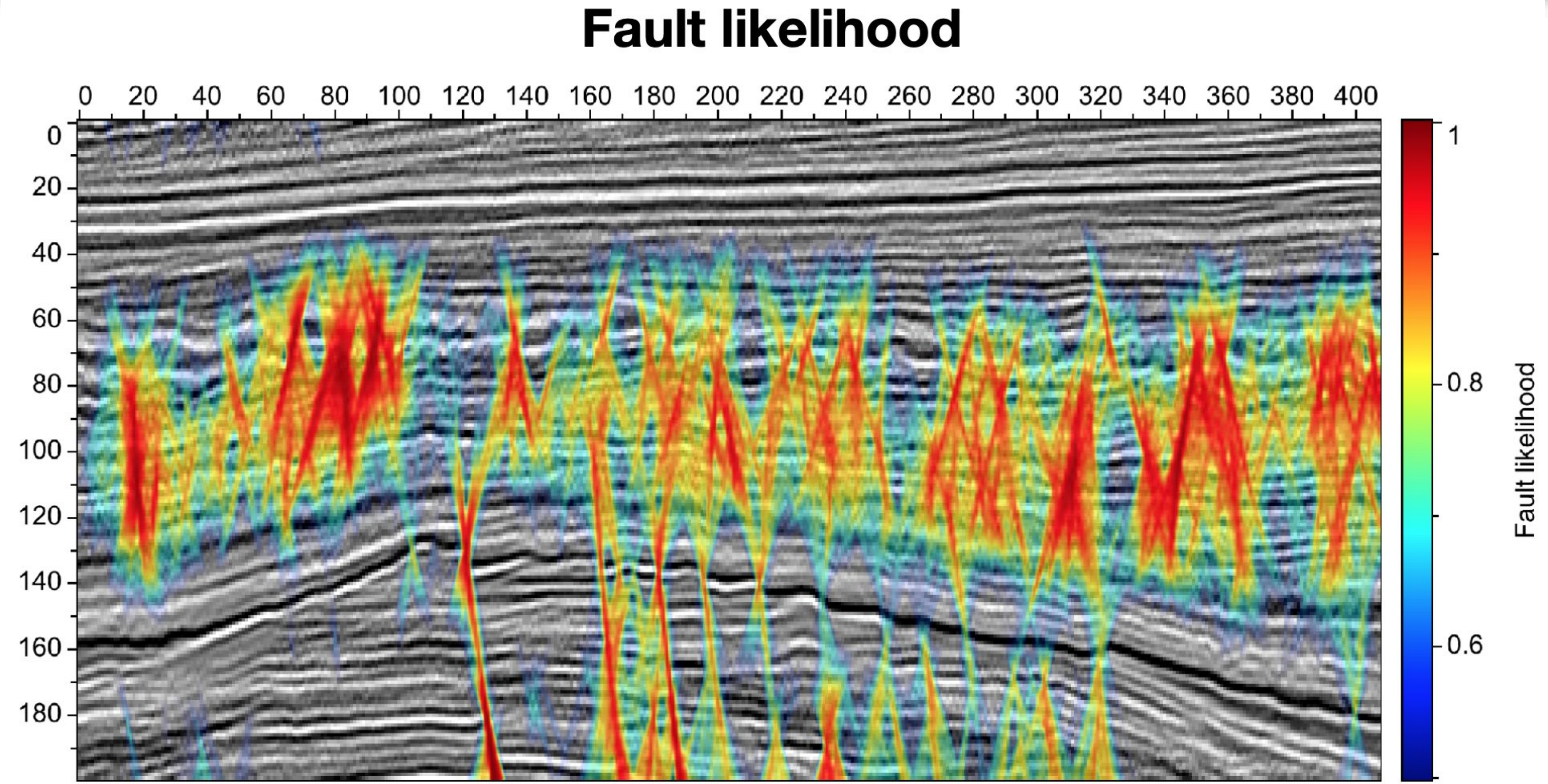


Simple fault classification



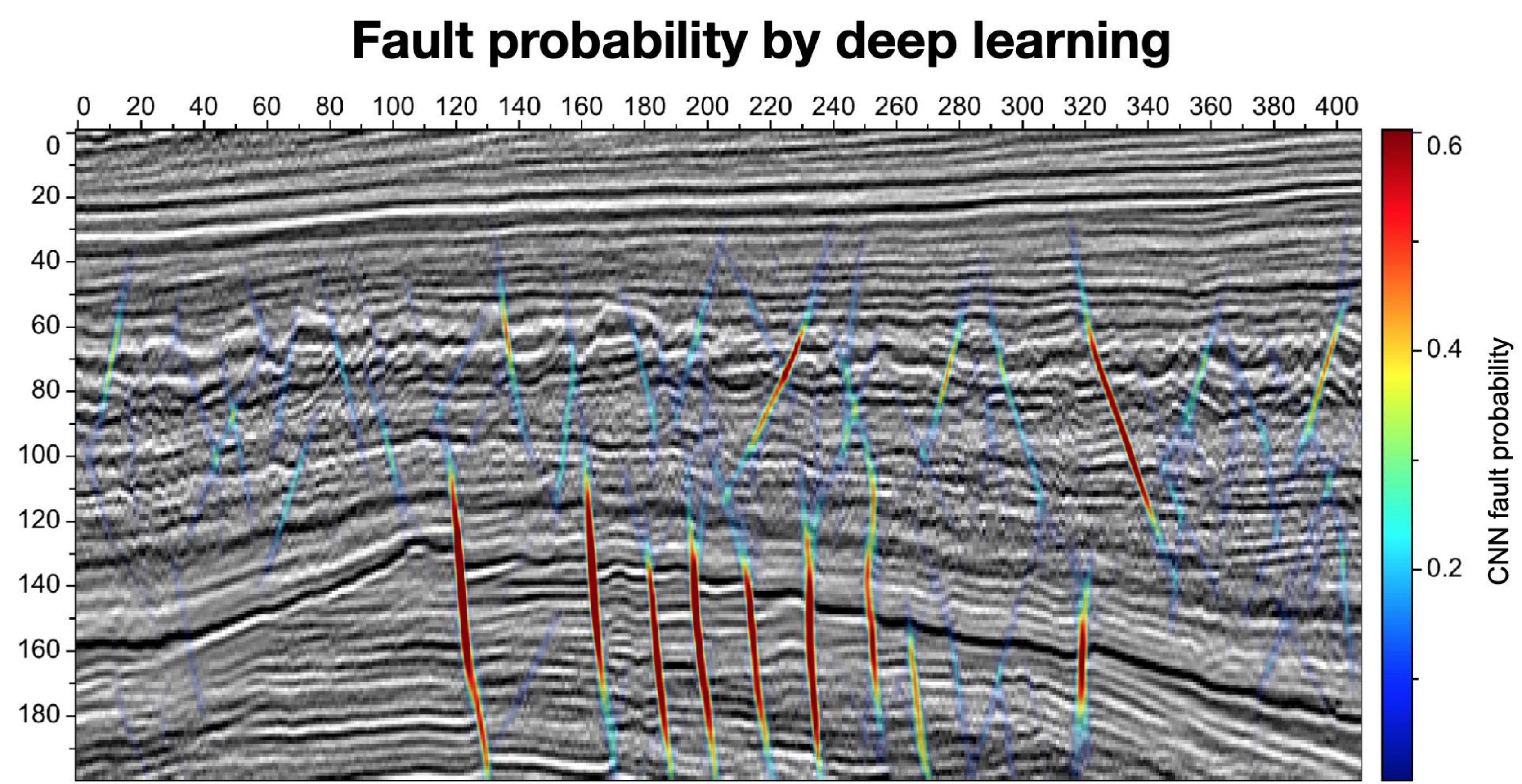
(Dave Hale, 2009)

Simple fault classification

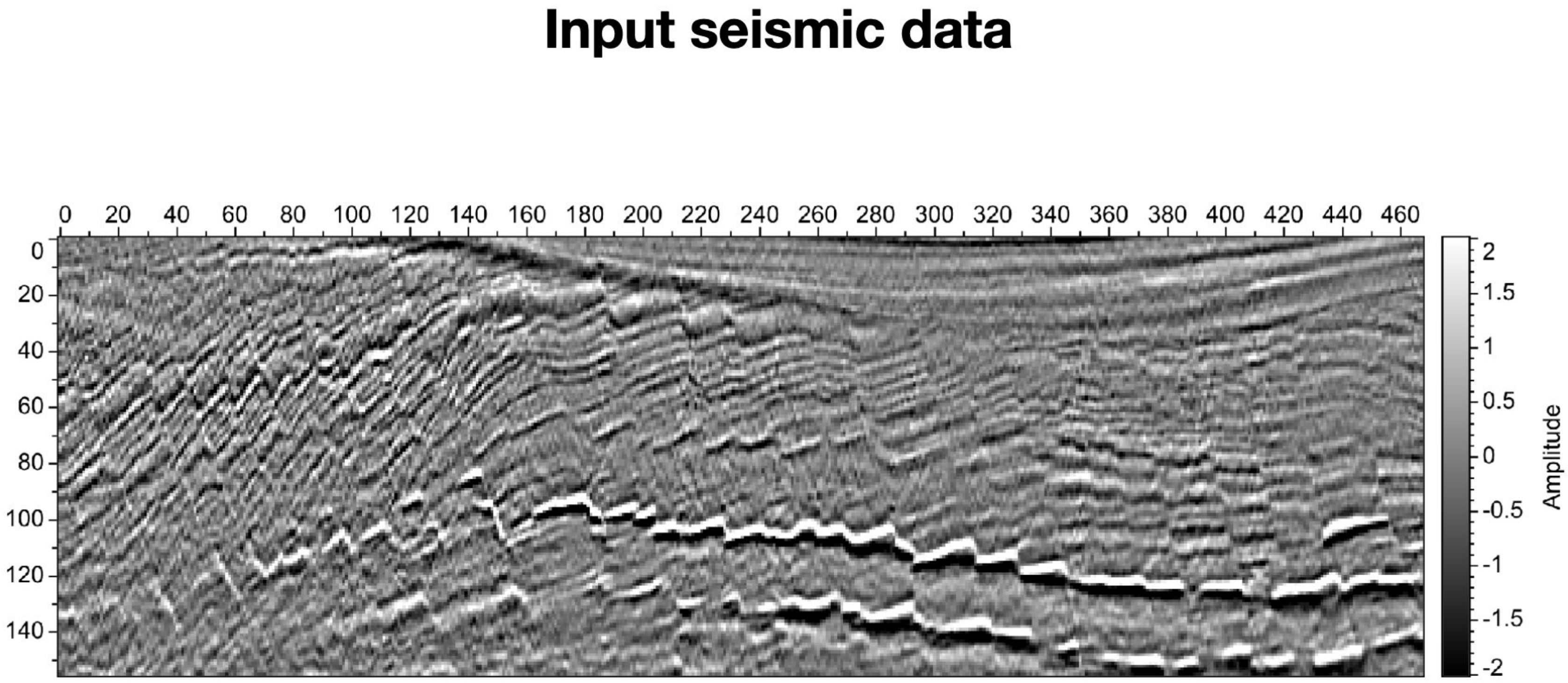


(Dave Hale , 2013; Dave Hale and Xinming Wu, 2016)

Simple fault classification

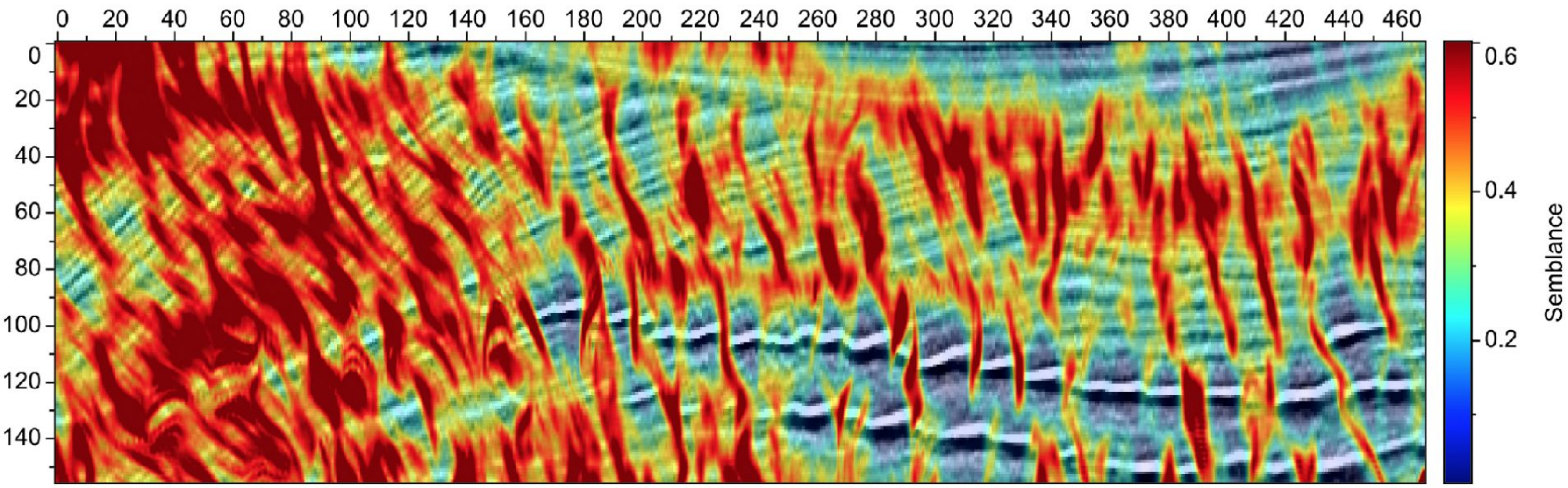


Simple fault classification



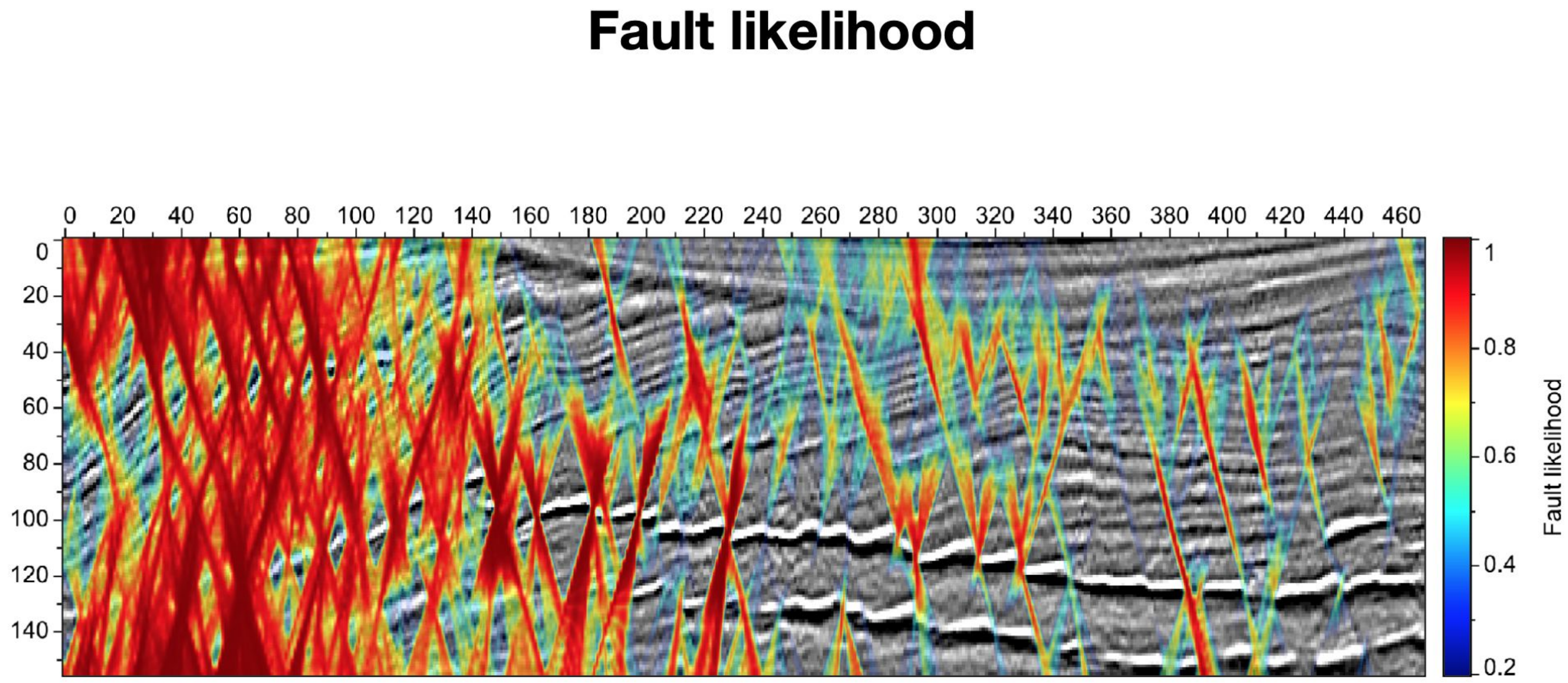
Simple fault classification

Semblance



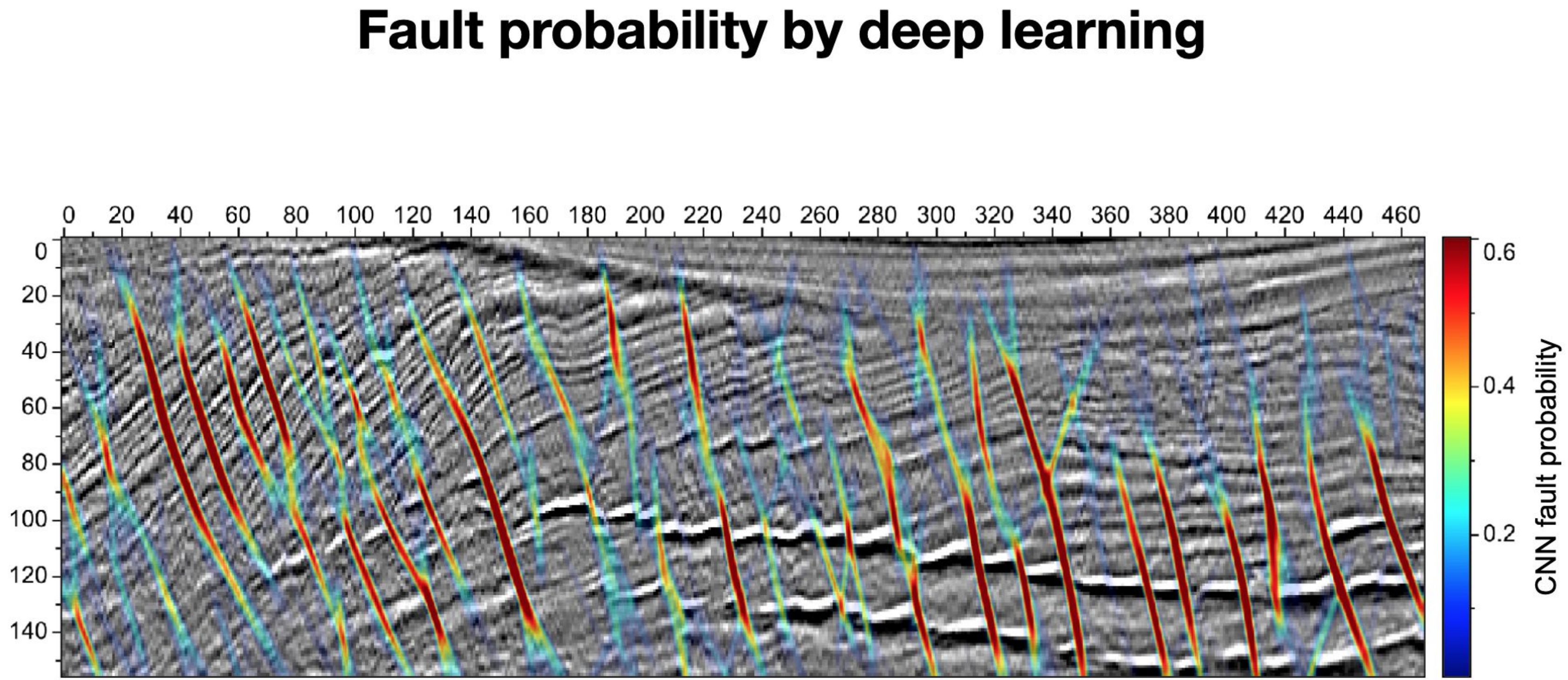
(Dave Hale , 2009)

Simple fault classification



(Dave Hale , 2013; Dave Hale and Xinming Wu, 2016)

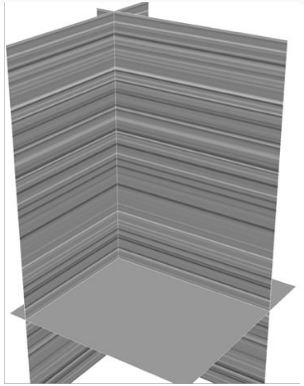
Simple fault classification



Outline

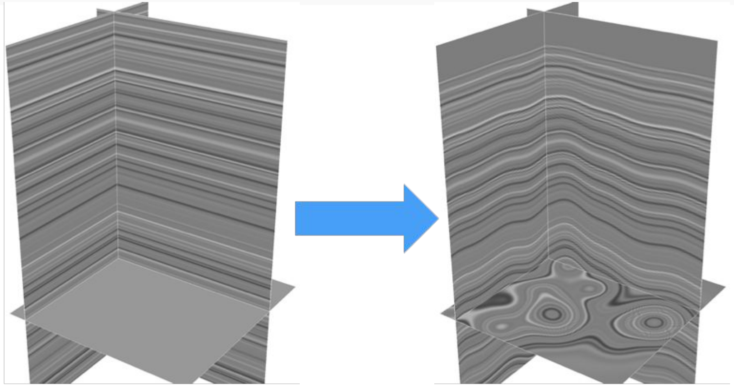
- Simple fault classification
- **Generating geophysical synthetic training data**
- From classification to segmentation:
  - Fault segmentation
  - Salt body segmentation
  - Channel segmentation
- Tracking geobody in a recurrent style
- Predicting relative geological time (RGT)

Workflow to generate 3D fake multi-fault sample



Initial reflectivity model

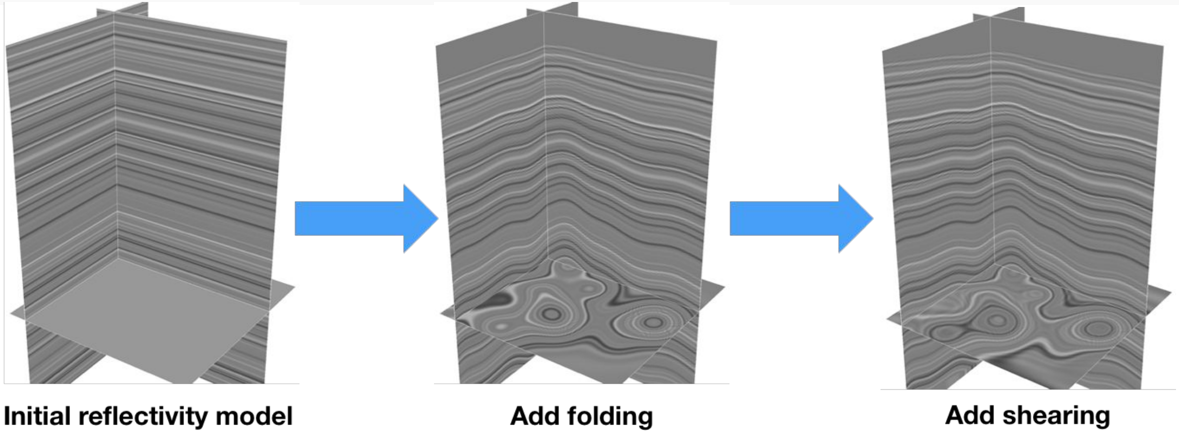
Workflow to generate 3D fake multi-fault sample



Initial reflectivity model **Add folding**

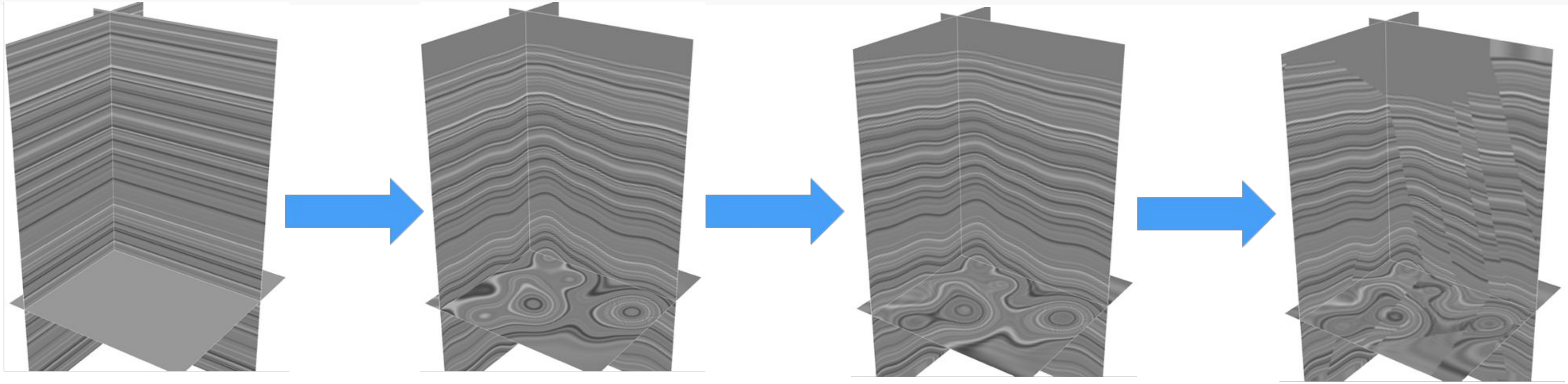
$$s_1(x, y, z) = a + \sum_{k=1}^{N} b_k \mathrm{e}^{\frac{(x-c_k)^2+(y-d_k)^2}{2\sigma_k^2}}$$

Workflow to generate 3D fake multi-fault sample



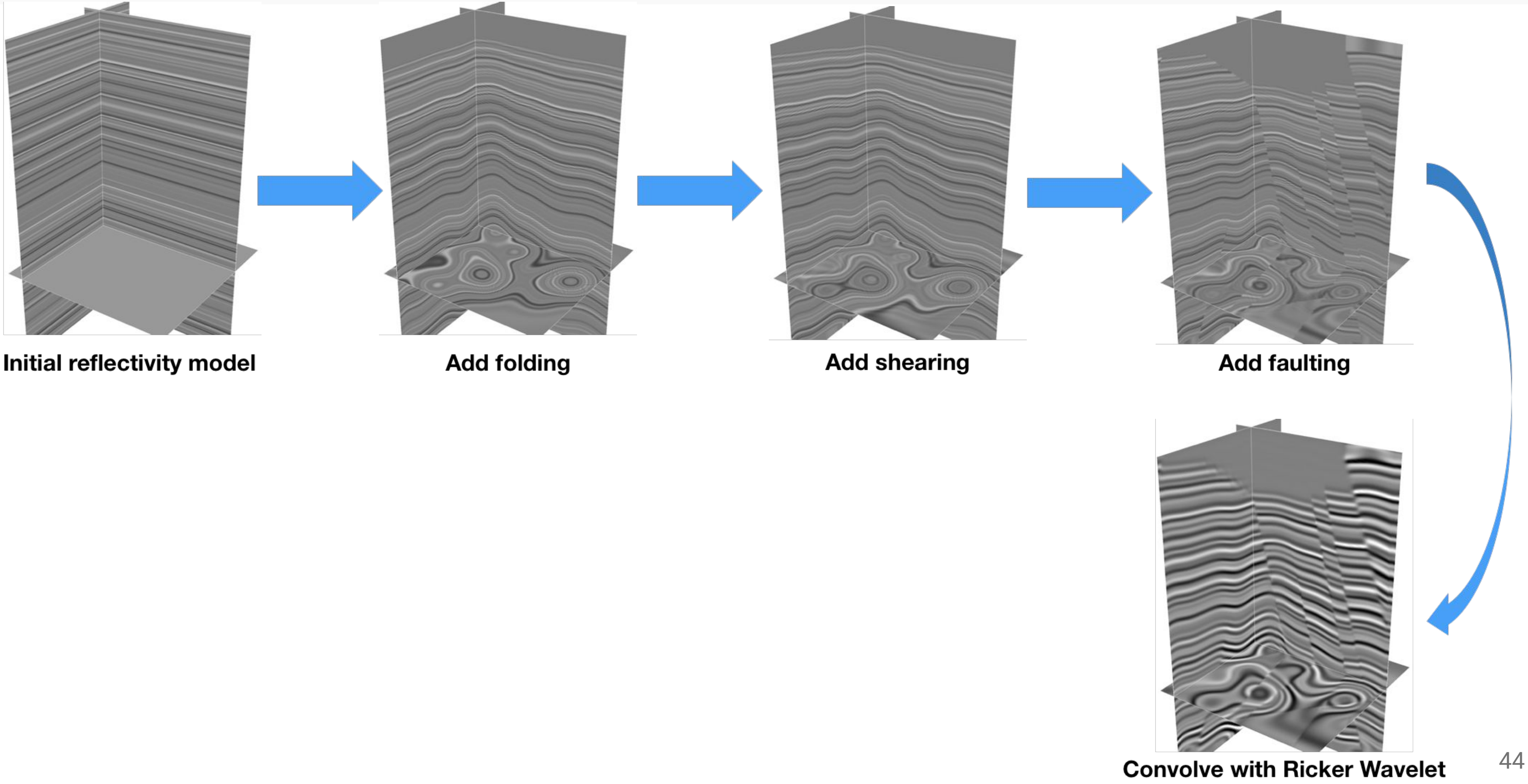
$$s_2(x, y, z) = e + fx + gy$$

Workflow to generate 3D fake multi-fault sample

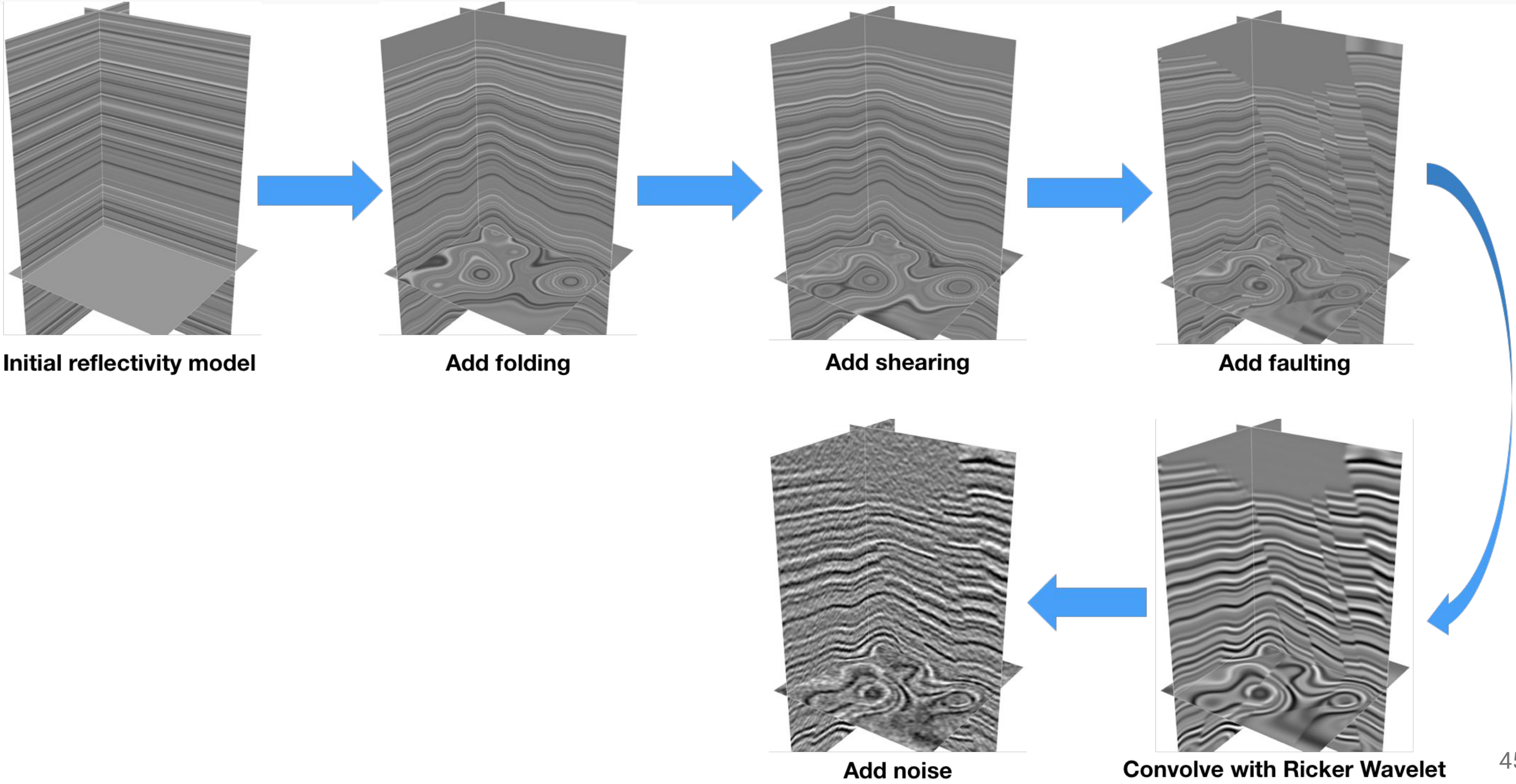


Initial reflectivity model | Add folding | Add shearing | Add faulting

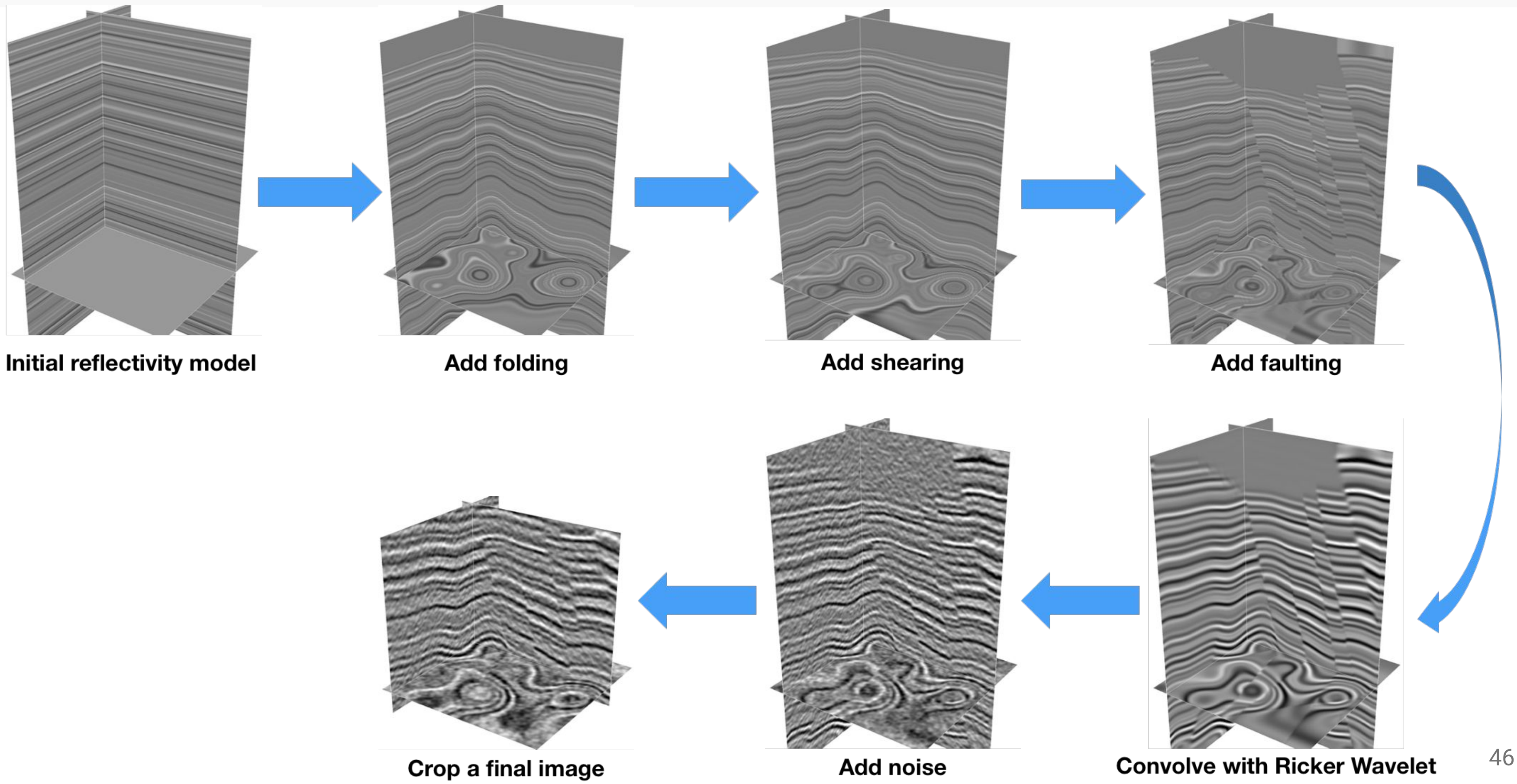
Workflow to generate 3D fake multi-fault sample



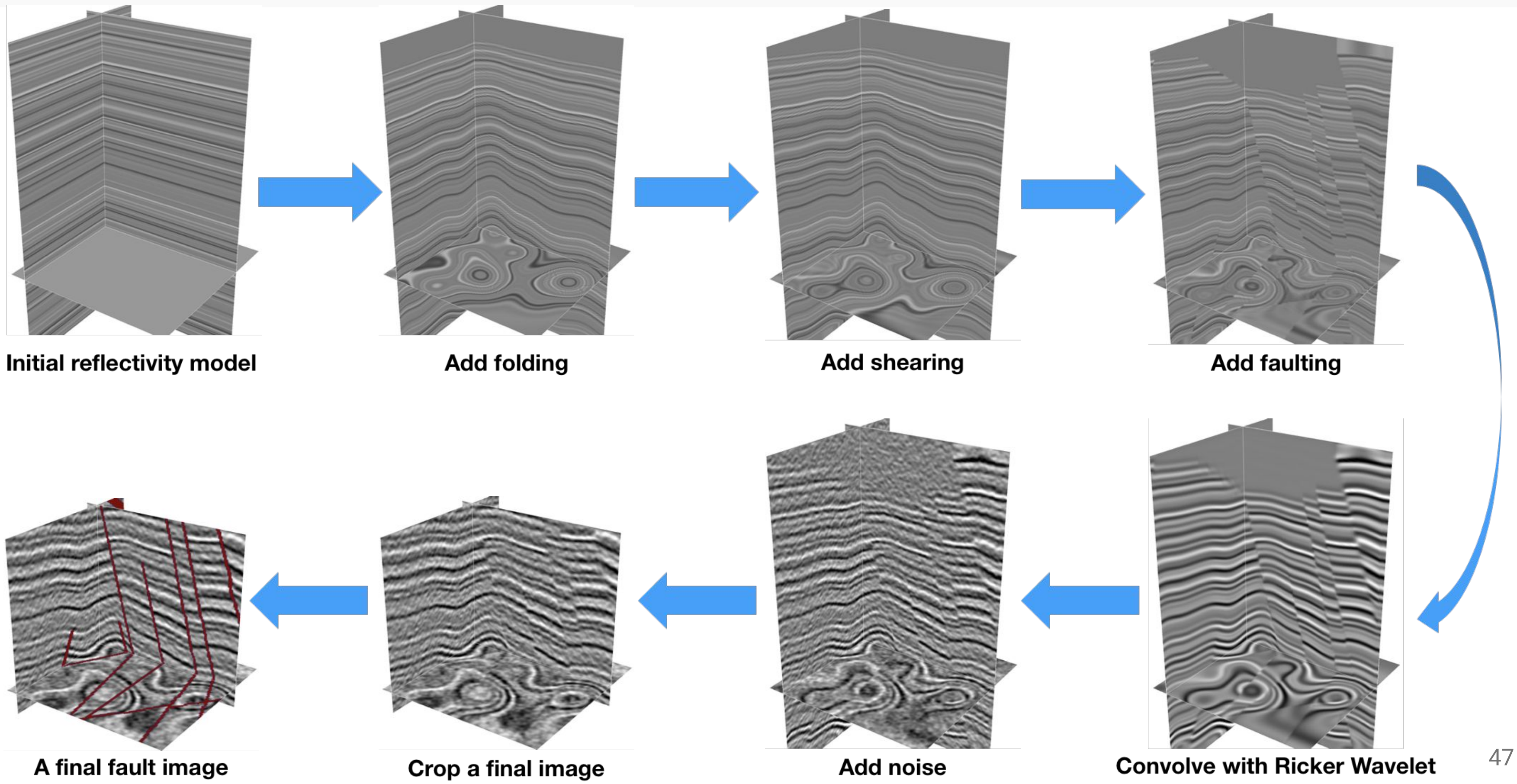
Workflow to generate 3D fake multi-fault sample



Workflow to generate 3D fake multi-fault sample

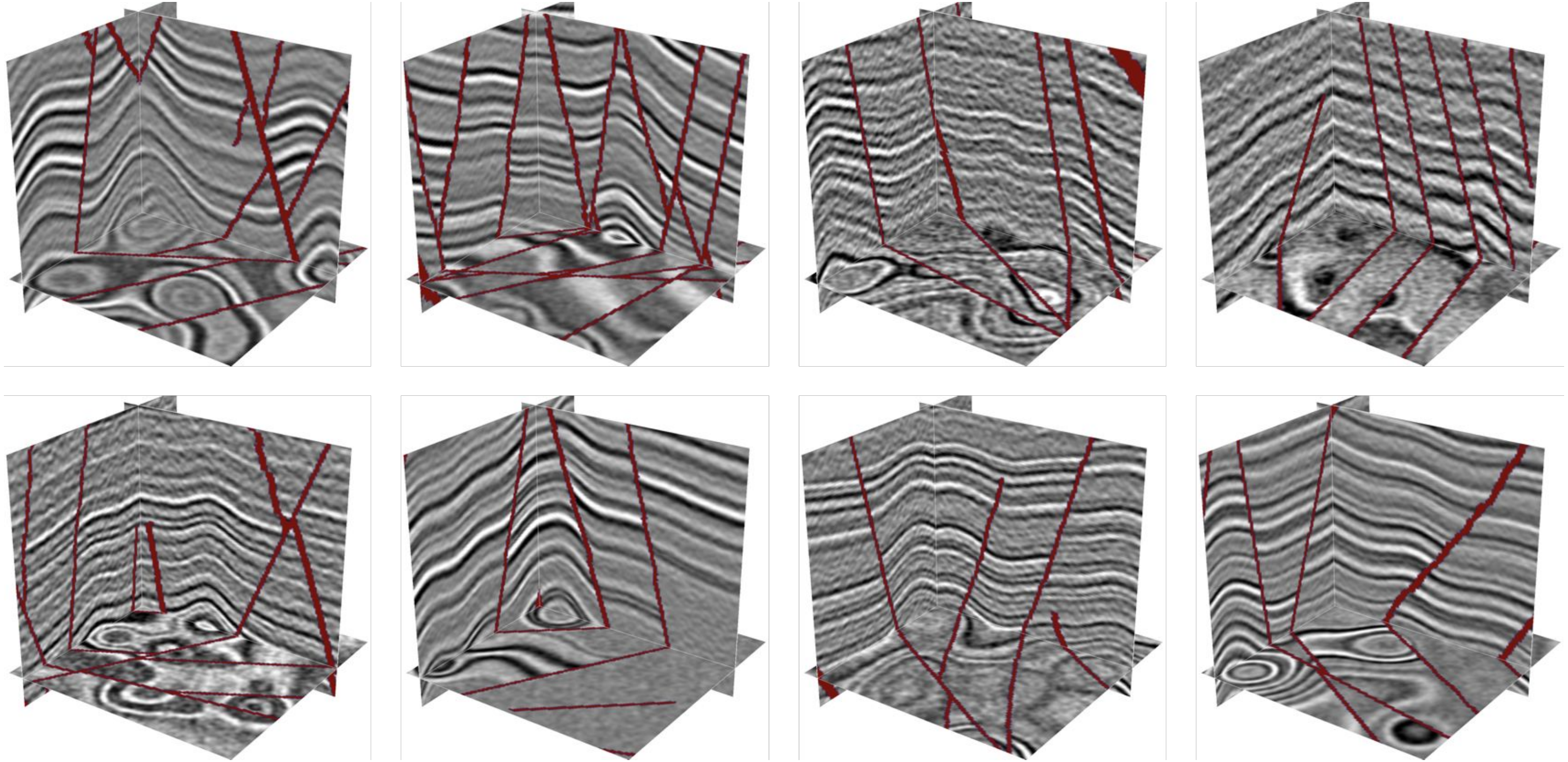


Workflow to generate 3D fake multi-fault sample

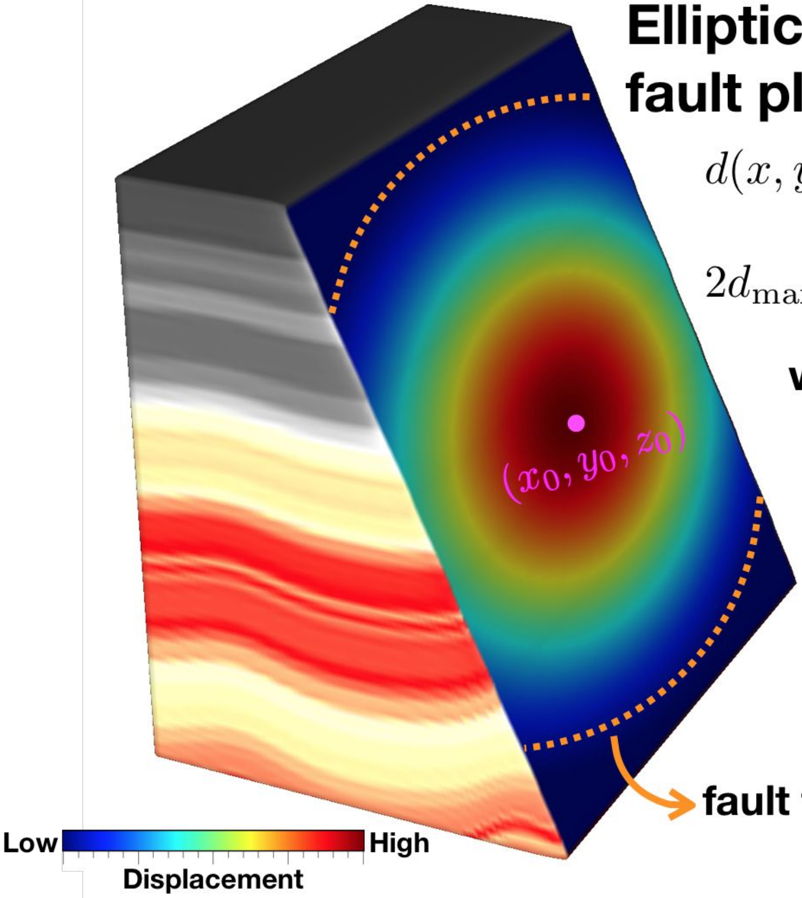


Training and validation datasets

200 synthetic training datasets + 20 synthetic validation datasets



Workflow to generate 3D fake multi-fault sample



Elliptic displacement field on the fault plane (*Walsh and Watterson, 1989*)

$$d(x, y; z = 0) =$$

$$2d_{\max}(1 - r(x, y))\sqrt{\frac{(1 + r(x, y))^2}{4} - r(x, y)^2}$$

where

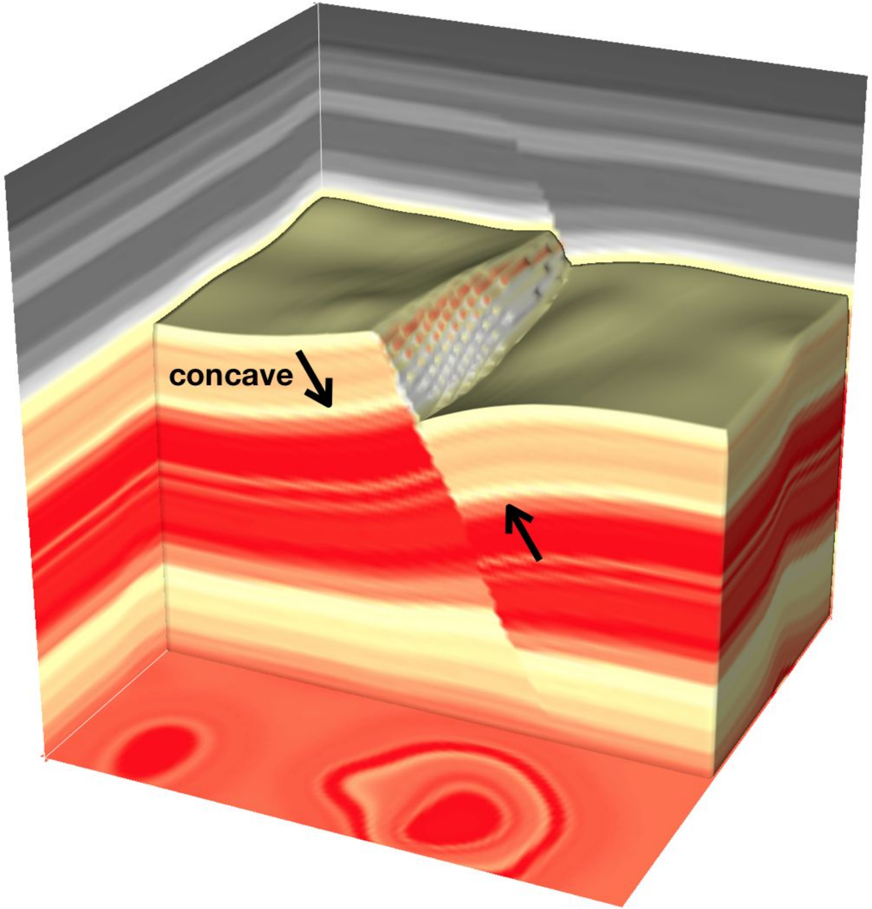
$$r(x, y) = \sqrt{(\frac{x - x_0}{l_x})^2 + (\frac{y - y_0}{l_y})^2}$$

$$(lx > l_y)$$

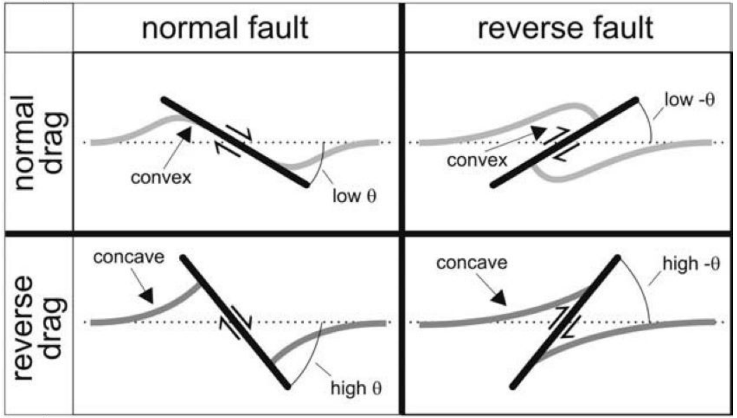
$d_{\max}$ **: the maximum displacement at the center point** (x_0, y_0, z_0)

fault tip-line: $r(x, y) = 1$

Workflow to generate 3D fake multi-fault sample

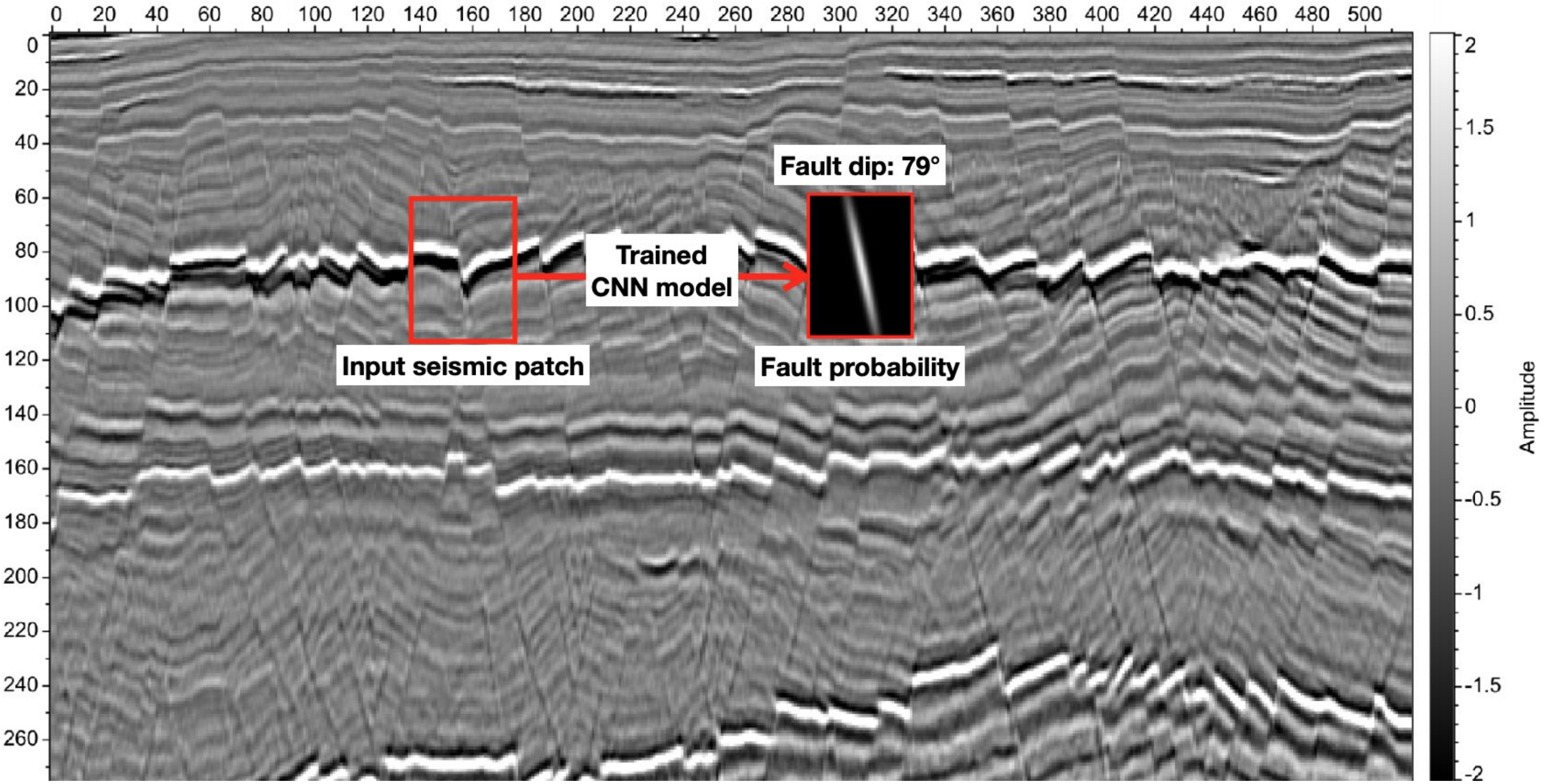


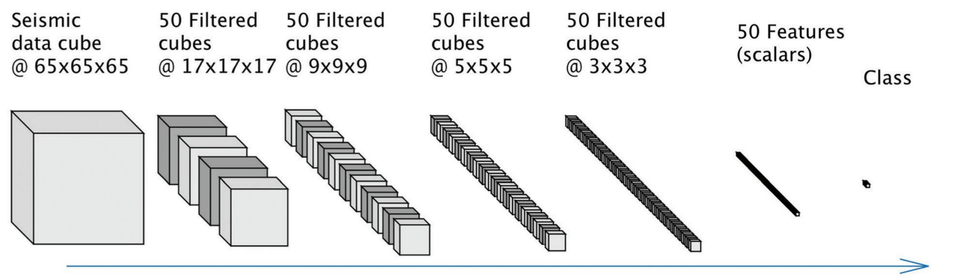
reverse drag by faulting

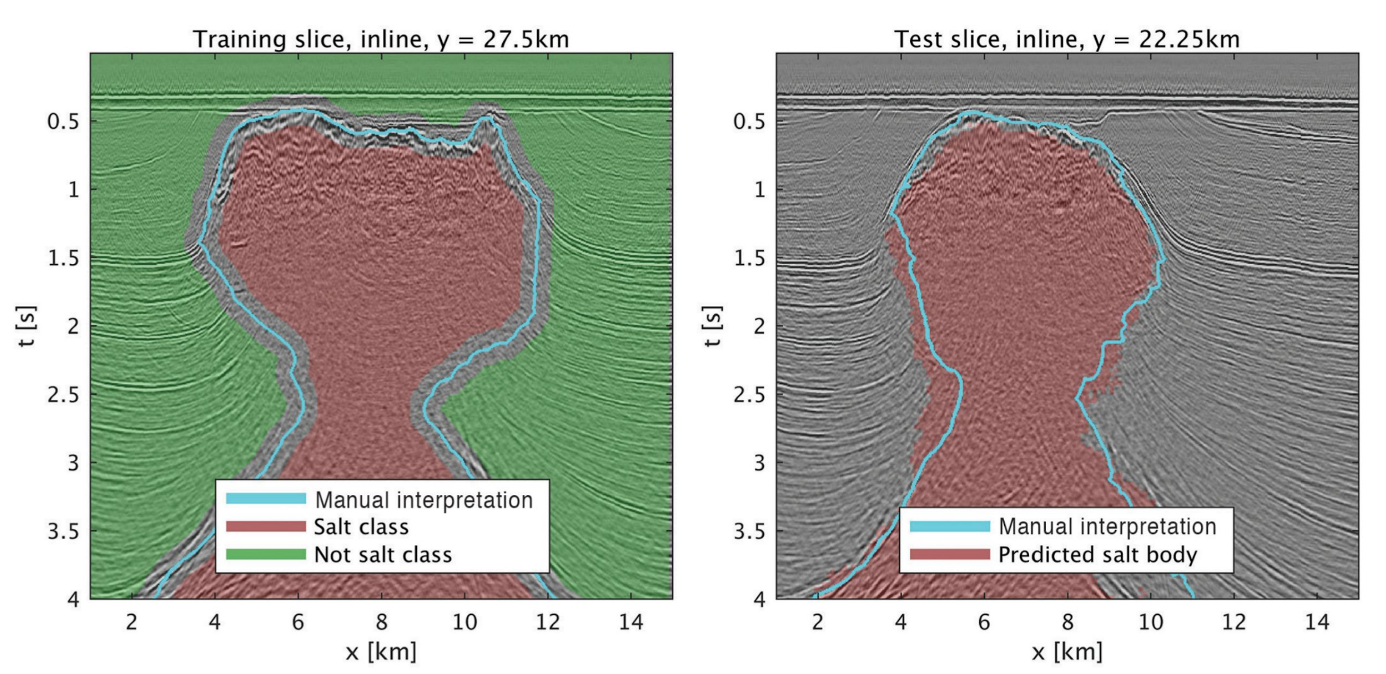


(Grasemann et al., 2005)

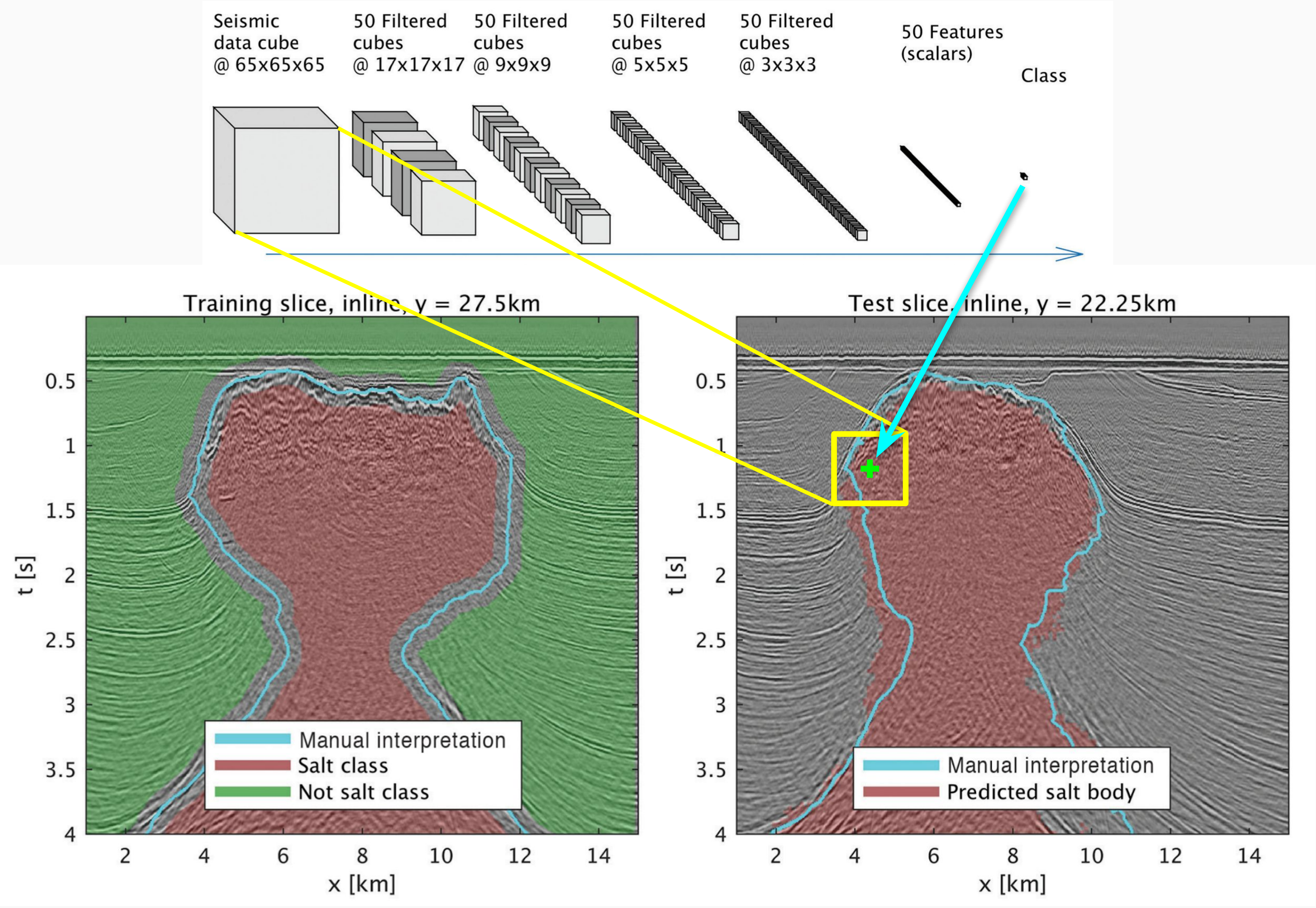
Simple fault classification



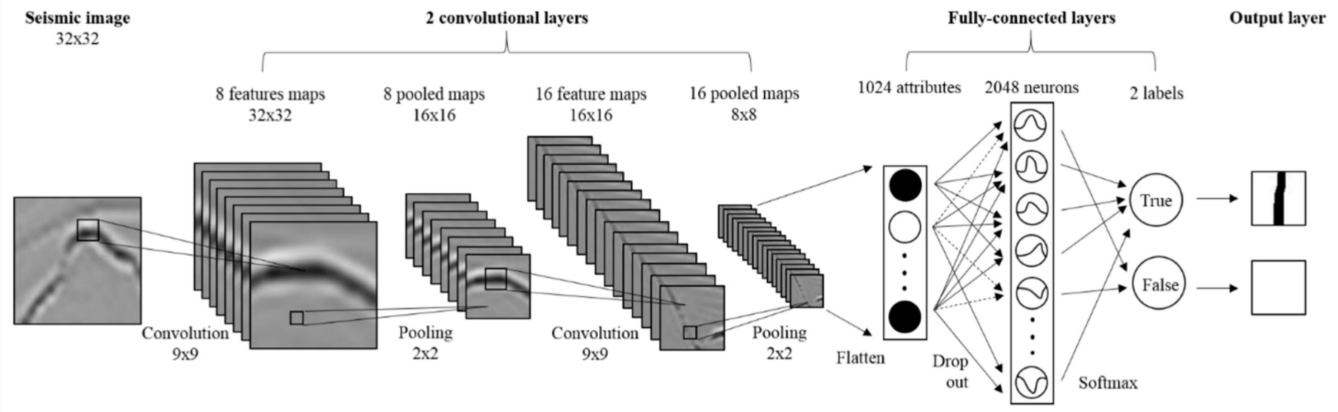


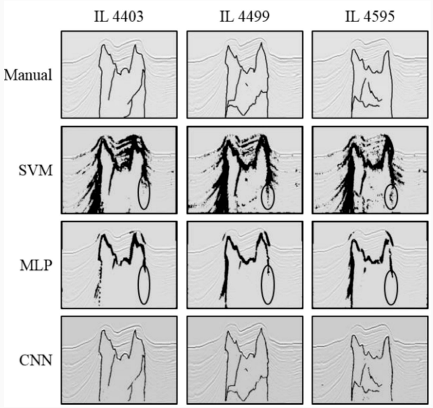


Waldeland et al., 2018, *Convolutional neural networks for automated seismic interpretation*, TLE



Waldeland et al., 2018, *Convolutional neural networks for automated seismic interpretation*, TLE

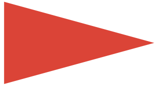

Seismic image
32x32
2 convolutional layers
Fully-connected layers
Output layer
8 features maps
32x32
8 pooled maps
16x16
16 feature maps
16x16
16 pooled maps
8x8
1024 attributes
2048 neurons
2 labels
Convolution
9x9
Pooling
2x2
Convolution
9x9
Pooling
2x2
Flatten
Drop
out
Softmax
True
False


IL 4403
IL 4499
IL 4595
Manual
SVM
MLP
CNN

Classification vs Segmentation

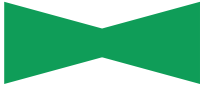
Deep learning is powerful.

But different problem setup could unlock more power!



Classification

- Difficult to handle the border of different classes;
- Sliding window could be computational intensive.



Segmentation

- Suitable setup for geobody detection problems;
- Less subdomains than sliding windows

Outline

- Simple fault classification
- Generating geophysical synthetic training data
- **From classification to segmentation:**
  - **Fault segmentation**
  - **Salt body segmentation**
  - **Channel segmentation**
- Tracking geobody in a recurrent style
- Predicting relative geological time (RGT)

Segmentation problems

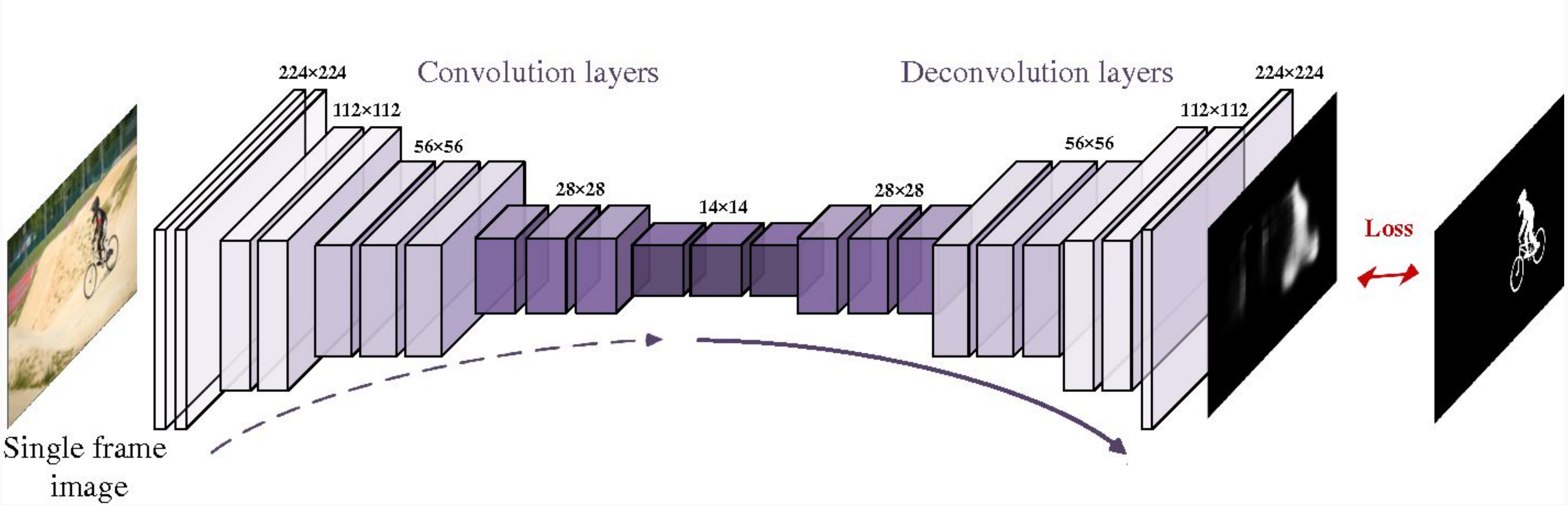
Input image

Semantic segmentation

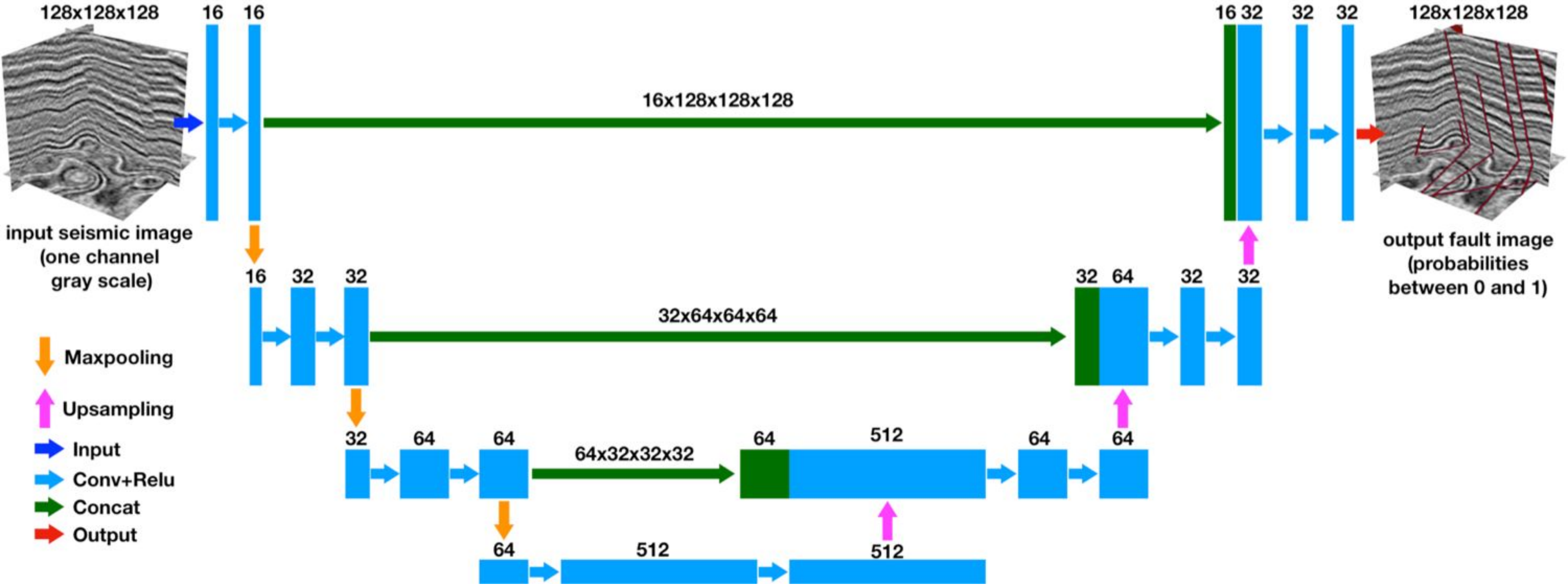
- Autonomous driving
- Satellite surveillance
- Video processing

- Medical image analysis
- **Geophysical data interpretation**

Earliest network architecture for segmentation problems: Fully convolutional networks (FCN).



Simplified U-net fault segmentation



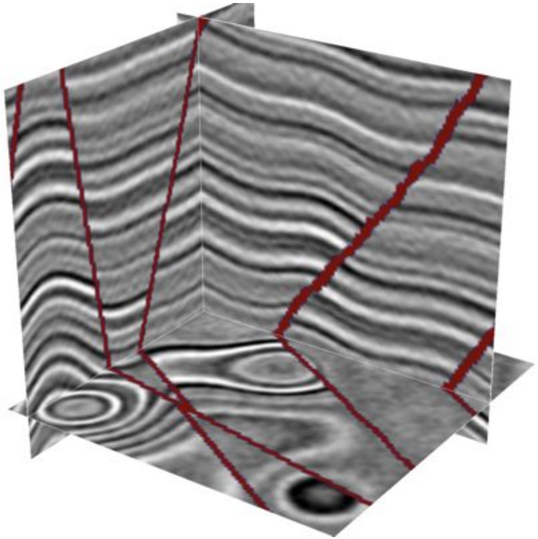
The original U-Net is more complicated than necessary for fault segmentation.
We simplified the U-Net by reducing both the layers and number of features at each layer.

Balanced cross-entropy loss

A fault image is highly imbalanced between zeros (non-fault) and ones (fault)

Conventional cross-entropy loss:

$$\mathbb{L} = -\sum_{i=0}^{N} y_i \log p_i - \sum_{i=0}^{N} (1 - y_i) \log (1 - p_i)$$

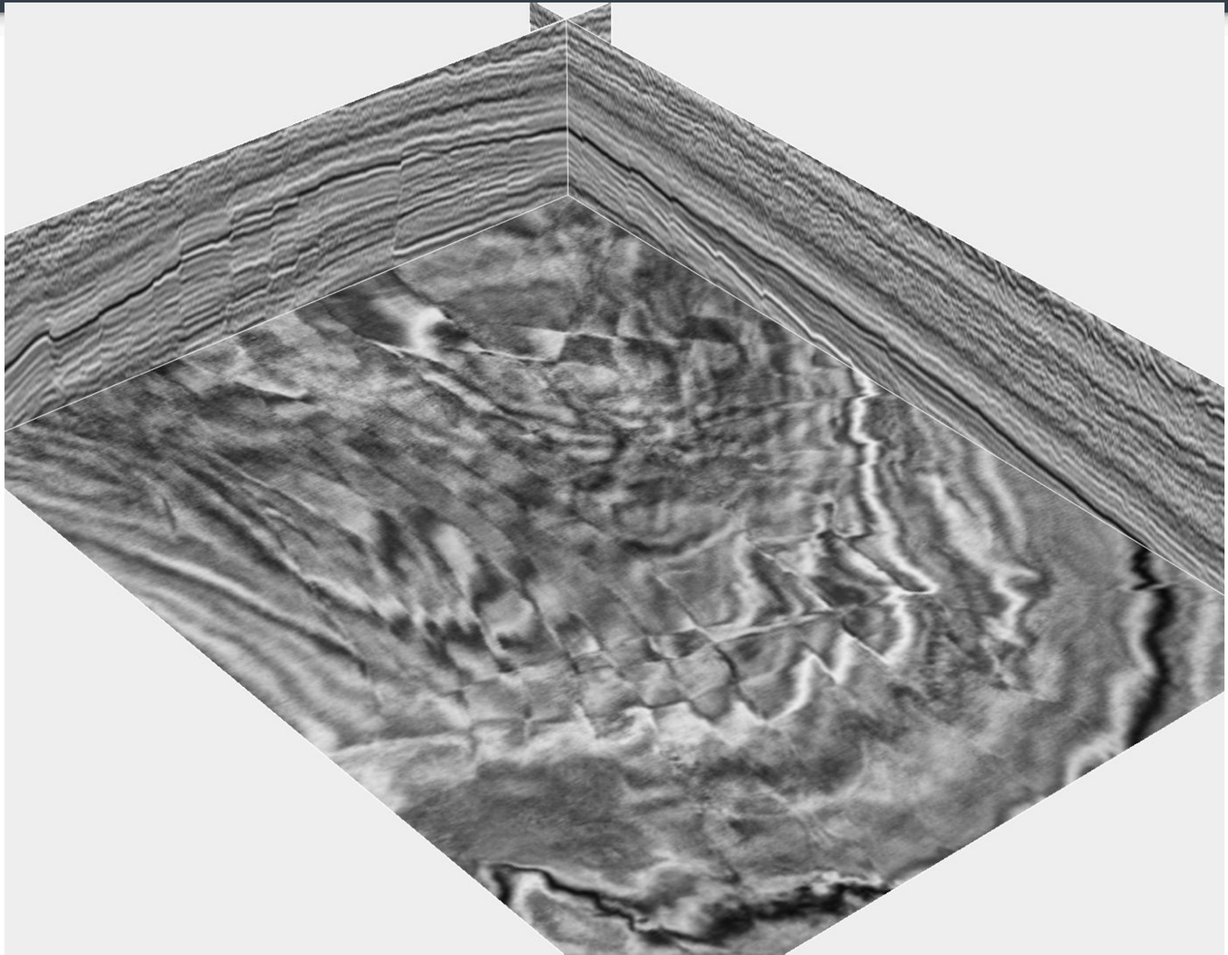


Balanced cross-entropy loss:

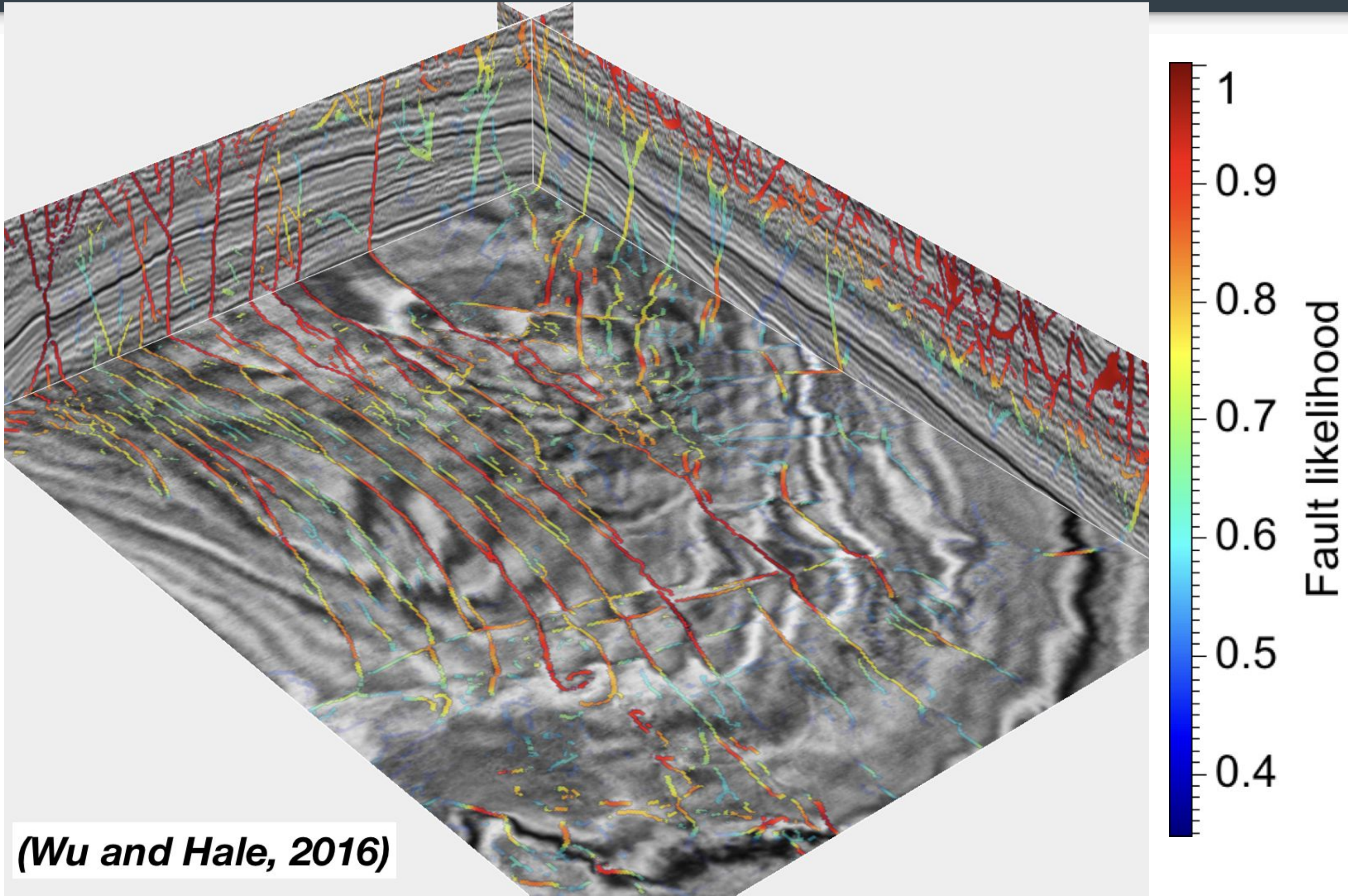
$$\mathbb{L} = -(1 - \beta) \sum_{i=0}^{N} y_i \log p_i - \beta \sum_{i=0}^{N} (1 - y_i) \log (1 - p_i)$$

where: $\beta = \frac{\sum_{i=0}^{N} y_i}{N}$

Field examples: Subset of F3

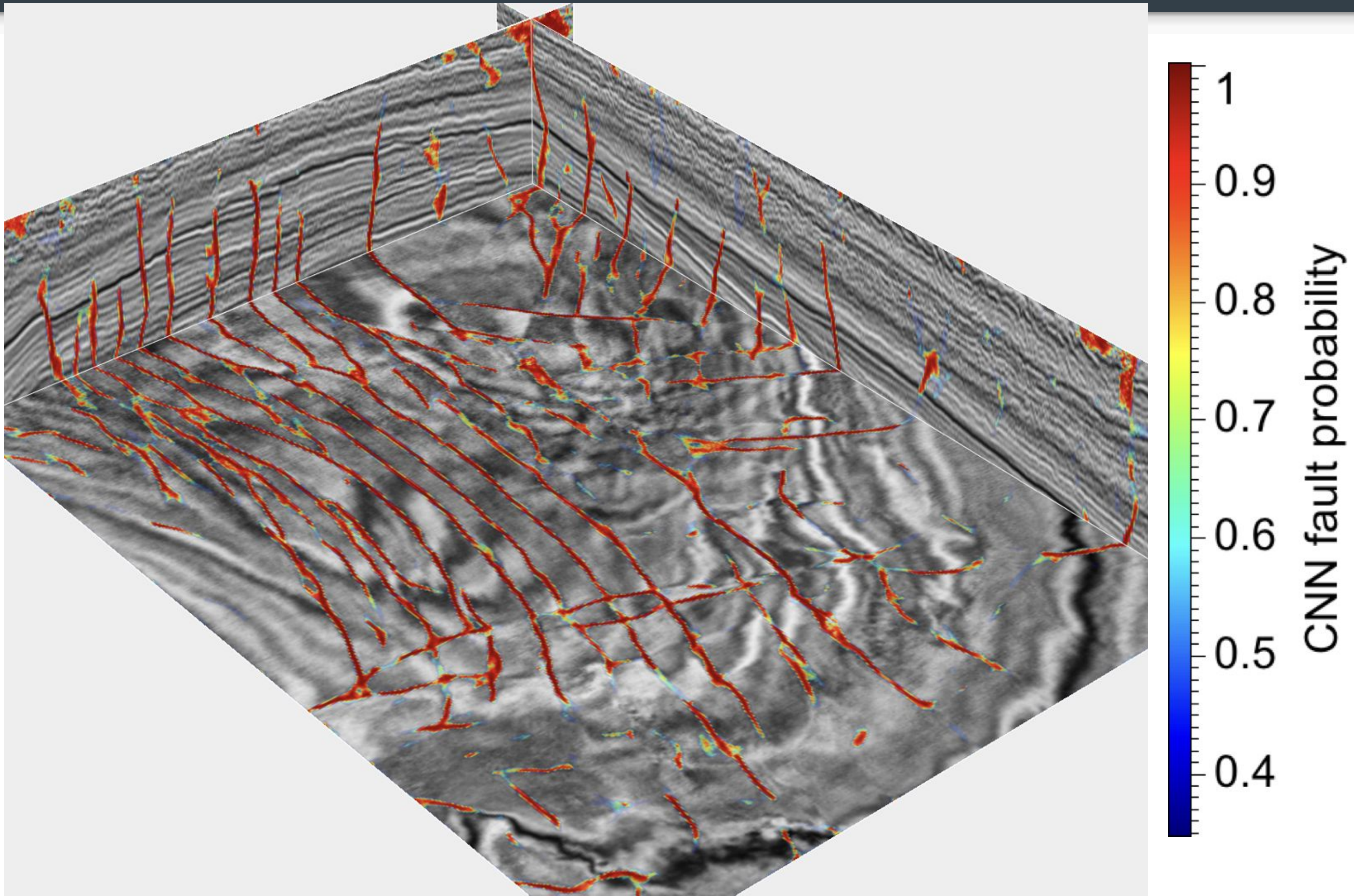


Thinned fault likelihood (previous method)

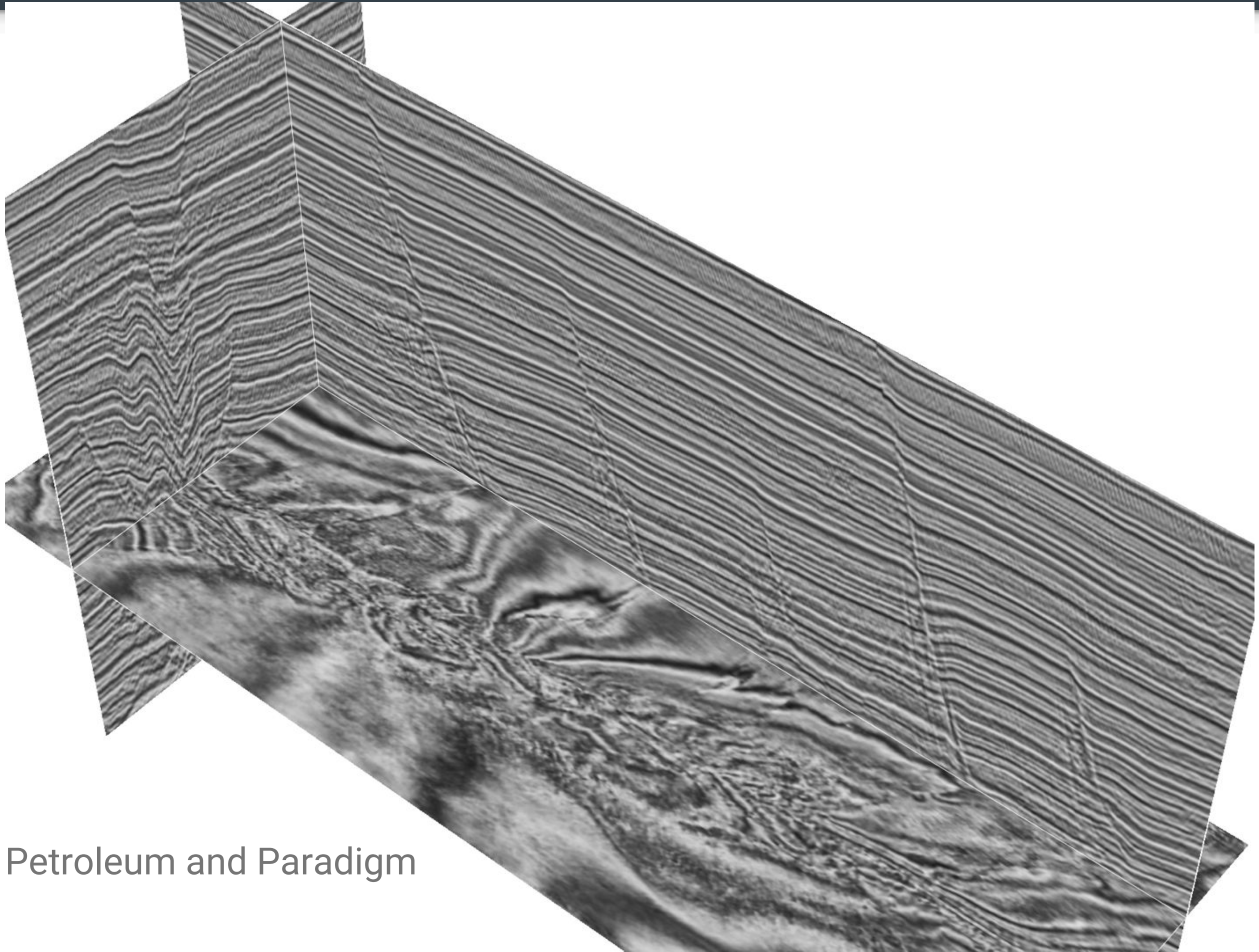


(Wu and Hale, 2016)

CNN fault probability

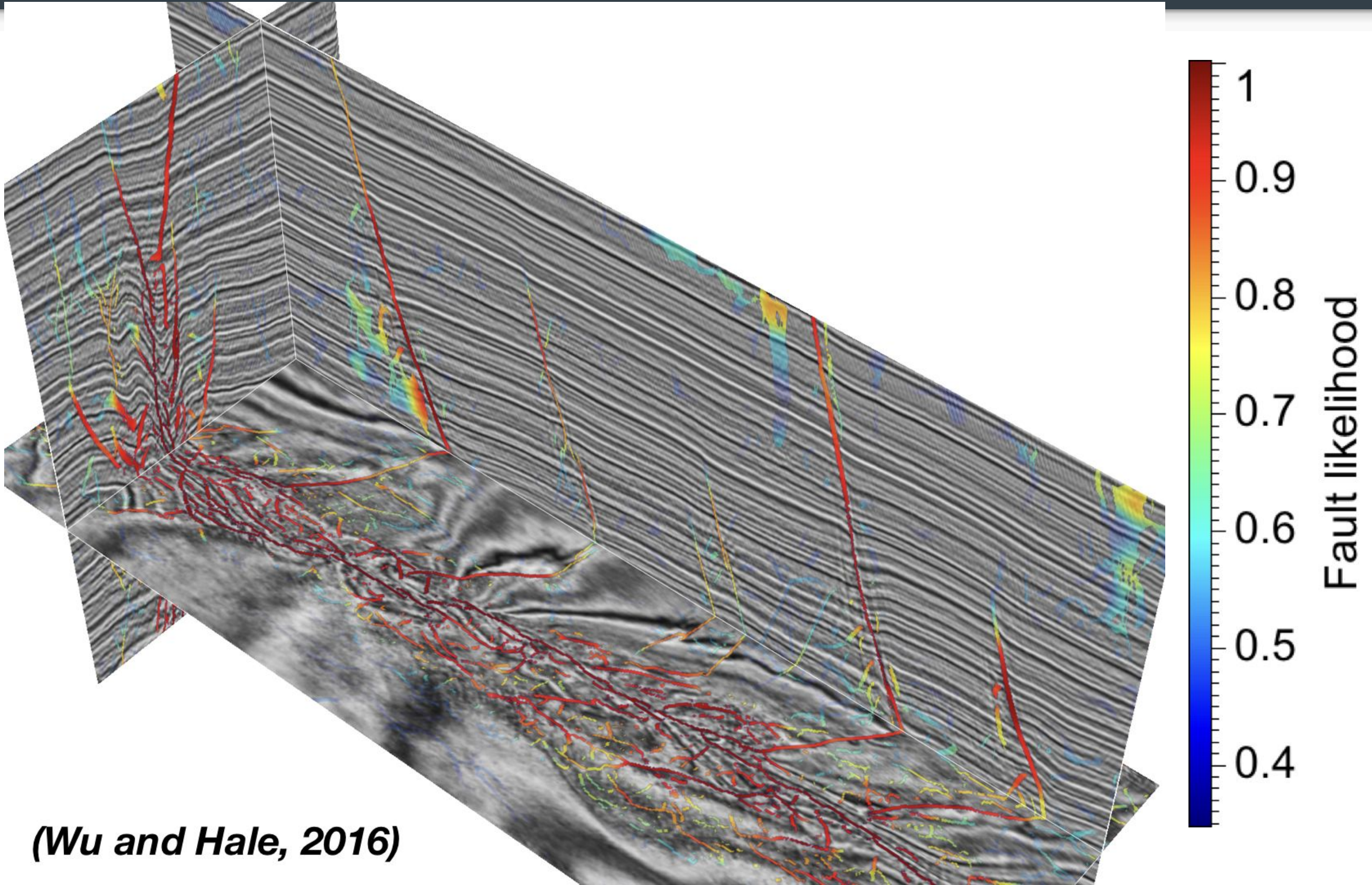


Field example 2

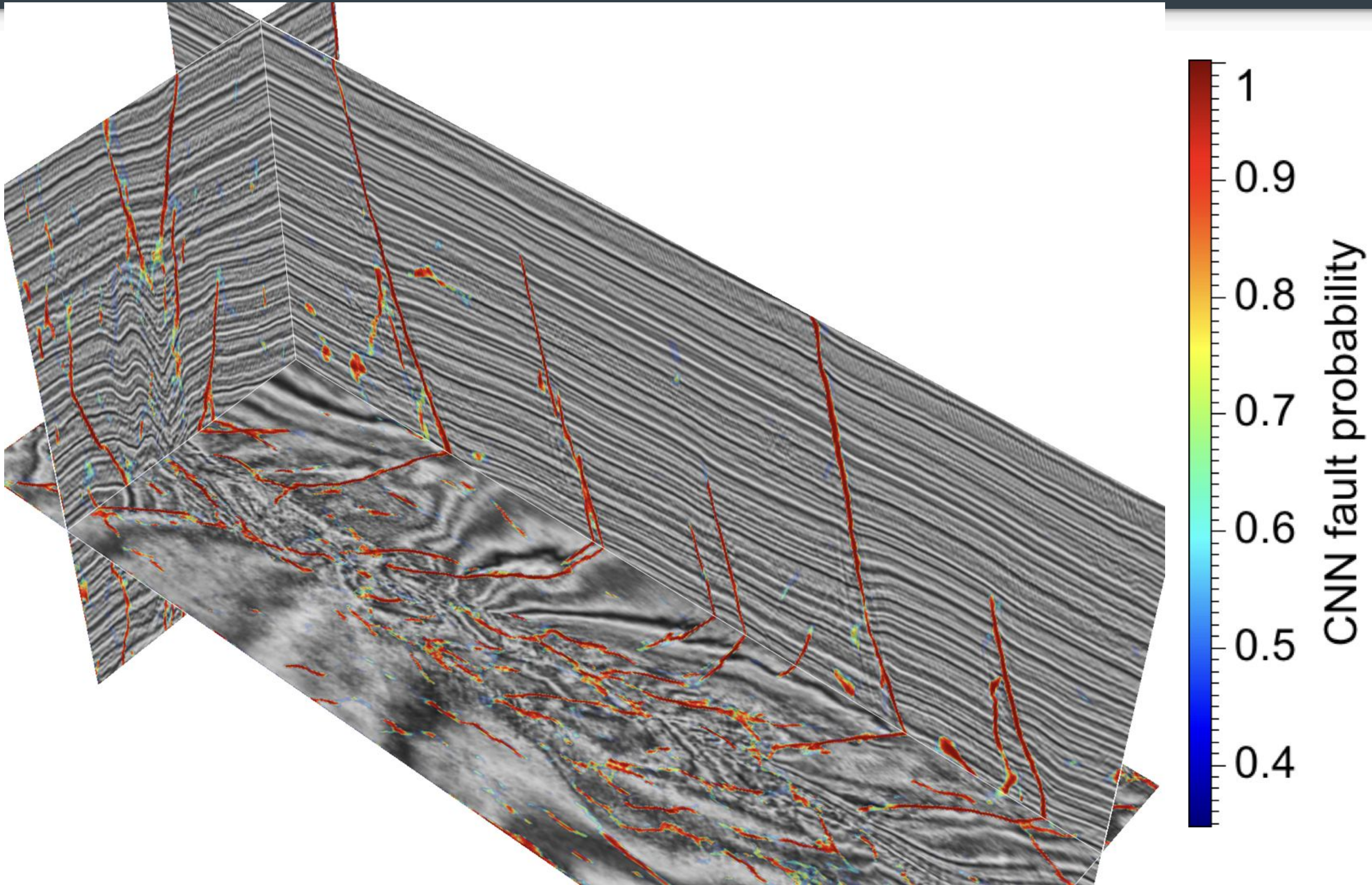


Provided by Clyde Petroleum and Paradigm

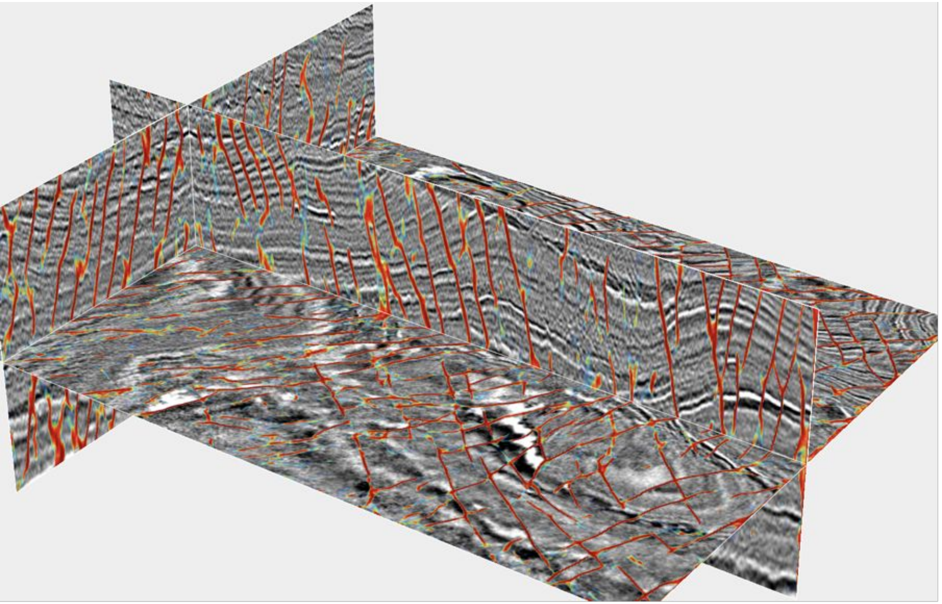
Thinned fault likelihood (conventional method)

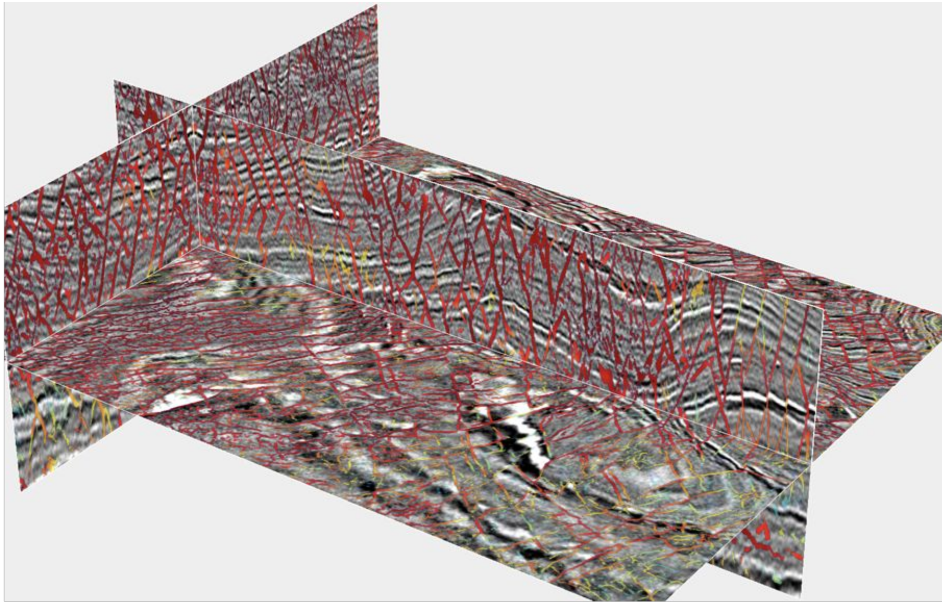


CNN fault probability



Field example 3: Costa Rica Margin

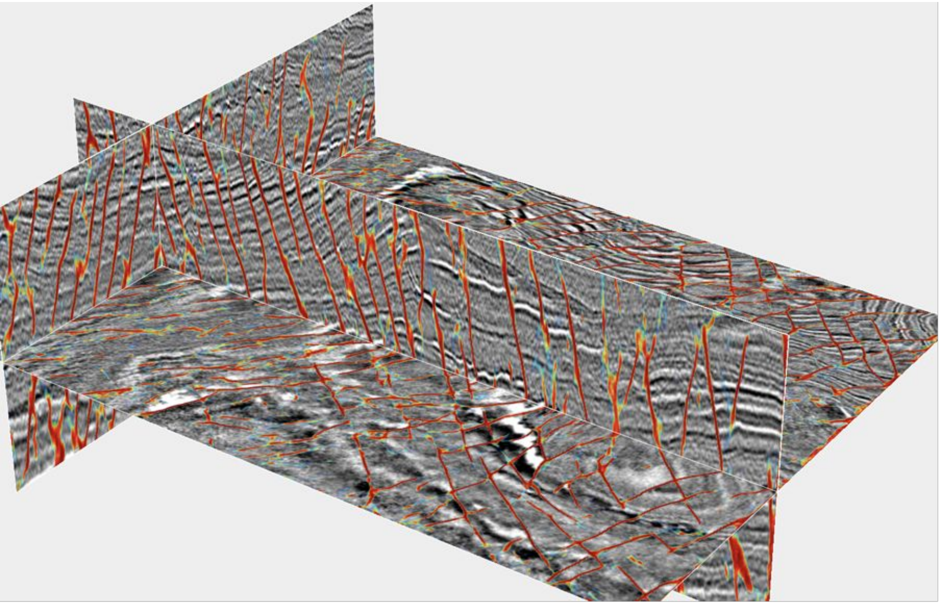




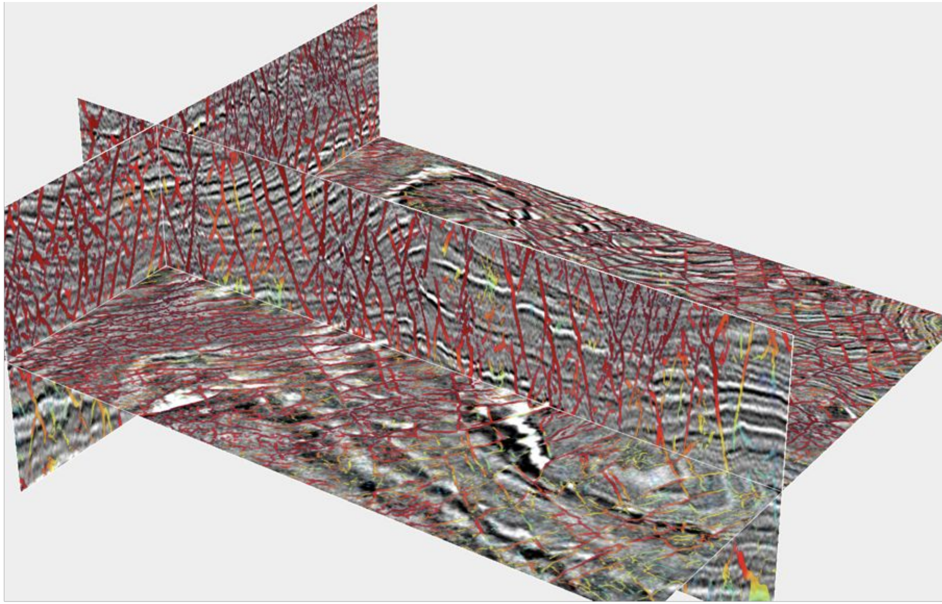
CNN fault probability

Thinned fault likelihood

Field example 3: Costa Rica Margin

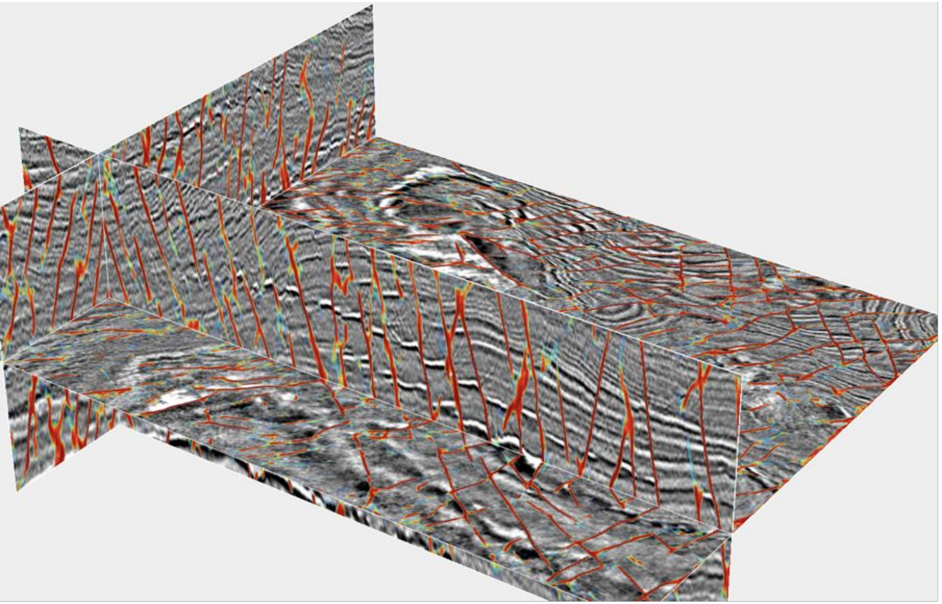


CNN fault probability

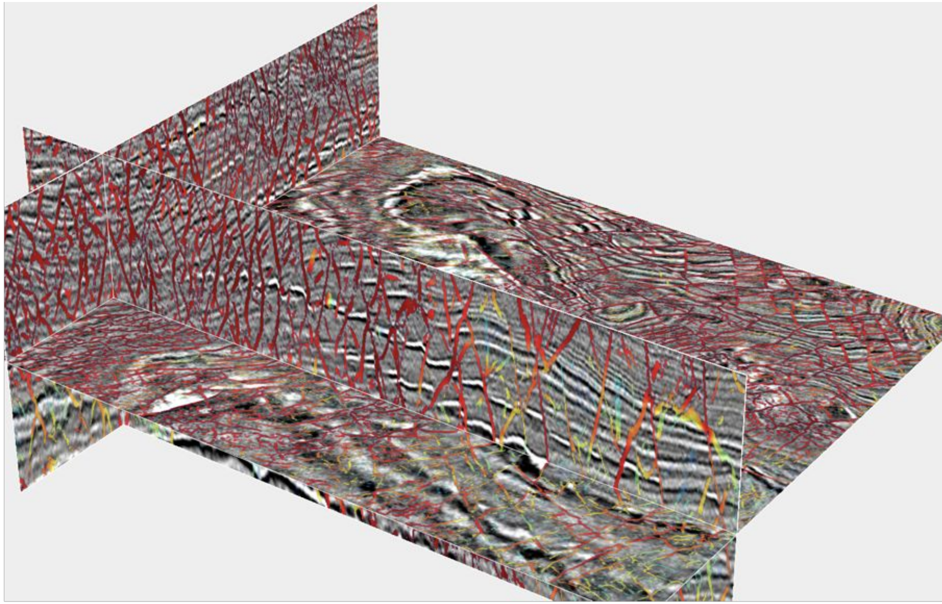


Thinned fault likelihood

Field example 3: Costa Rica Margin

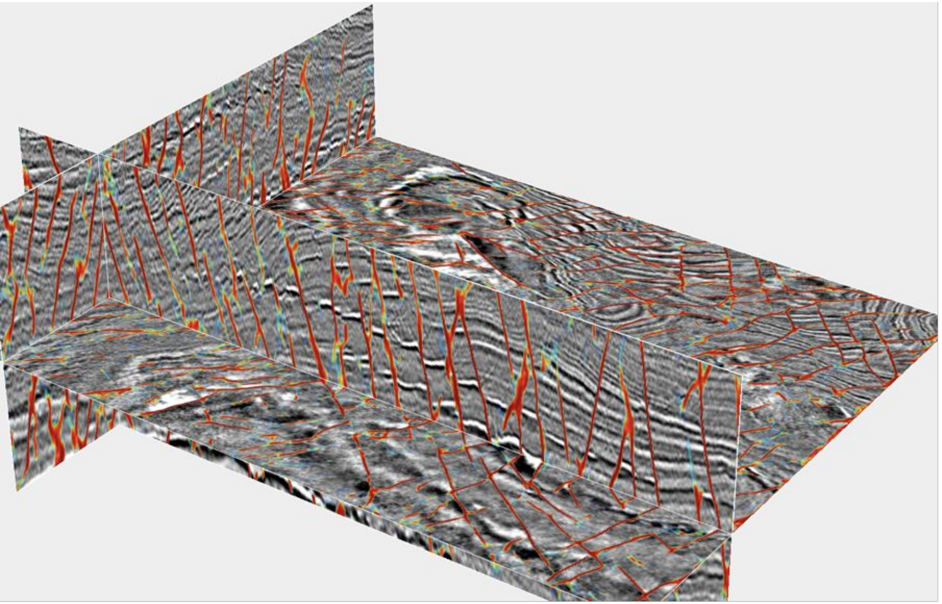


CNN fault probability

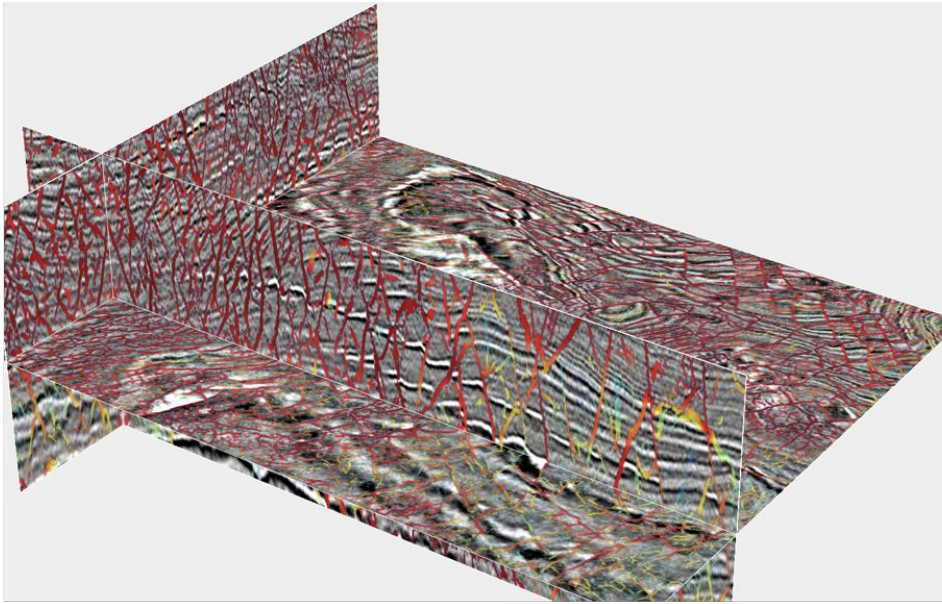


Thinned fault likelihood

Field example 3: Costa Rica Margin

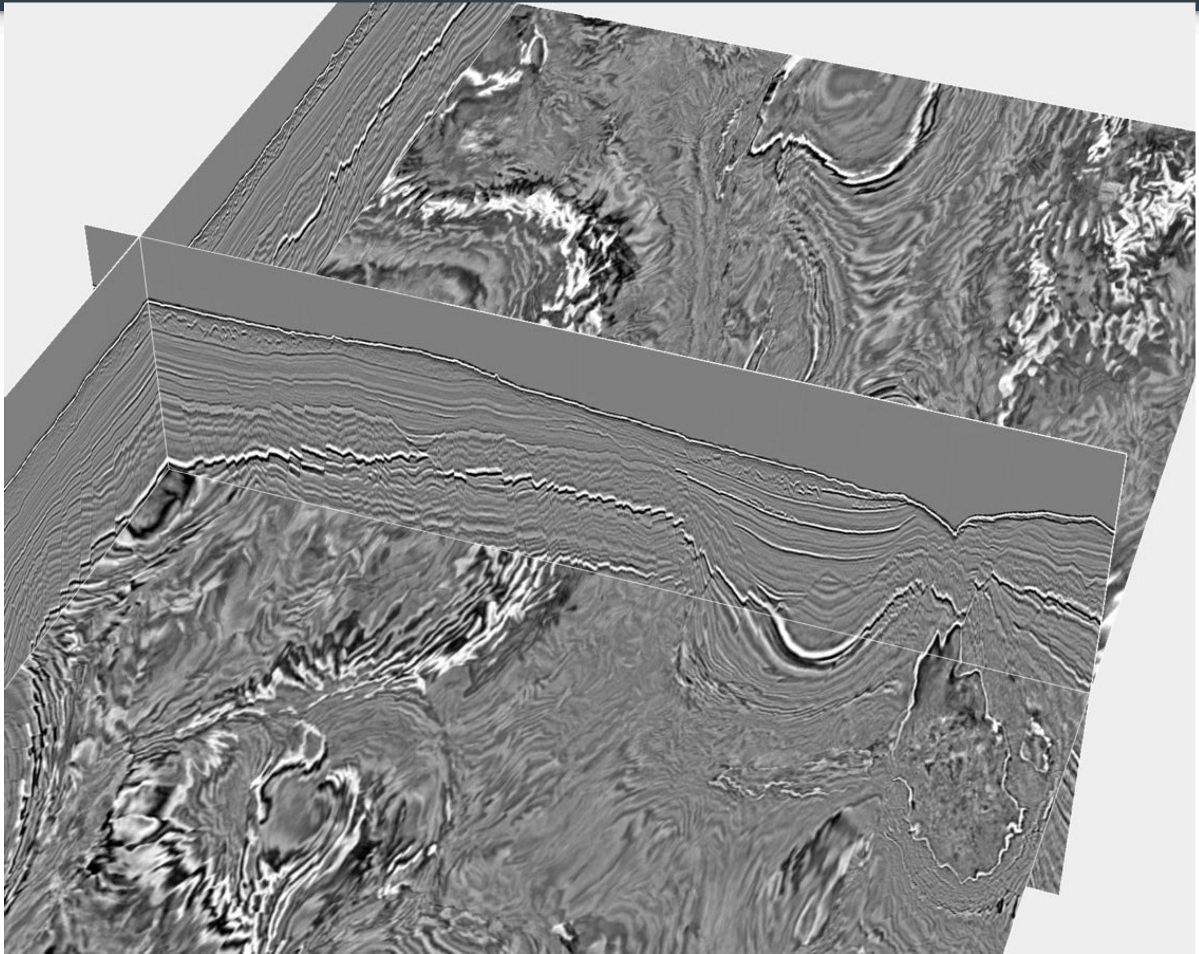


CNN fault probability



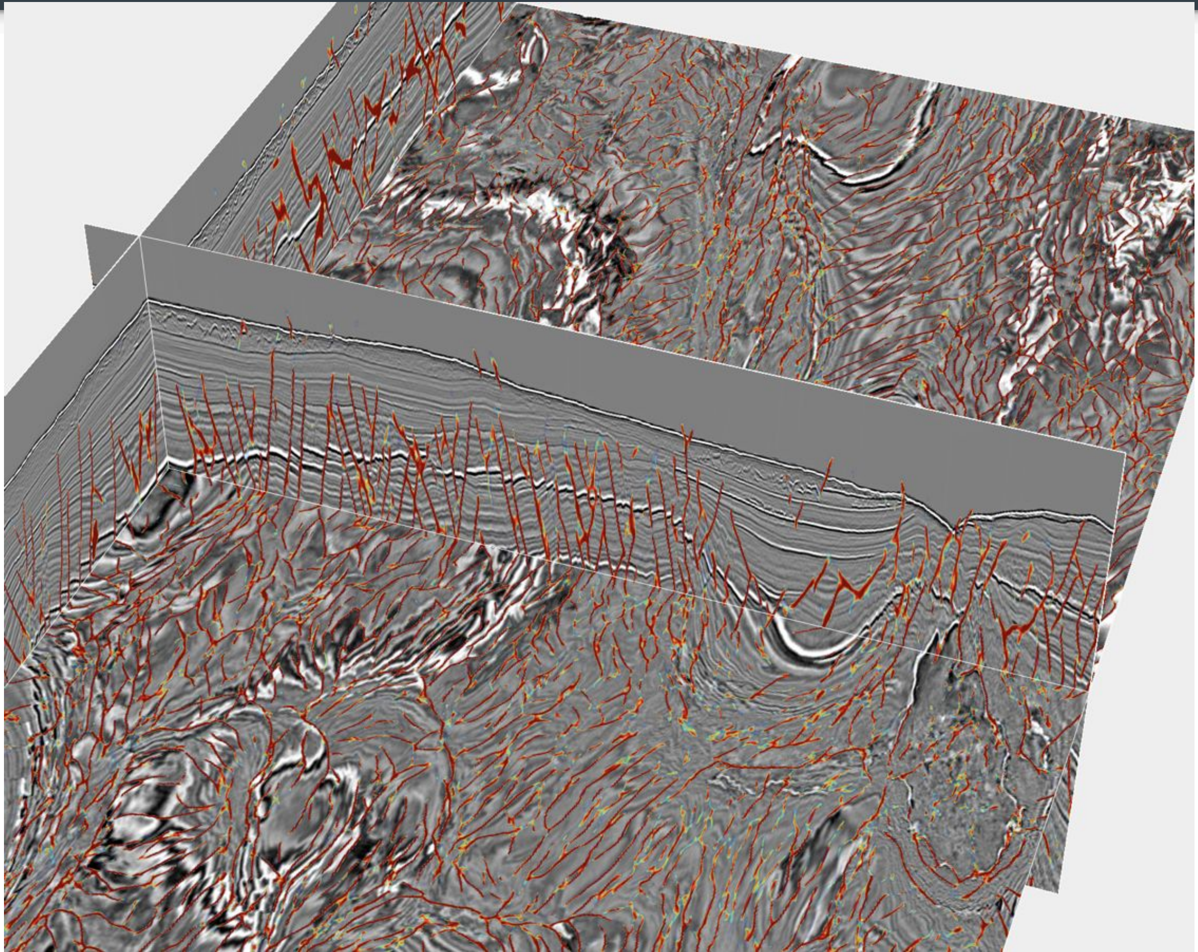
Thinned fault likelihood

Field example 4: Campos Basin, offshore Brazil



450 x 1950 x 1200 samples

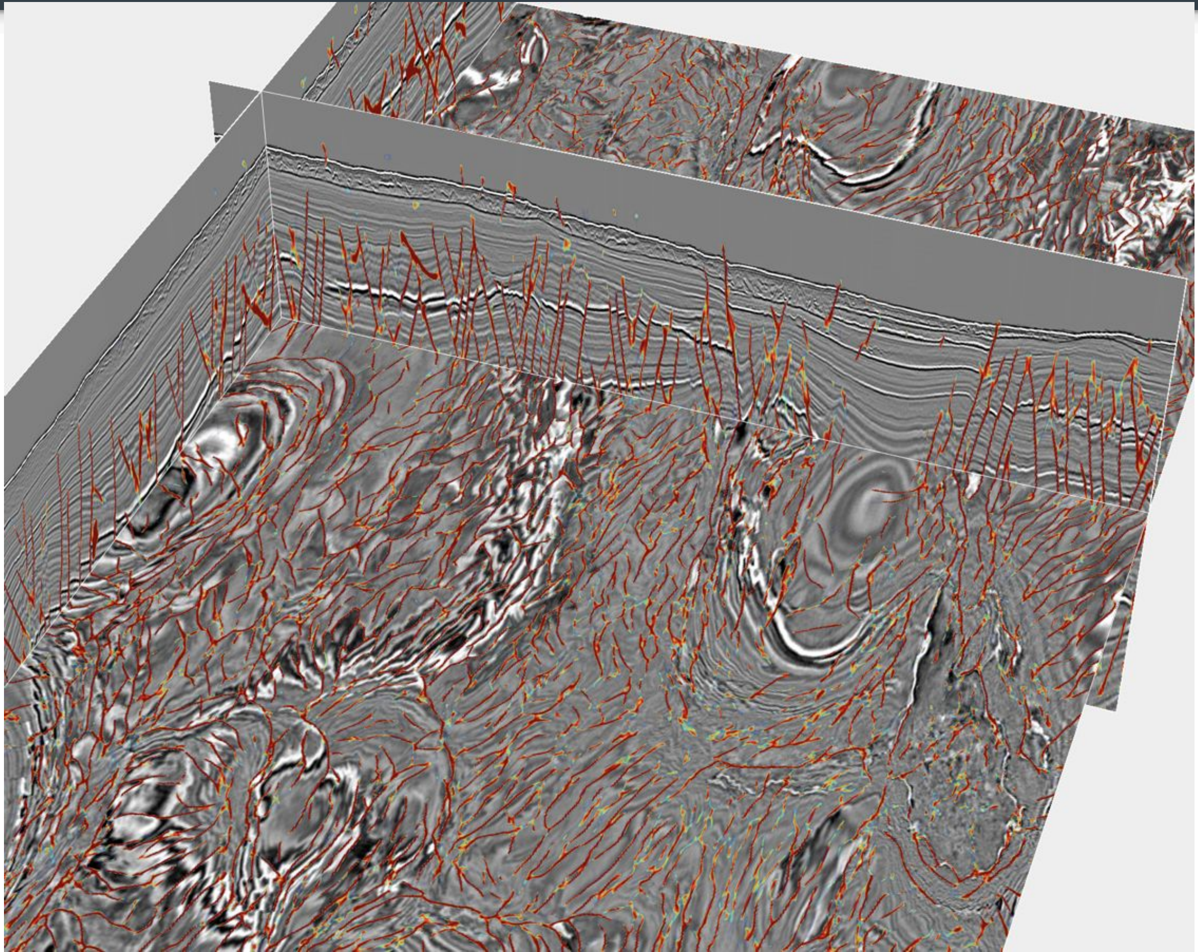
CNN fault probability



FaultSeg method only takes 2-3 min with a Titan Xp GPU!

450 x 1950 x 1200 samples

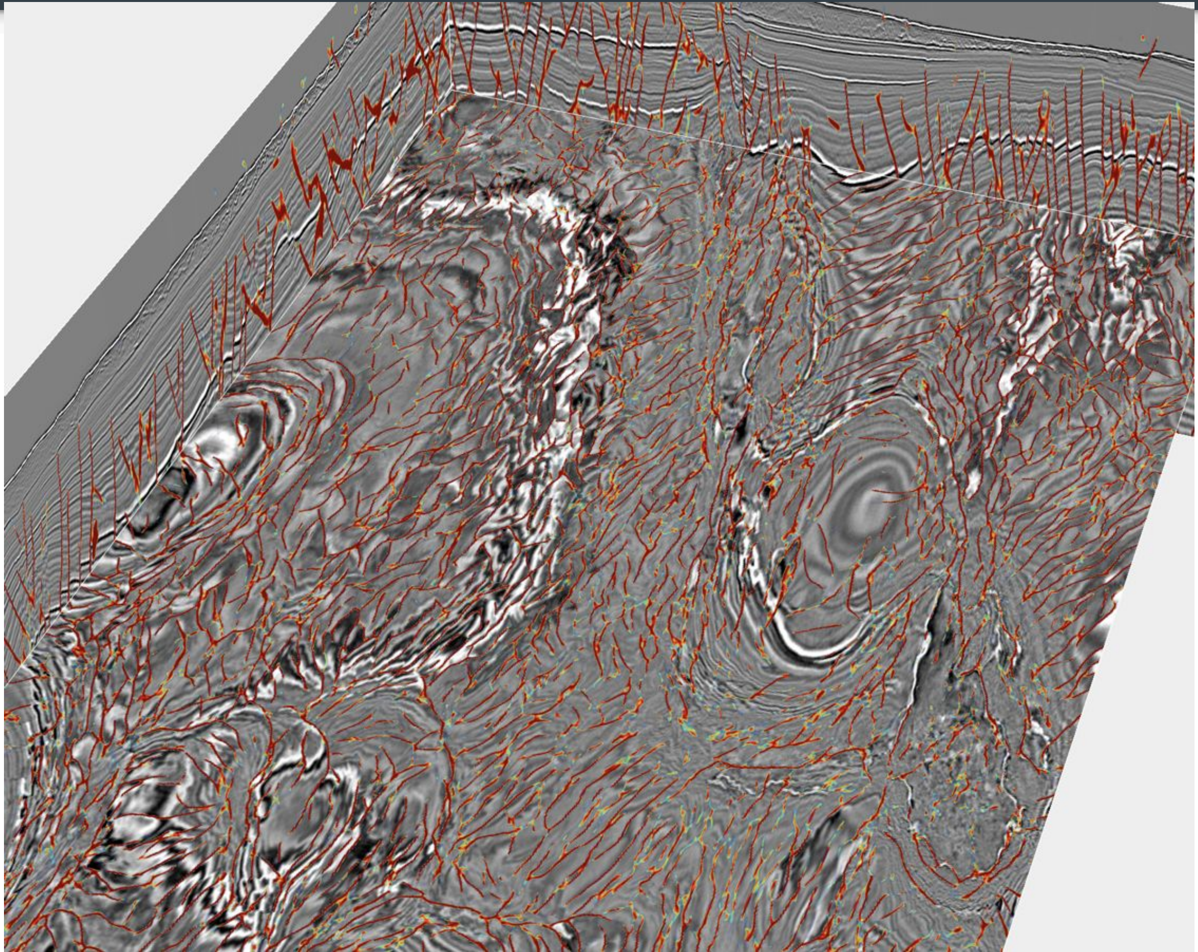
CNN fault probability



FaultSeg method only takes 2-3 min with a Titan Xp GPU!

450 x 1950 x 1200 samples

CNN fault probability



FaultSeg method only takes 2-3 min with a Titan Xp GPU!

450 x 1950 x 1200 samples

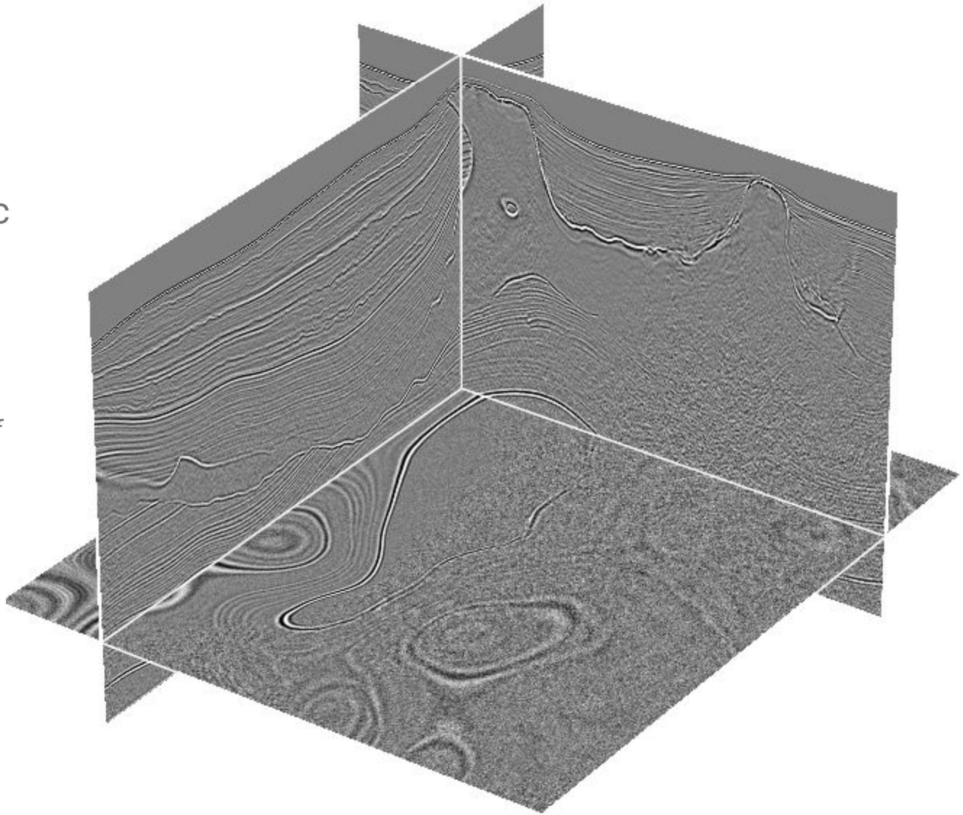
Outline

- Simple fault classification
- Generating geophysical synthetic training data
- From classification to segmentation:
  - Fault segmentation
  - **Salt body segmentation**
  - Channel segmentation
- Tracking geobody in a recurrent style
- Predicting relative geological time (RGT)

Training data

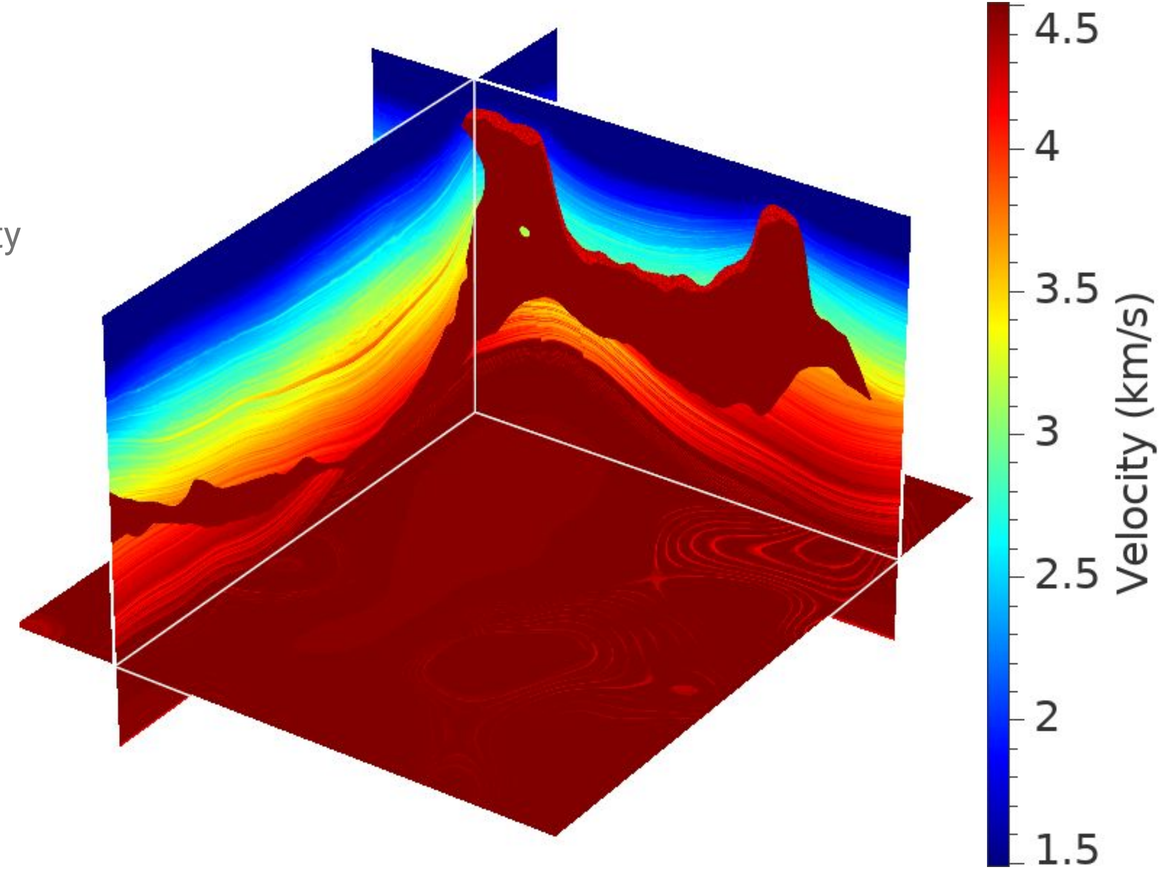
SEAM Phase I seismic volume:

Fehler and Keliher, 2011, SEAM phase I: Challenges of subsalt imaging in tertiary basins, with emphasis on deepwater Gulf of Mexico: SEG.



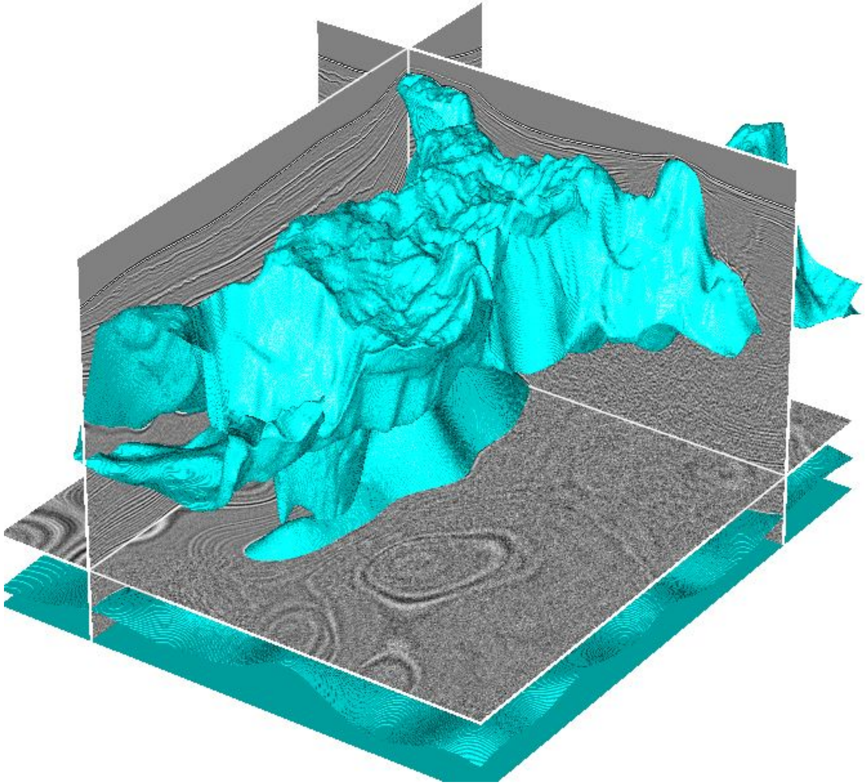
Extracting salt body mask from velocity model

SEAM Phase I velocity model:

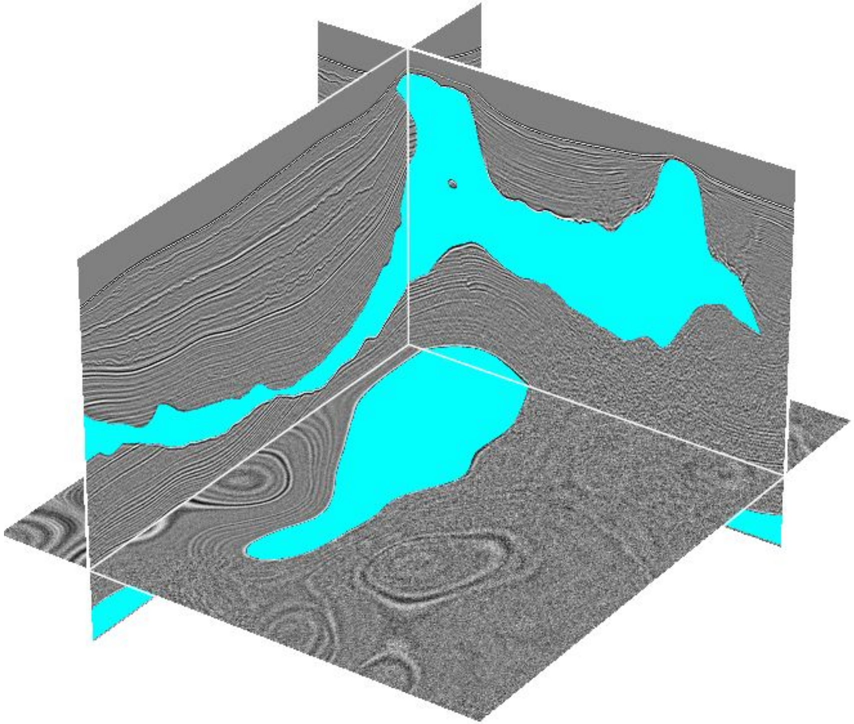


Extracting salt body mask from velocity model

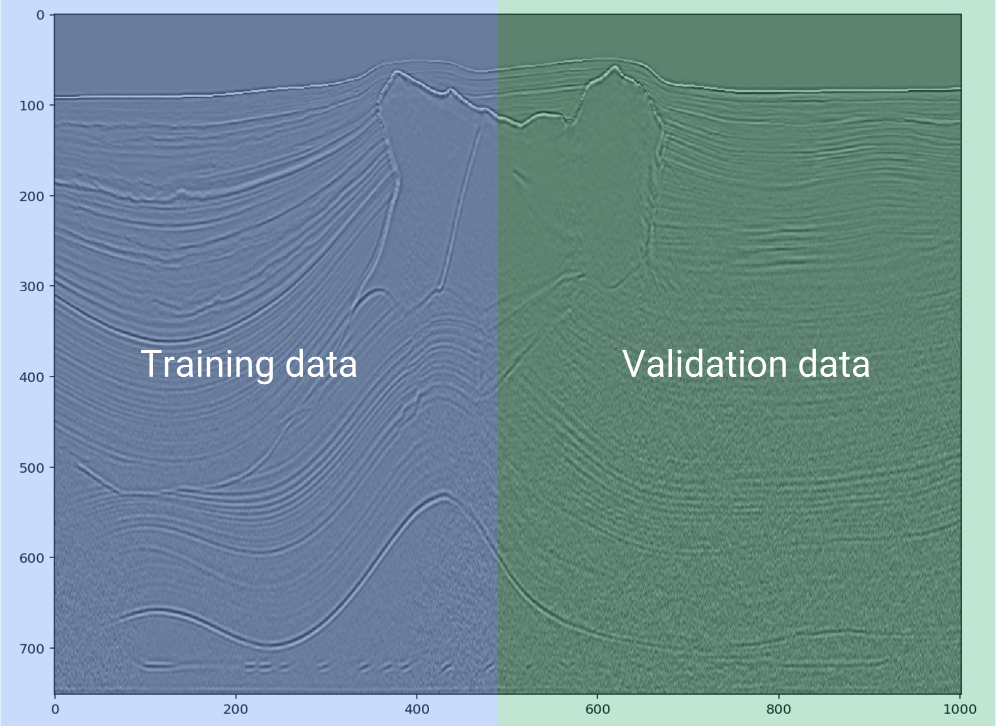
True salt model



Salt slice view



Split training and validation data

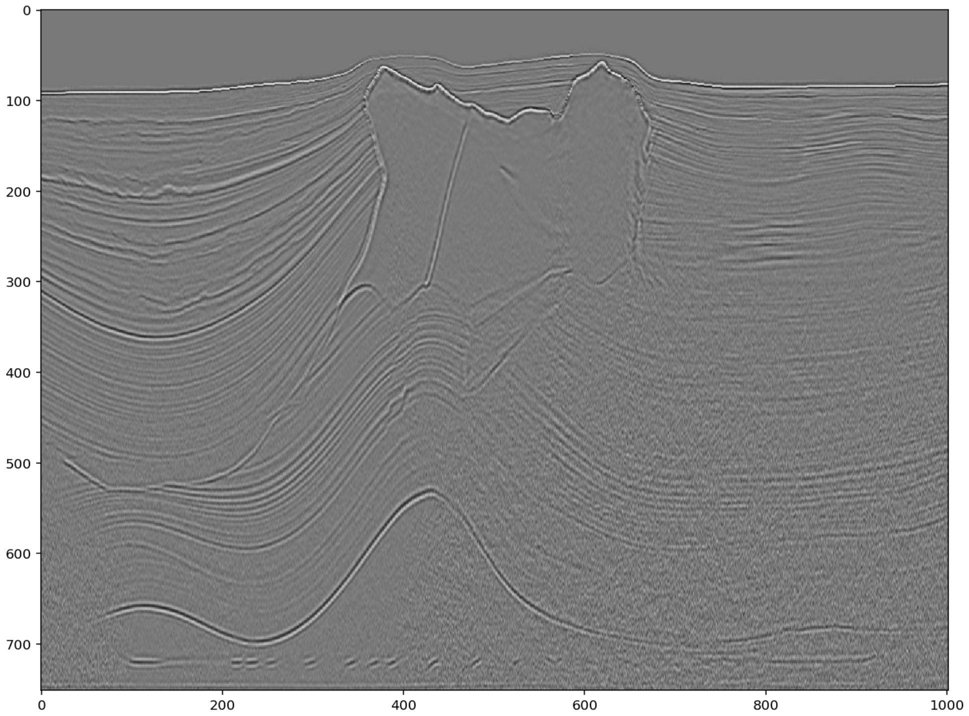


Size limitation — sliding window prediction

Dump the whole volume into training is infeasible.

We crop sliding windows from the volume:

- Predict on each window
- Merge all windows back the size of the volume

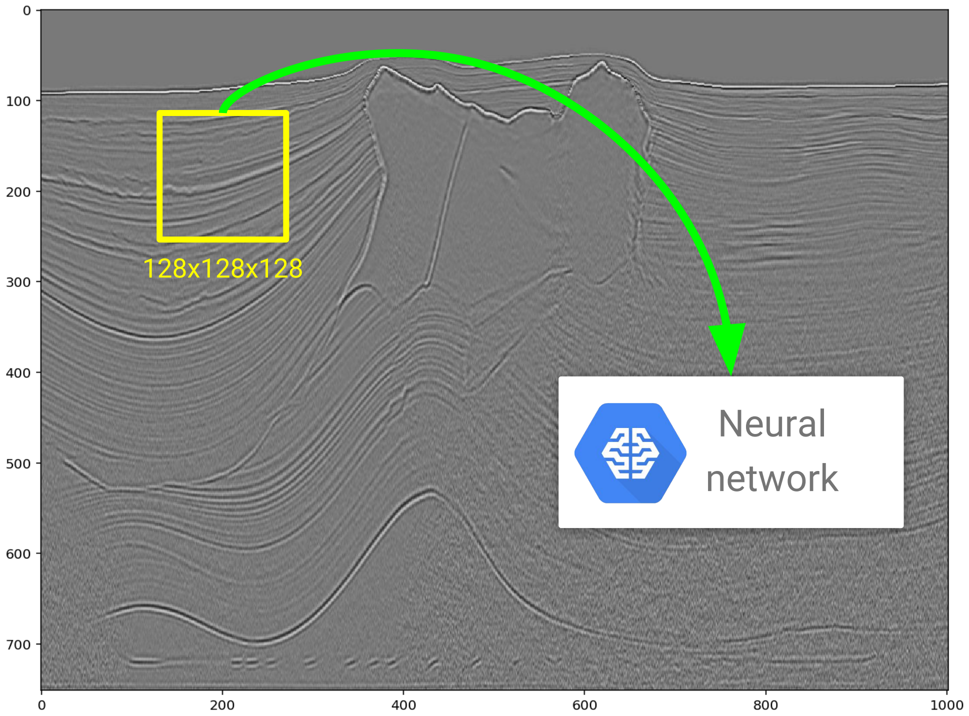


Size limitation — sliding window prediction

Dump the whole volume into training is infeasible.

We crop sliding windows from the volume:

- Predict on each window
- Merge all windows back the size of the volume

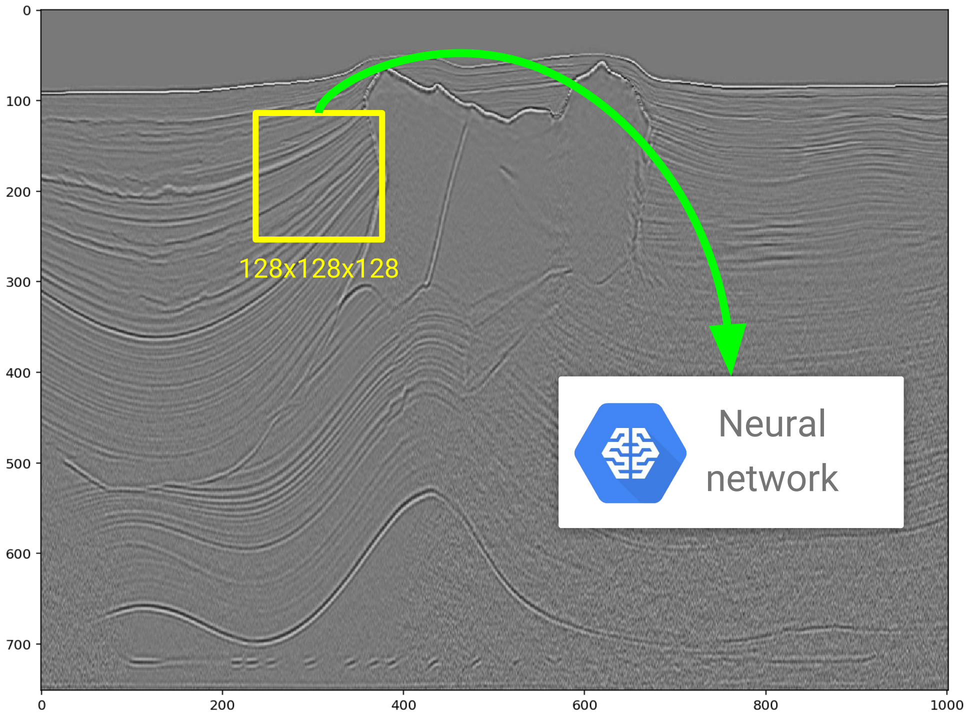


Size limitation — sliding window prediction

Dump the whole volume into training is infeasible.

We crop sliding windows from the volume:

- Predict on each window
- Merge all windows back the size of the volume

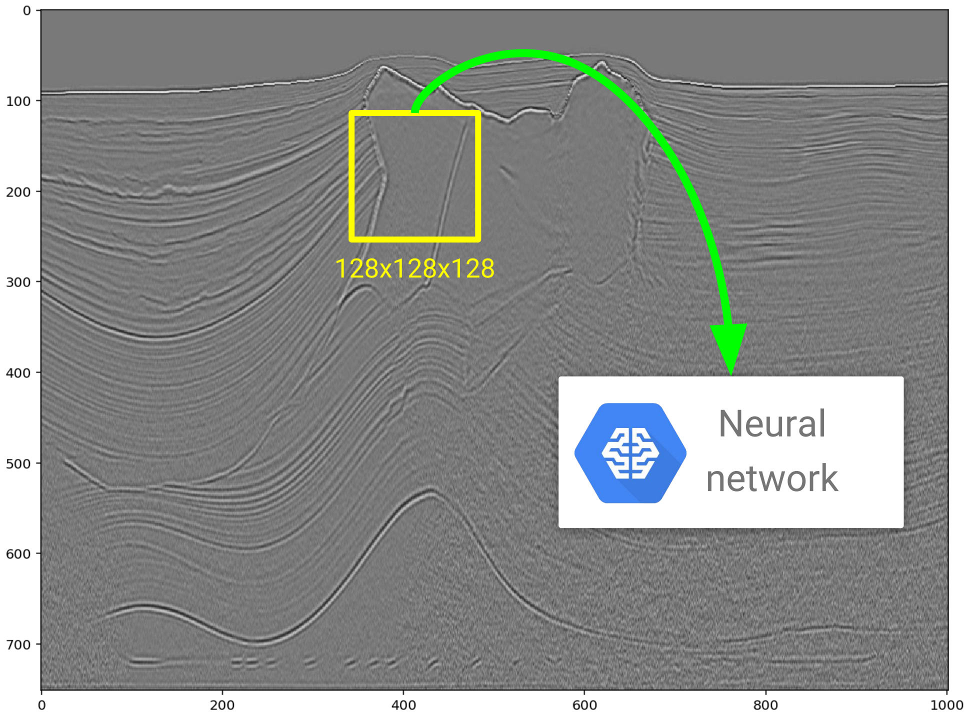


Size limitation — sliding window prediction

Dump the whole volume into training is infeasible.

We crop sliding windows from the volume:

- Predict on each window
- Merge all windows back the size of the volume

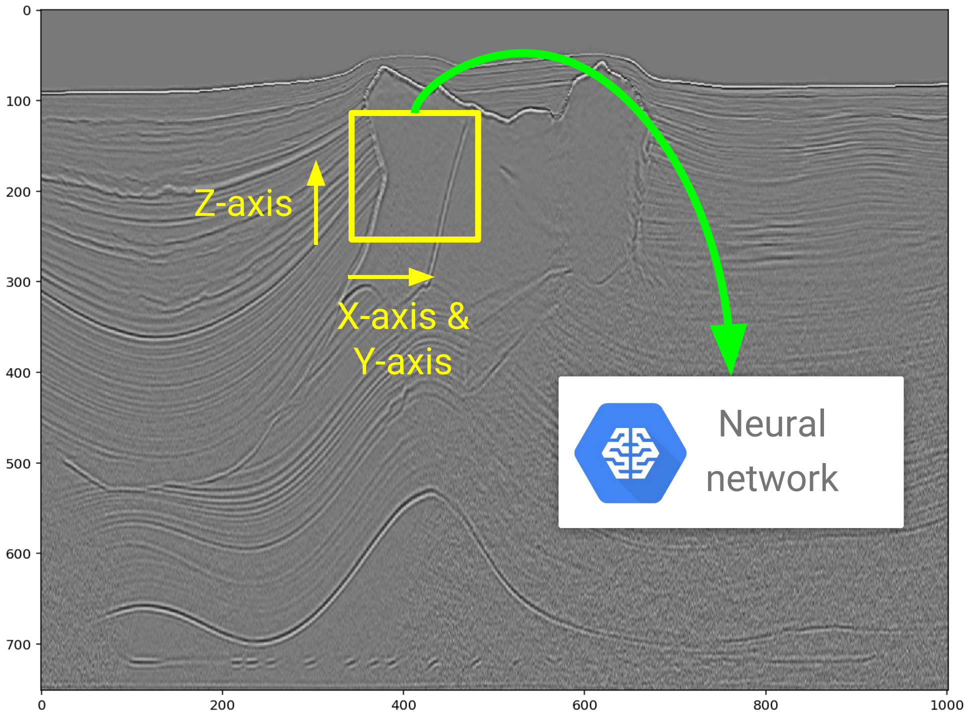


Size limitation — sliding window prediction

We build a data generator that randomly crop a window from the volume.

Note that:

- Z-axis is not permutable;
- X-axis and Y-axis are permutable.
- **We can reflect / rotate in X-Y dimension!**

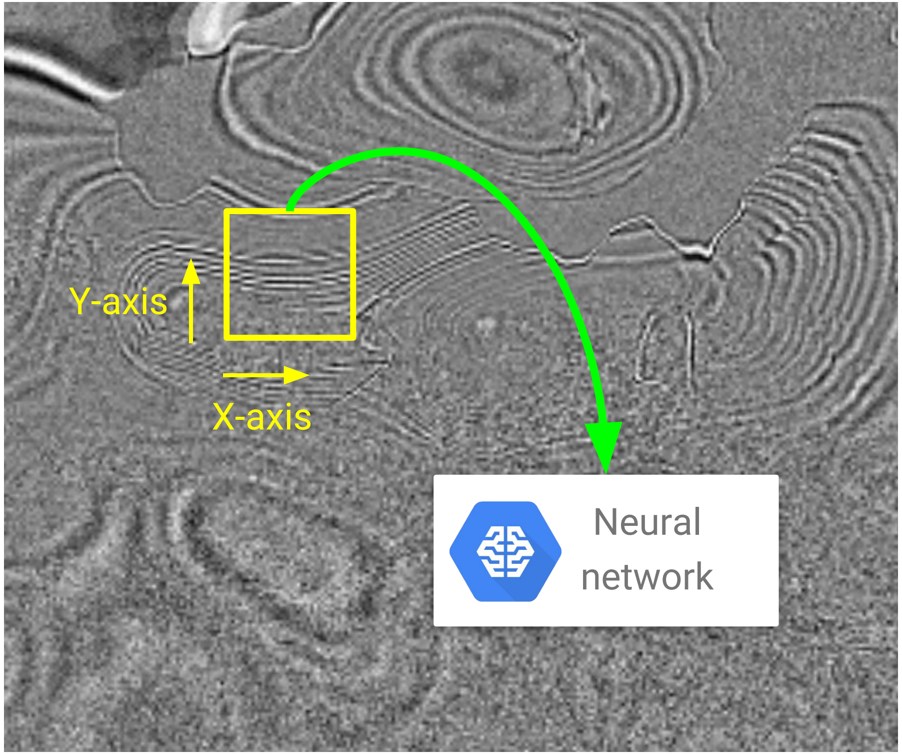


Size limitation — sliding window prediction

We build a data generator that randomly crop a window from the volume.

Note that:

- Z-axis is not permutable;
- X-axis and Y-axis are permutable.
- **We can reflect / rotate in X-Y dimension!**

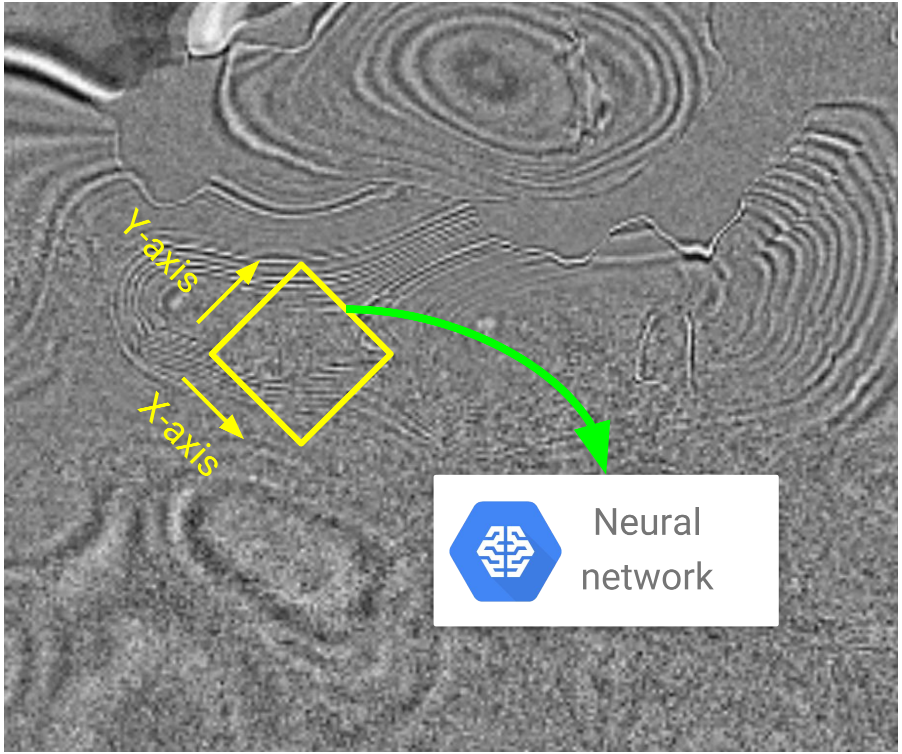


Size limitation — sliding window prediction

We build a data generator that randomly crop a window from the volume.

Note that:

- Z-axis is not permutable;
- X-axis and Y-axis are permutable.
- **We can reflect / rotate in X-Y dimension!**

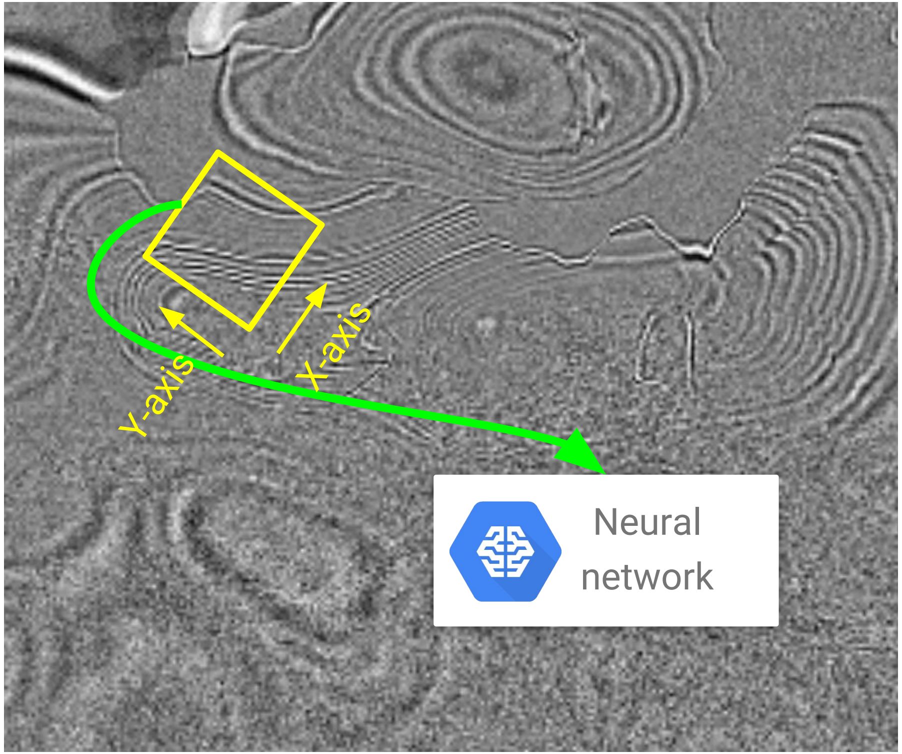


Size limitation — sliding window prediction

We build a data generator that randomly crop a window from the volume.

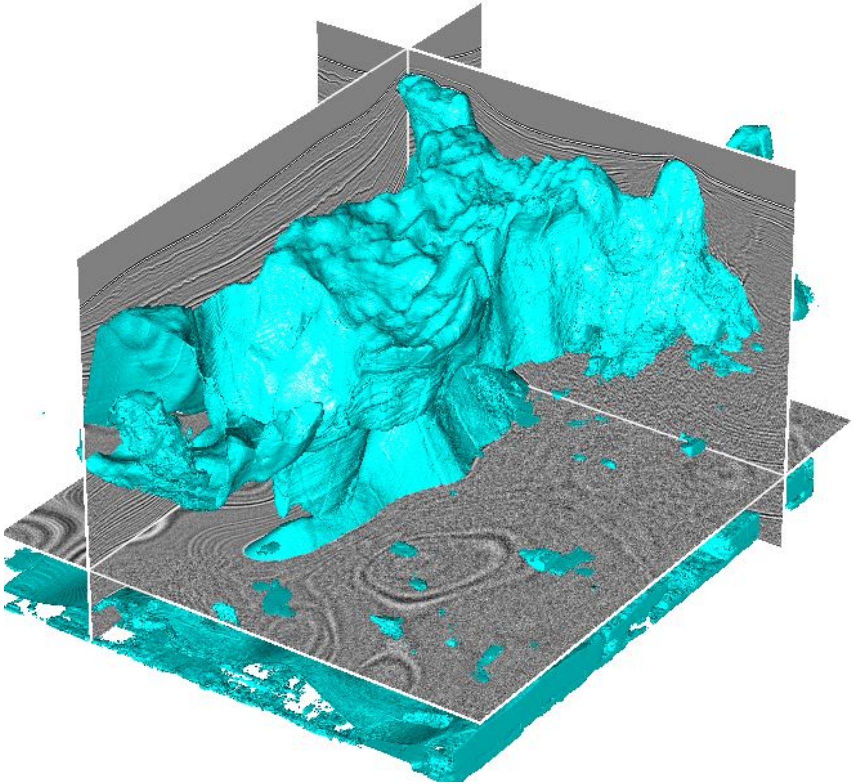
Note that:

- Effectively we can generate infinite number of data samples;
- The rotated window is a powerful **augmentation technique** that improves the **model generalization**.

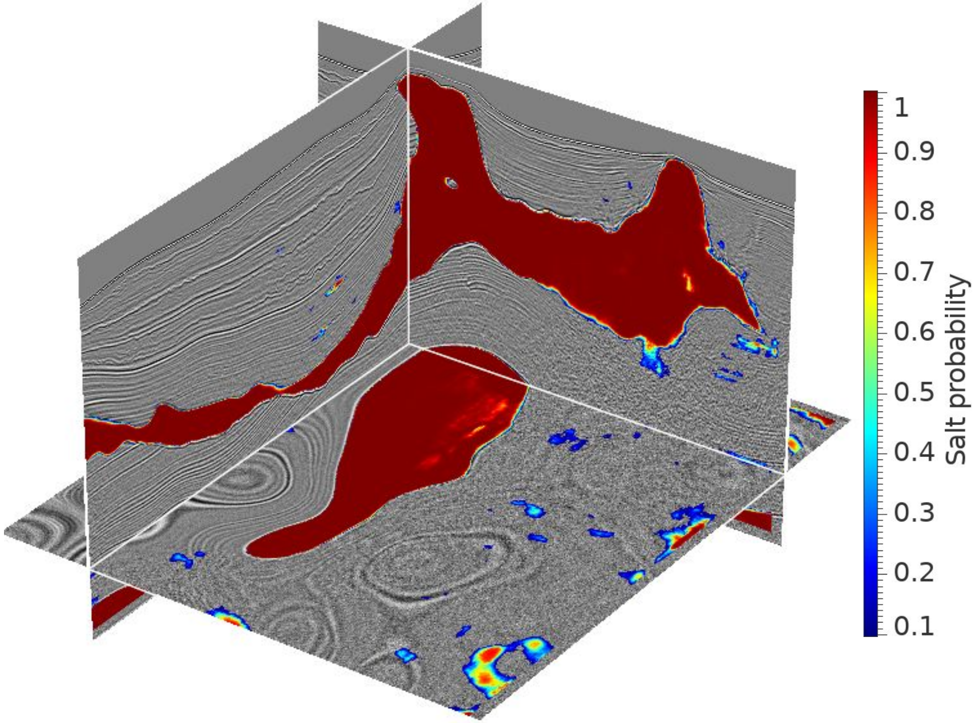


Result showcase — SEAM Phase I

Predicted salt model

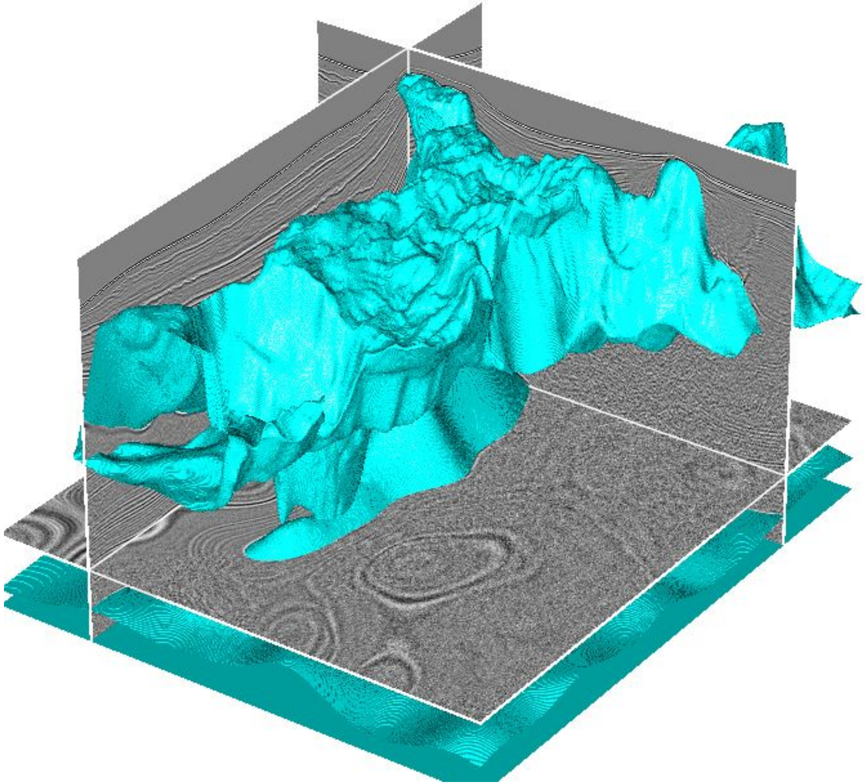


Prediction probability

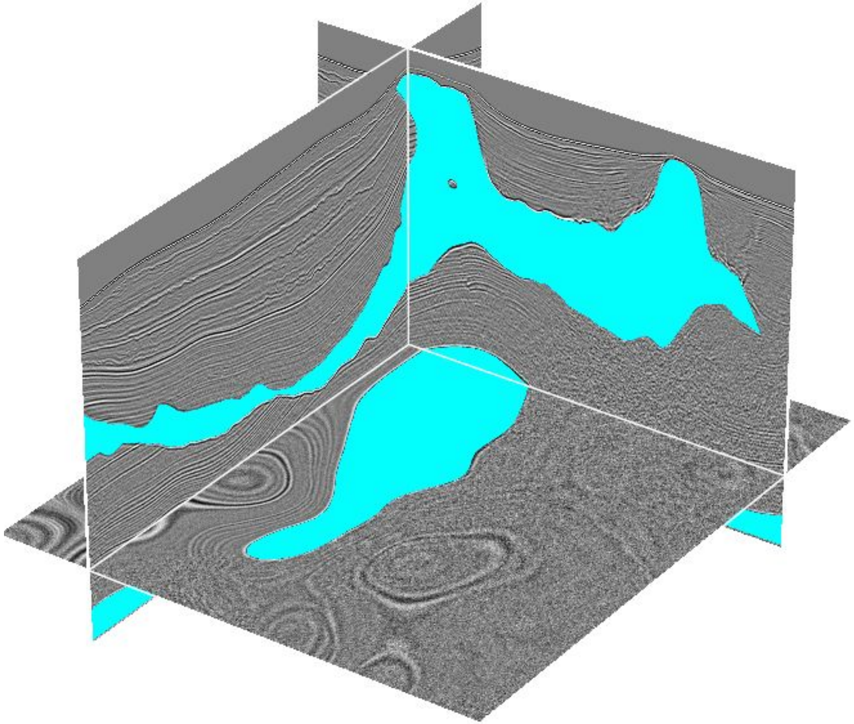


Result showcase — SEAM Phase I

True salt model



True slice view



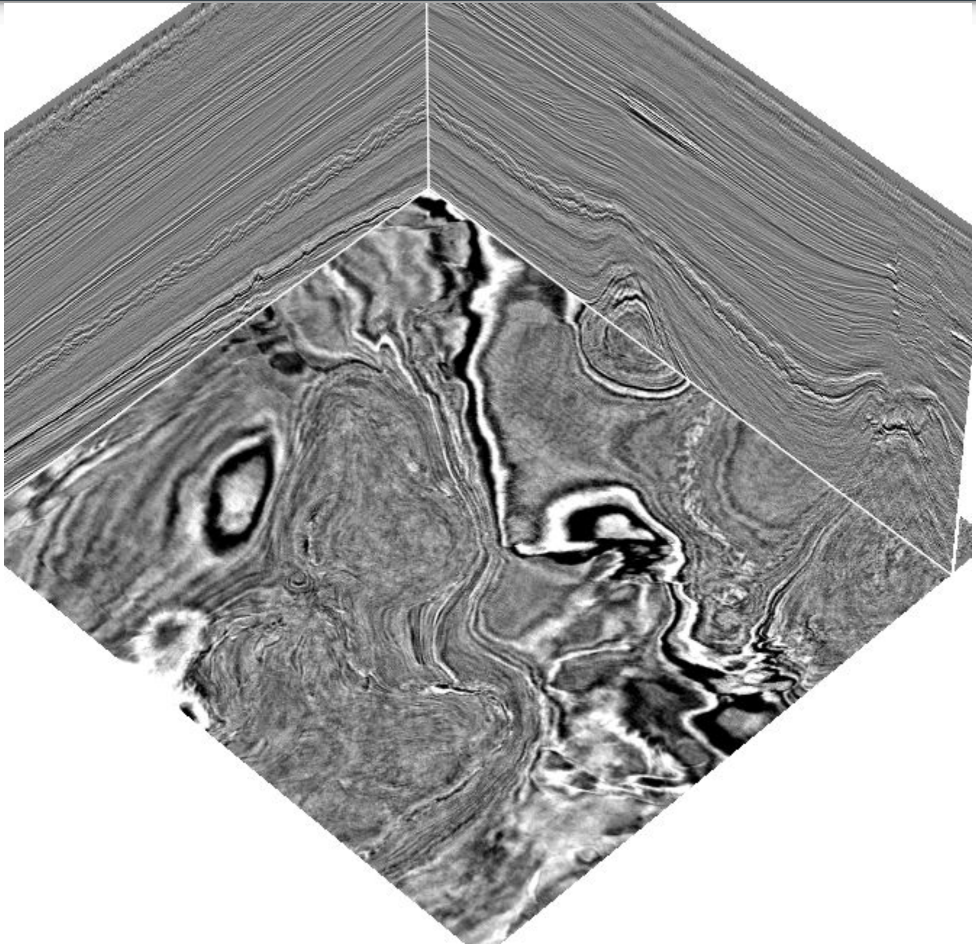
Result showcase — SEAM Phase I

Metric name	Metric scores	Metric definition
Accuracy	**0.9609**	$\text{Accuracy} = \dfrac{T_P + T_N}{T_P + F_P + F_N + T_N}$
Precision	**0.9004**	$\text{Precision} = \dfrac{T_P}{T_P + F_P}$
Recall	**0.9468**	$\text{Recall} = \dfrac{T_P}{T_P + F_N}$
F1 score	**0.9230**	$\text{F1} = \dfrac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$

Test on the field data

Netherlands F3
seismic volume:

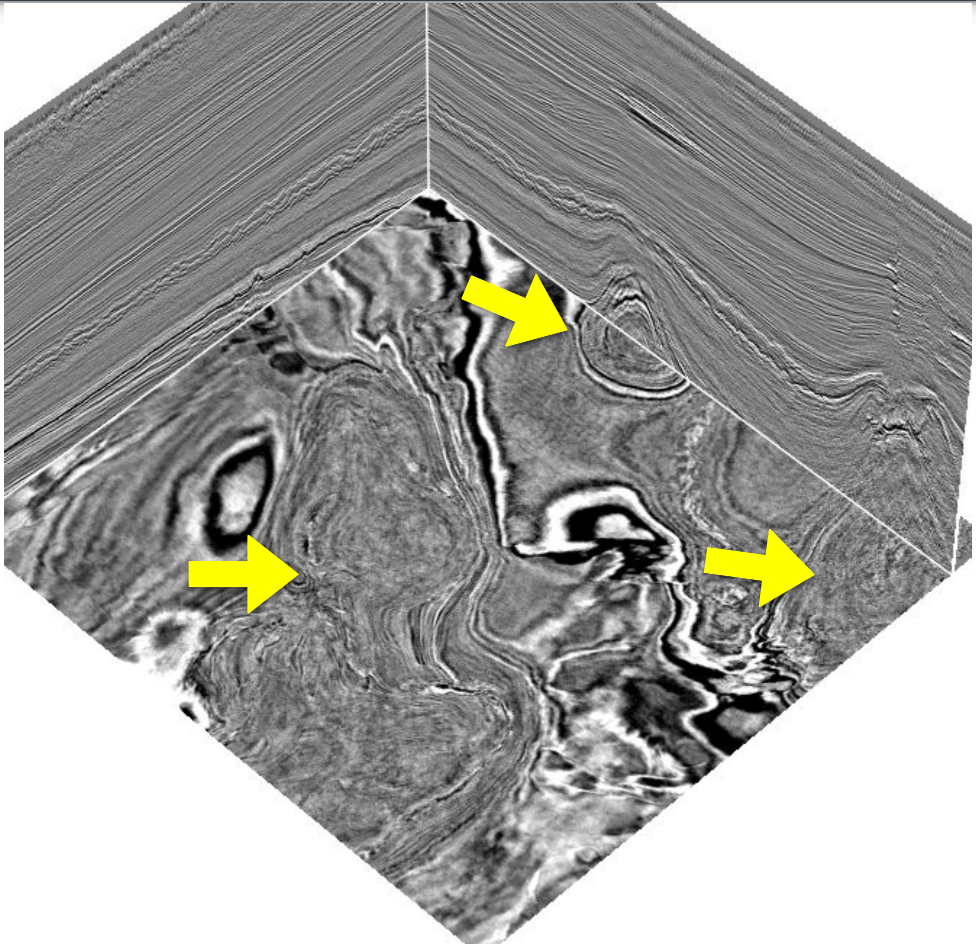
From dGB Earth Sciences B.V., https://opendtect.org/osr/Main/NetherlandsOffshoreF3BlockComplete4GB



Test on the field data

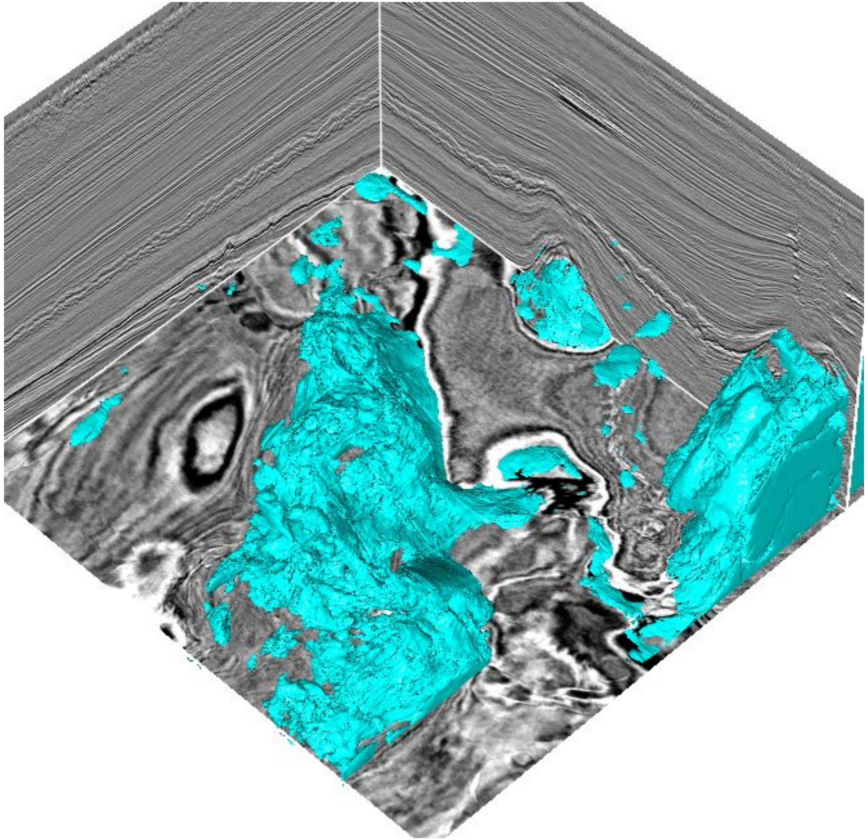
Netherlands F3 seismic volume:

From dGB Earth Sciences B.V., https://opendtect.org/osr/Main/NetherlandsOffshoreF3BlockComplete4GB

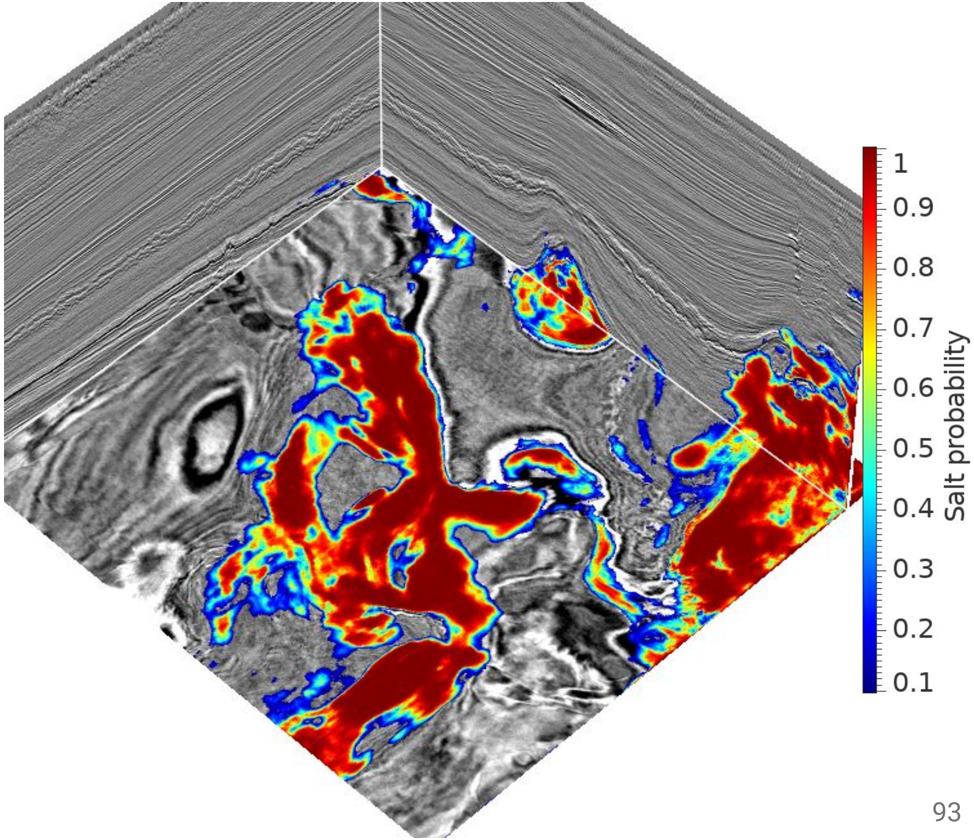


Test on the field data

Predicted salt model



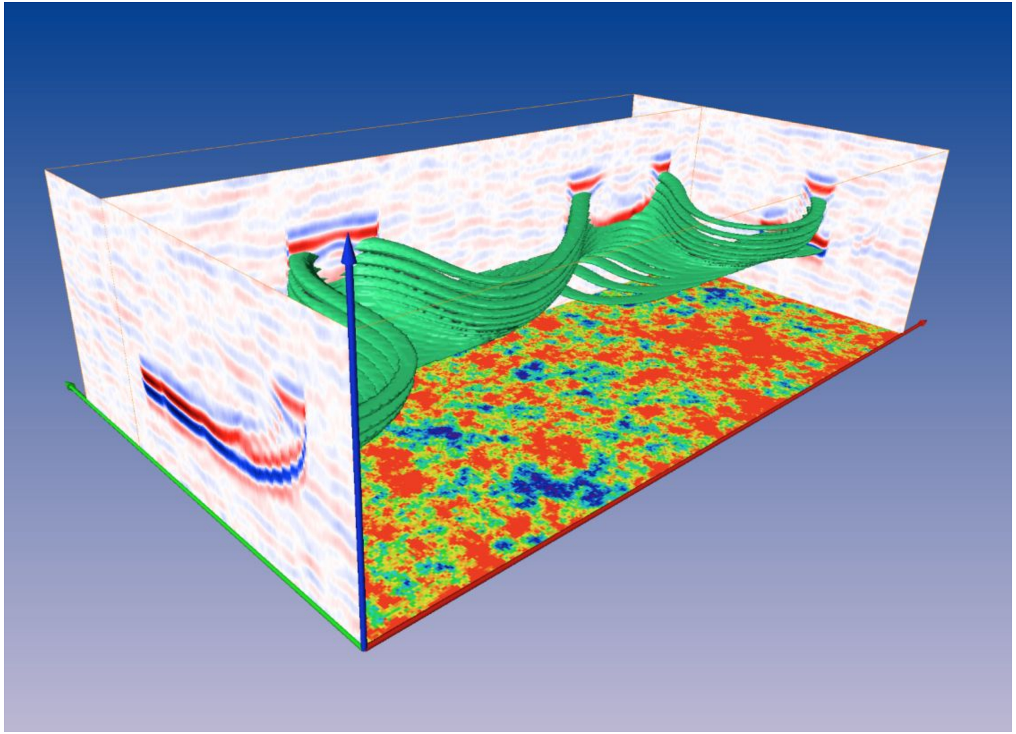
Prediction probability



Outline

- Simple fault classification
- Generating geophysical synthetic training data
- From classification to segmentation:
  - Fault segmentation
  - Salt body segmentation
  - **Channel segmentation**
- Tracking geobody in a recurrent style
- Predicting relative geological time (RGT)

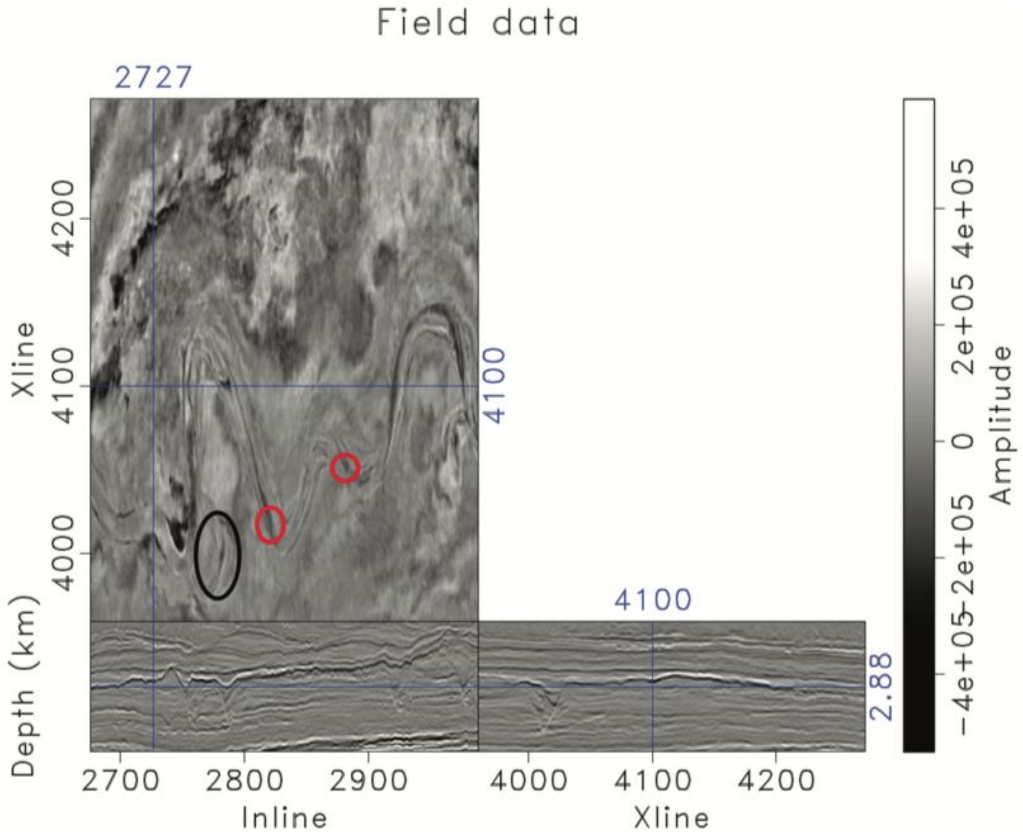
Channel detection example



The synthetic dataset is created from 3 pieces of information:

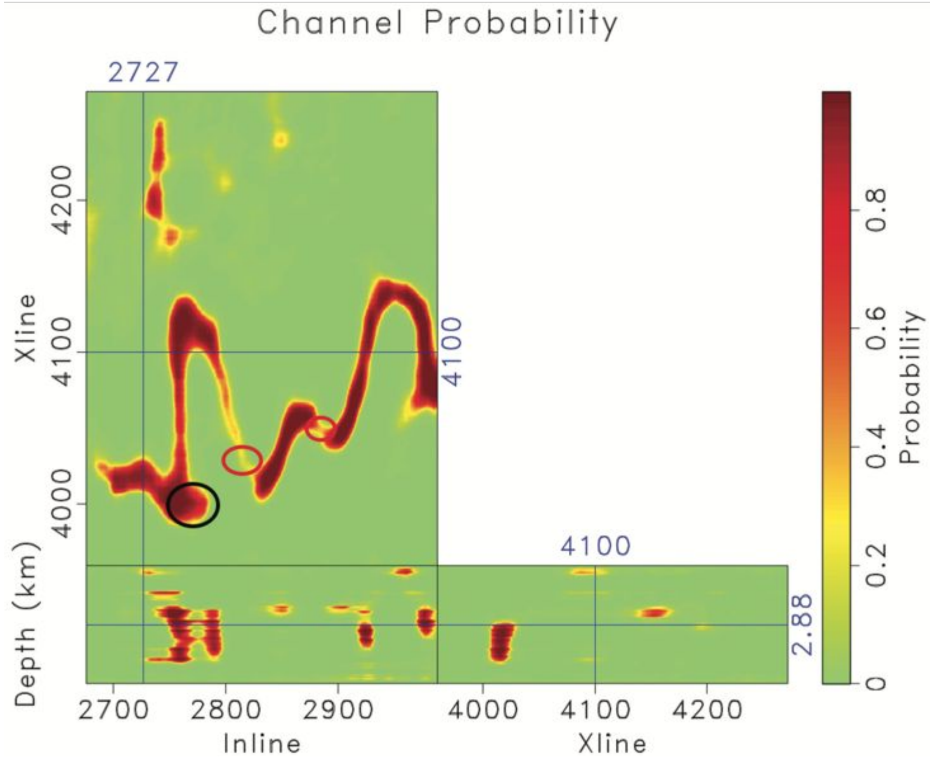
- Geological outcrop information from expert geologists.
- 3D shallow high resolution seismic data.
- Geostatistics information

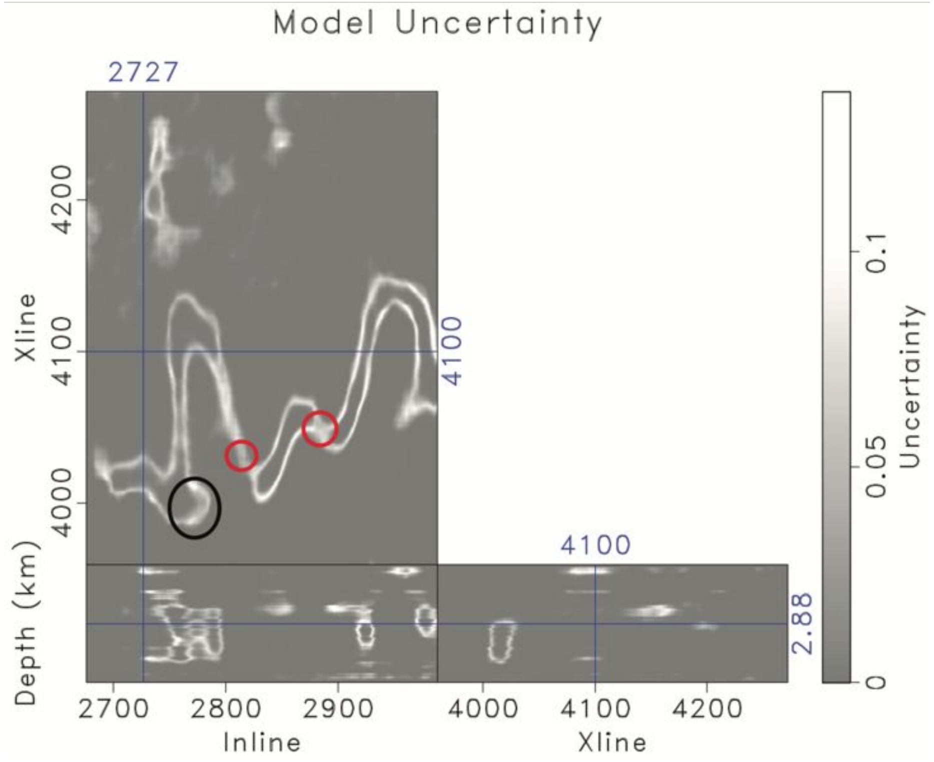
Channel detection example



- 3D offshore Australia released dataset from QCL group.
- Complex stacked channel system.

Channel detection example





Channel detection example

	156x156x100 sample	501x750x251 sample	312x312x100 sample
Deep learning	12.5 seconds	33.3 minutes	3.3 minutes
Conventional	About an hour	About 6 days	About 16 hours

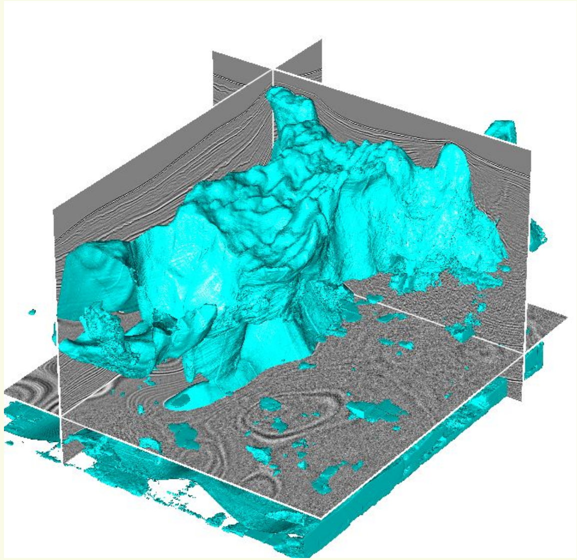
- Deep learning time is for generating a distribution of 30 detections of channel geobodies (GPU).
- Manual detection of channel geobodies time is approximate (CPU).

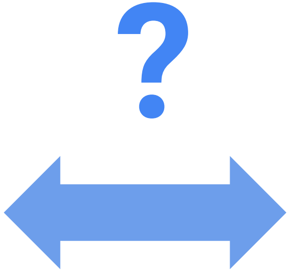
Outline

- Simple fault classification
- Generating geophysical synthetic training data
- From classification to segmentation:
  - Fault segmentation
  - Salt body segmentation
  - Channel segmentation
- **Tracking geobody in a recurrent style**
- Predicting relative geological time (RGT)

Attribute vs instance

Segmentation attribute





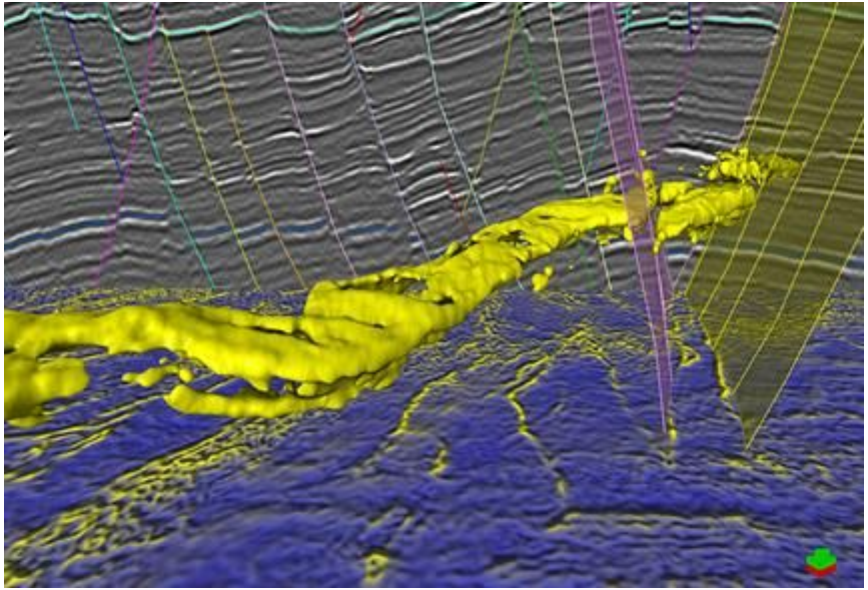


Figure from Petrel, Schlumberger

Current solutions

Skeletonization algorithm

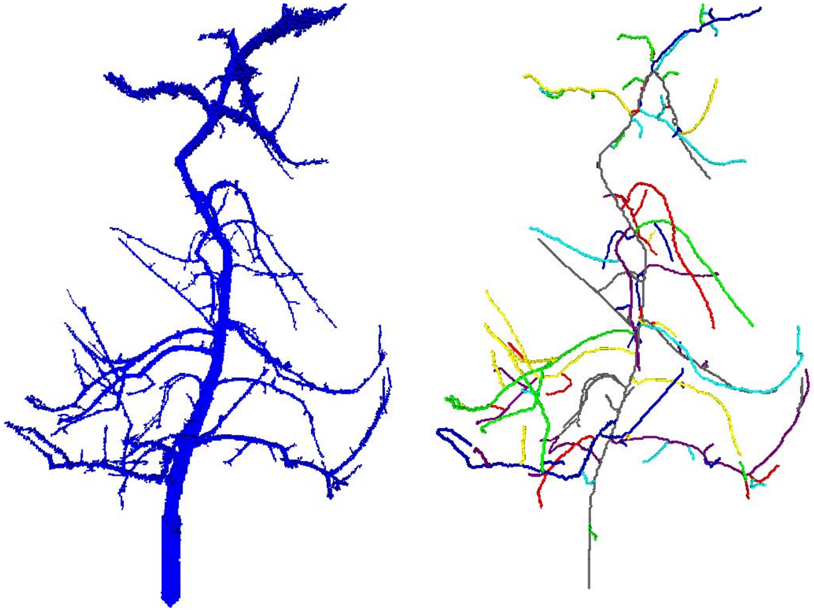


Figure from Tabb, Amy, and Henry Medeiros. "Fast and robust curve skeletonization for real-world elongated objects." 2018 IEEE Winter Conference on Applications of Computer Vision (WACV). IEEE, 2018.

Current solutions

Clustering analysis

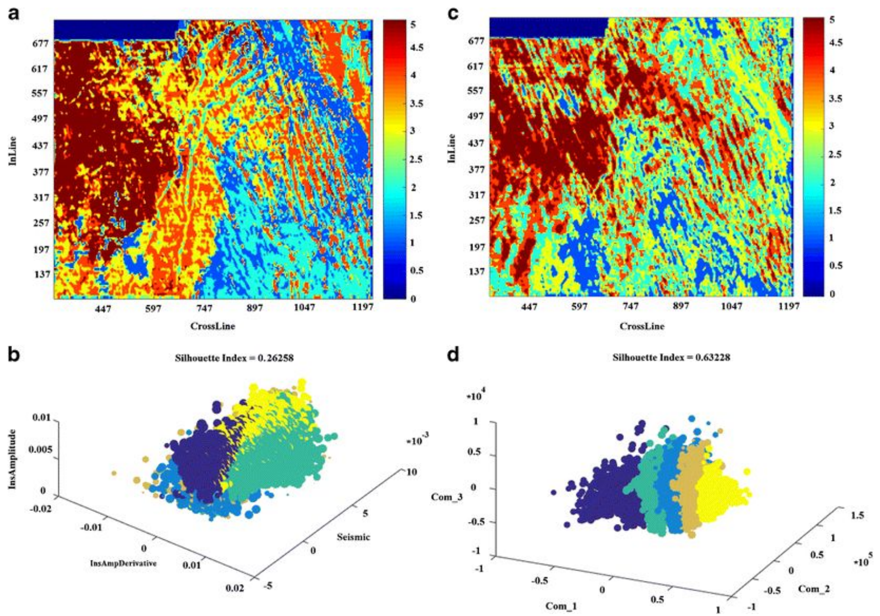


Figure from Hadiloo, S., et al. "SeisART software: seismic facies analysis by contributing interpreter and computer." Arabian Journal of Geosciences 10.23 (2017): 519.

Current solutions

Tracking algorithms

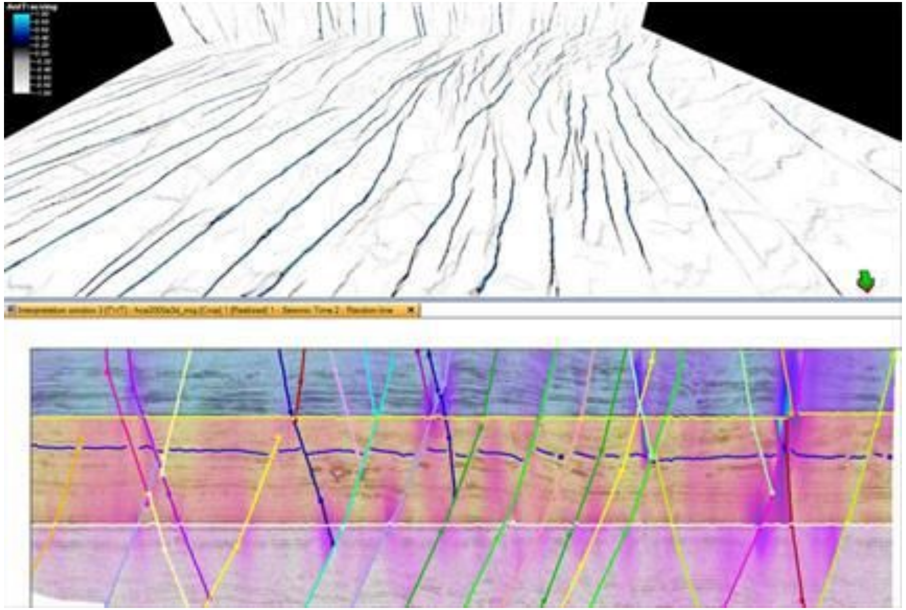
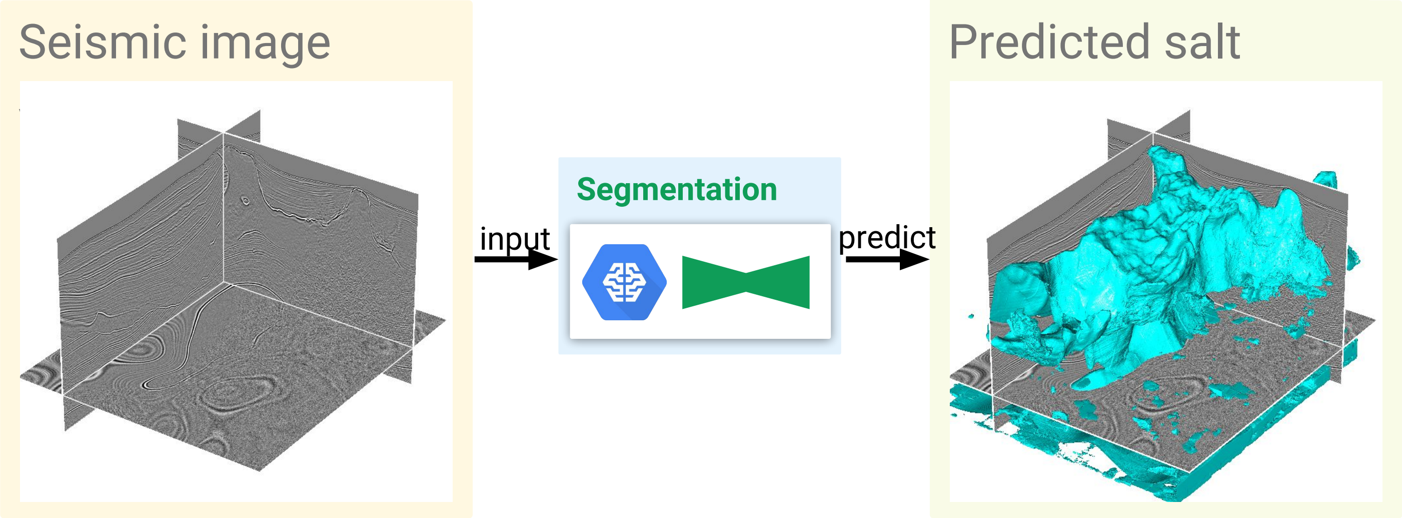


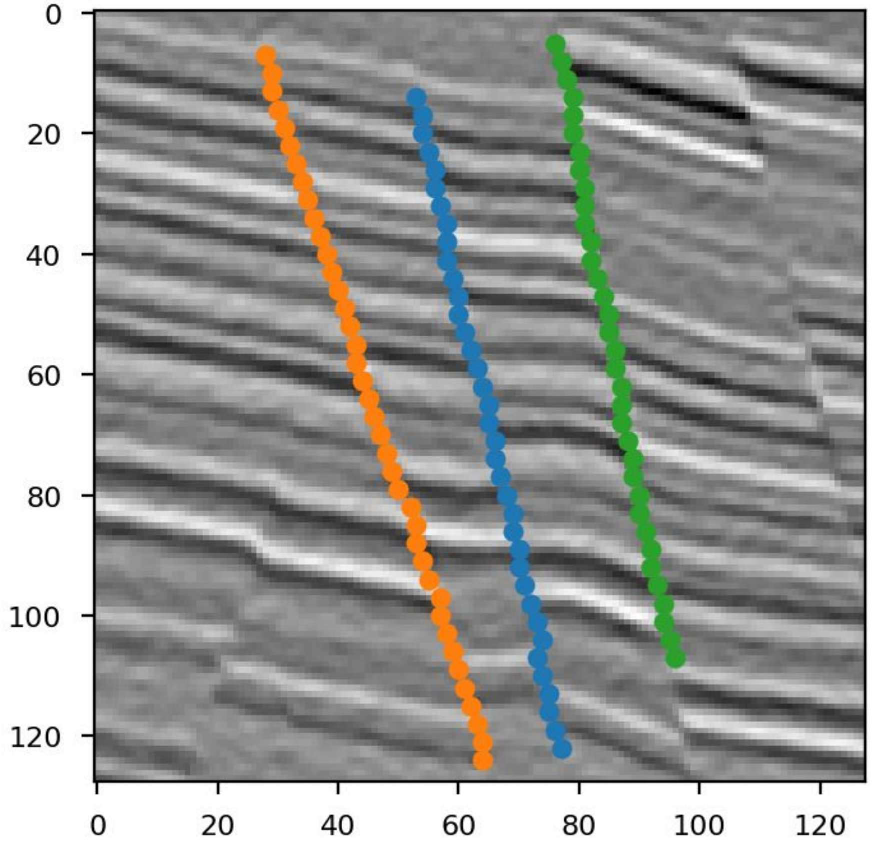
Figure from Pedersen, Stein Inge, et al. "Automatic fault extraction using artificial ants." SEG Technical Program Expanded Abstracts 2002. Society of Exploration Geophysicists, 2002. 512-515.

Disadvantages of the previous methods

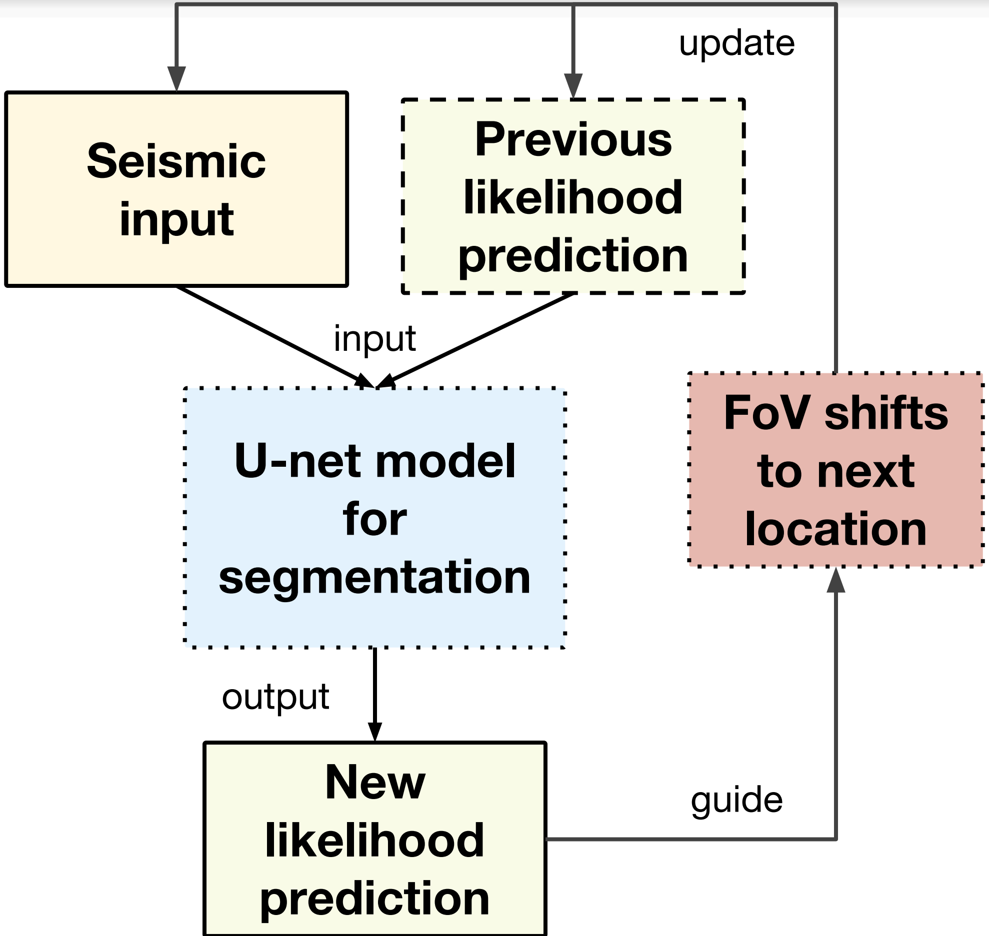
- Cannot separate individual geobody instances
- Do not allow interactivity on the user end



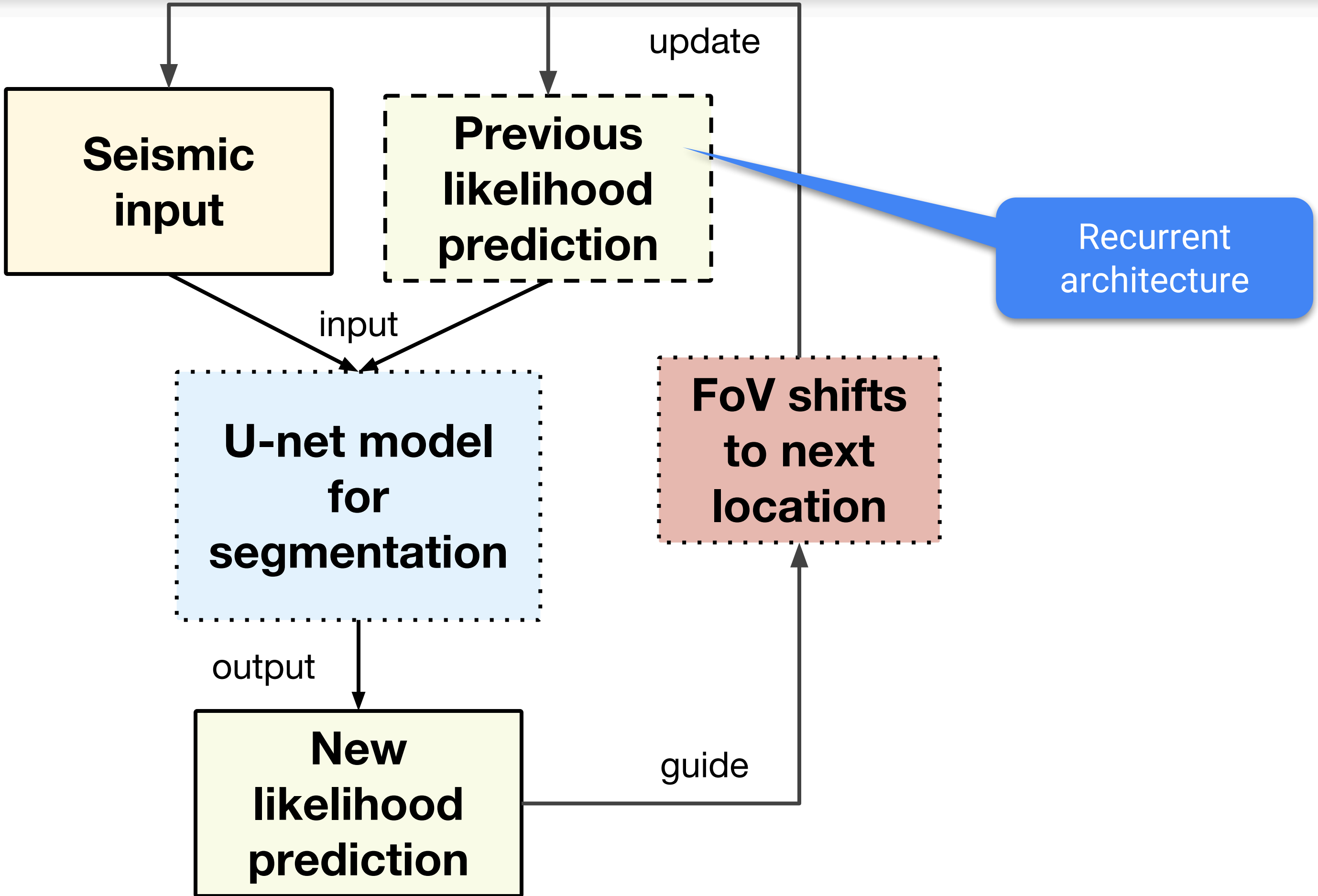
Motivations to flood-filling network (FFN)



Our architecture

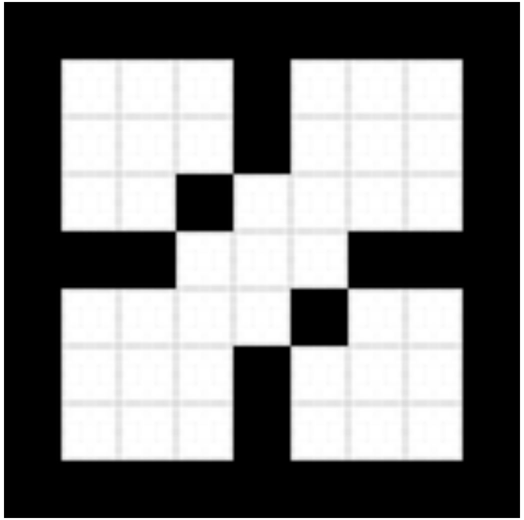


Our architecture

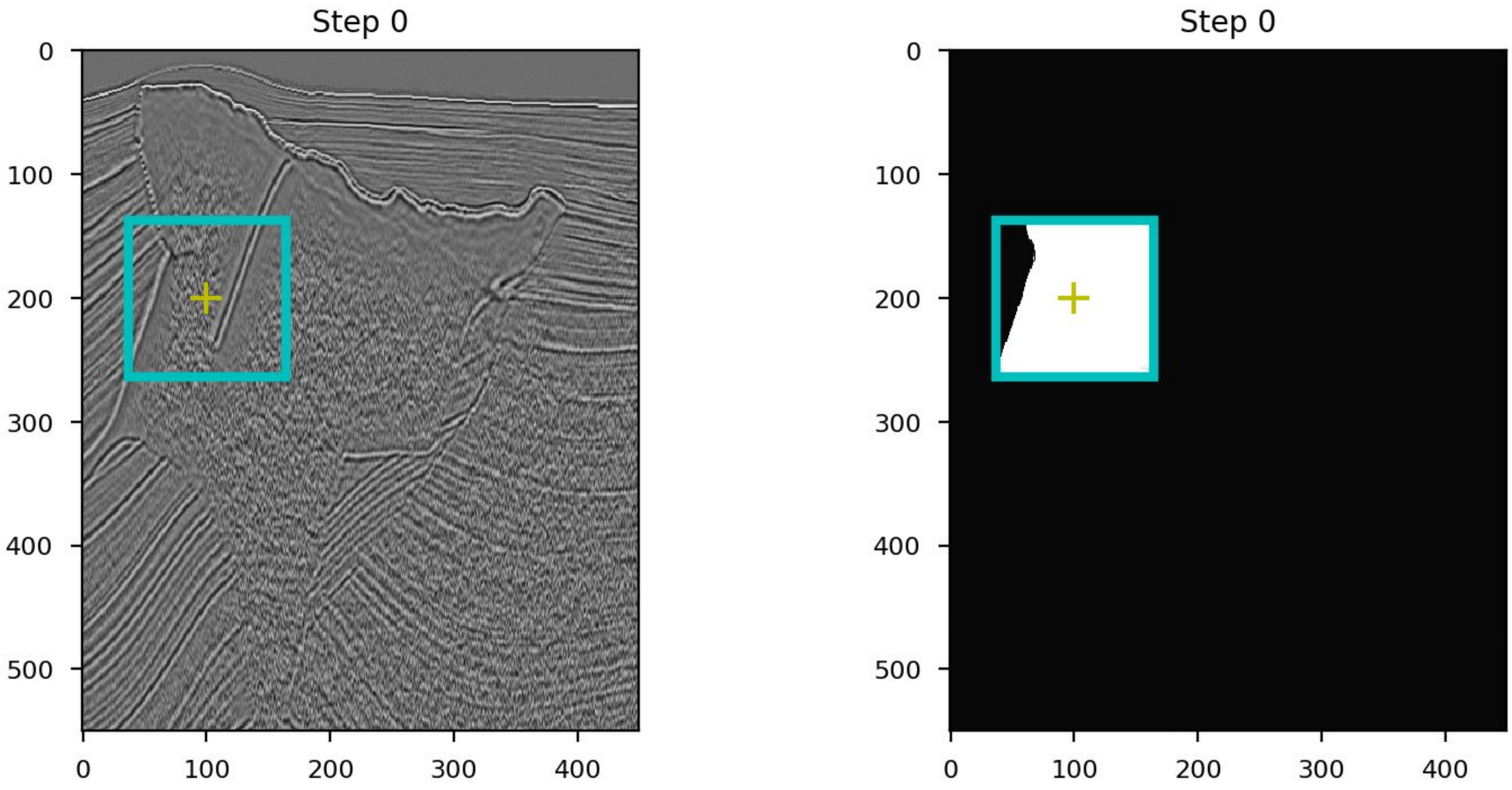


FoV movement - filling

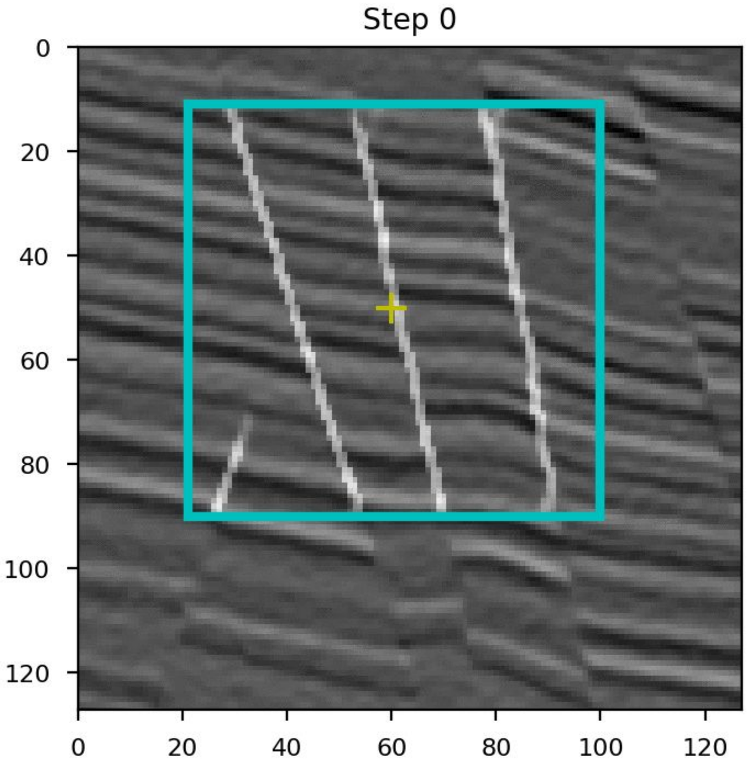
If a proposal q_i satisfies the criteria, we add it into a queue $Q = \{q_1, q_2, \ldots\}$. At the next step, Q pops a new FoV centroid and performs segmentation, likelihood update, and movement proposal in this fashion iteratively until Q is exhausted.

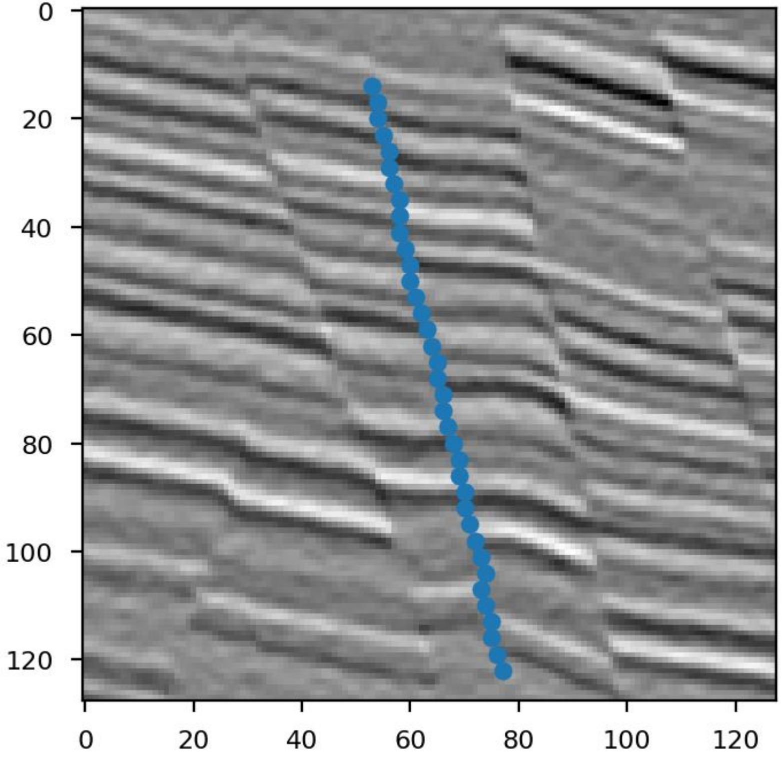


Results preview

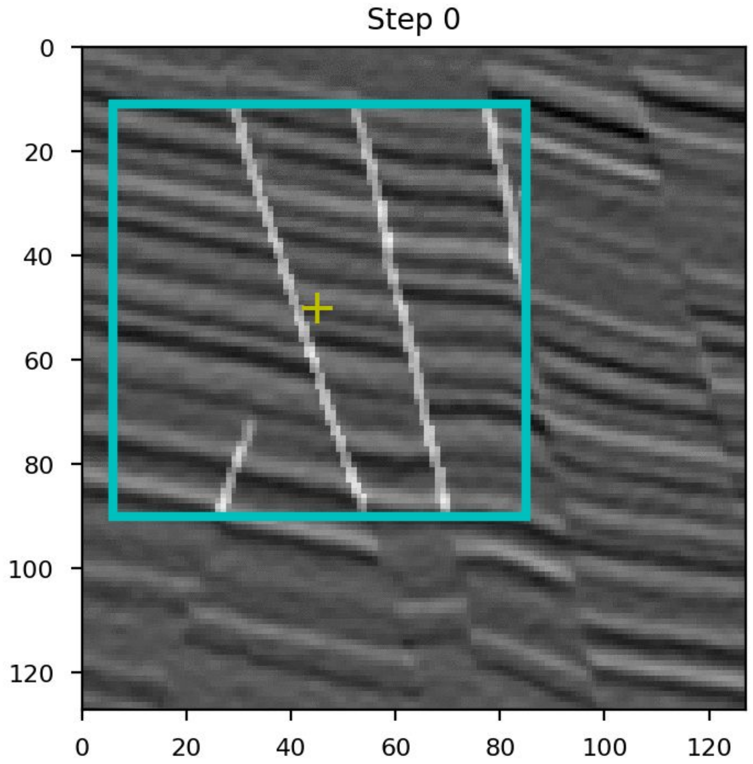


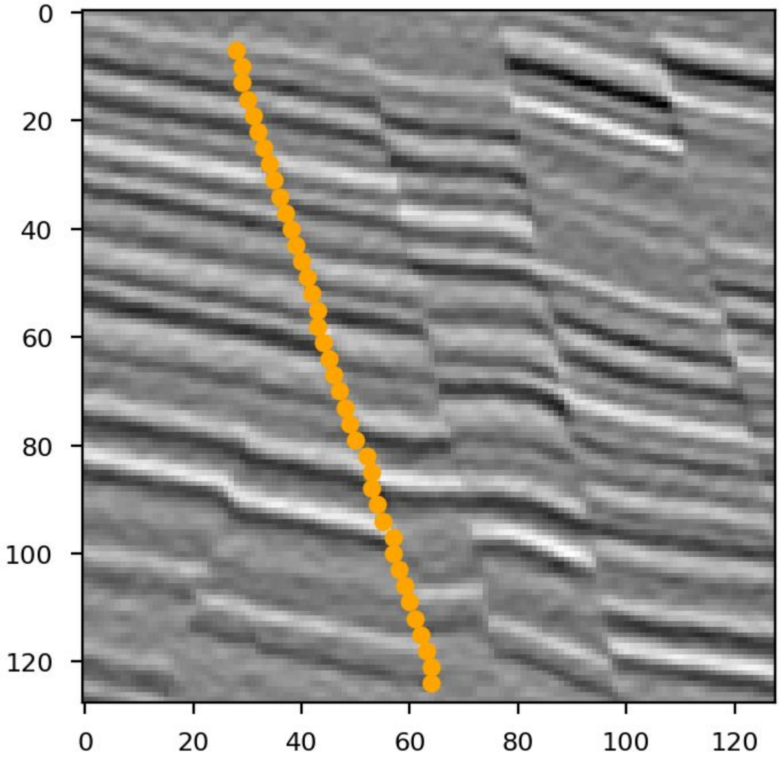
Multi-instances separation



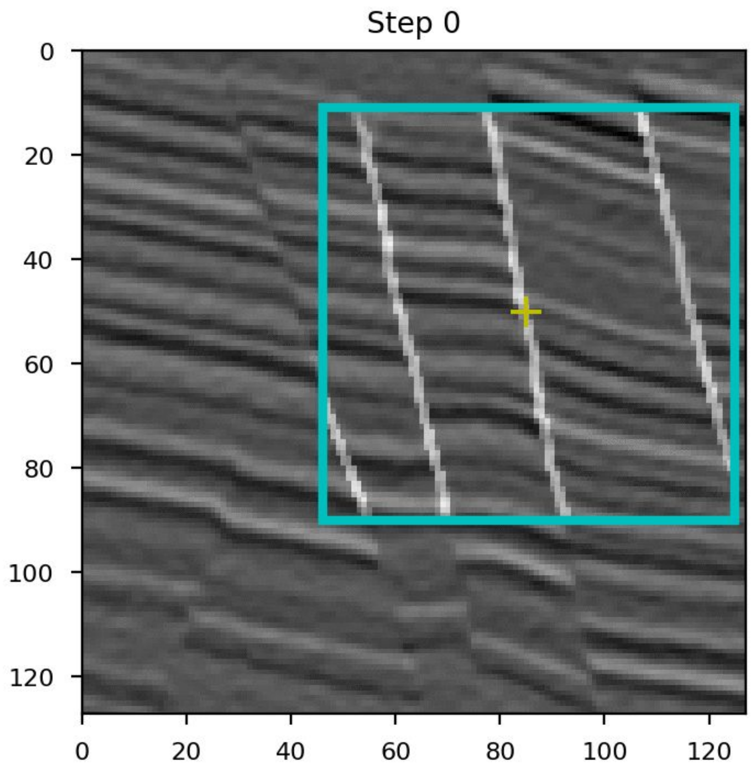


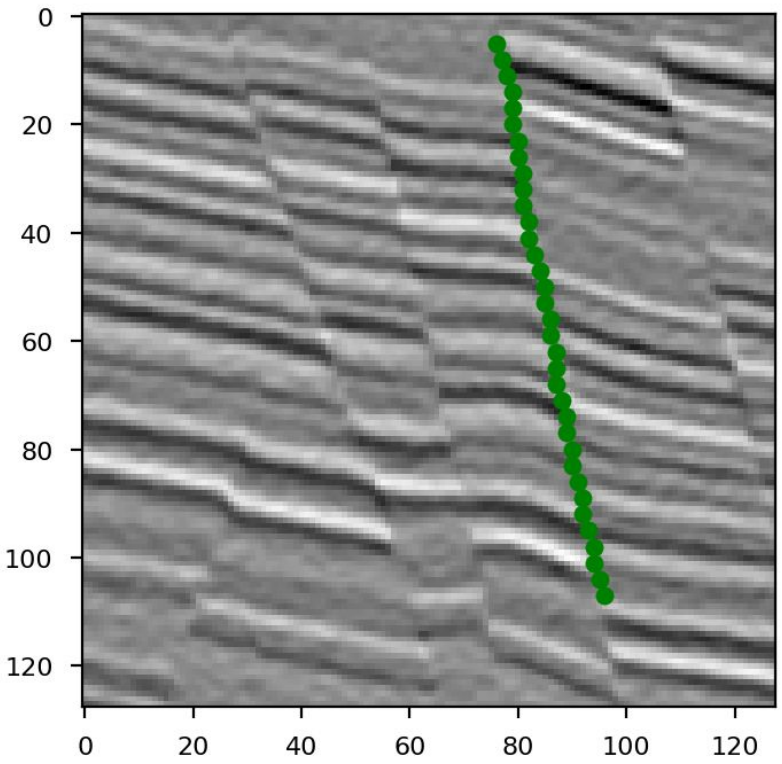
Multi-instances separation



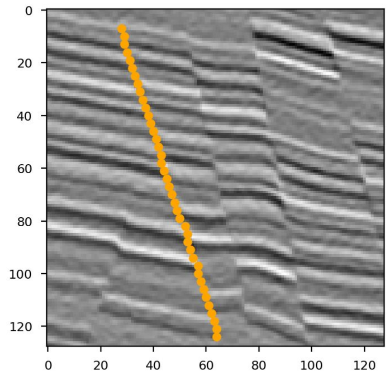


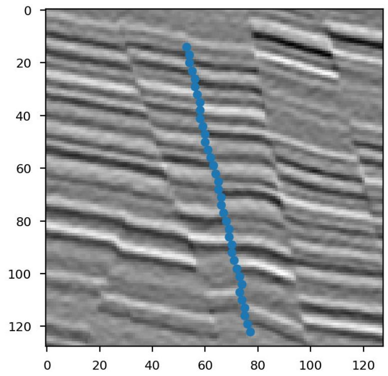
Multi-instances separation

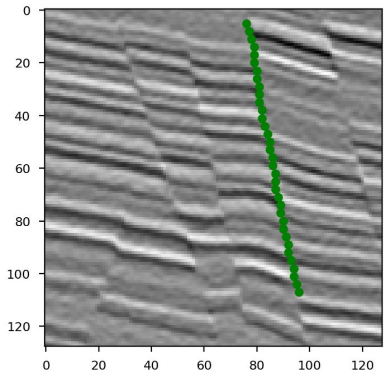


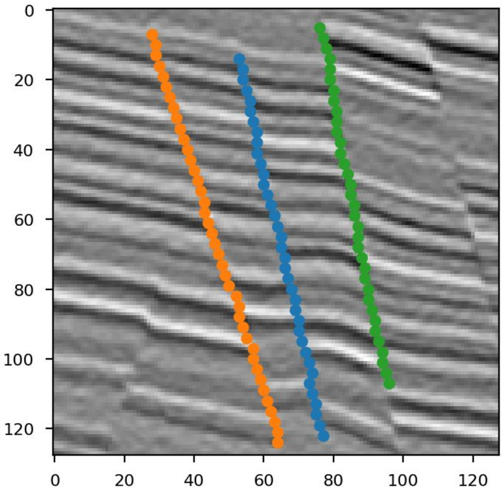


Multi-instances separation





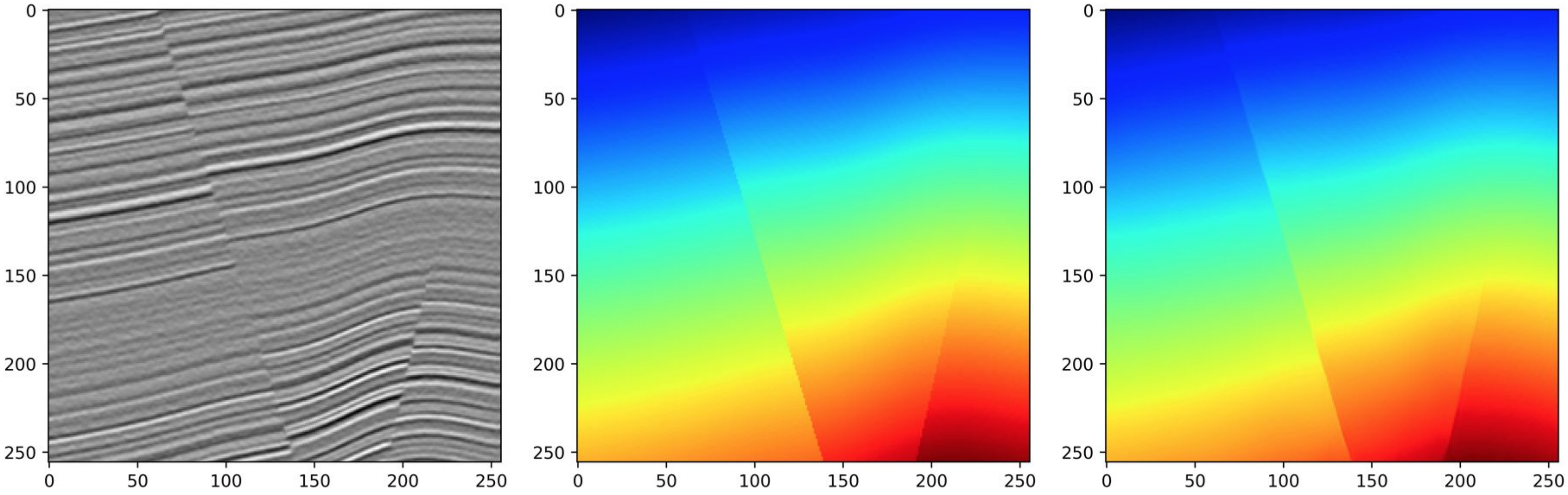




Outline

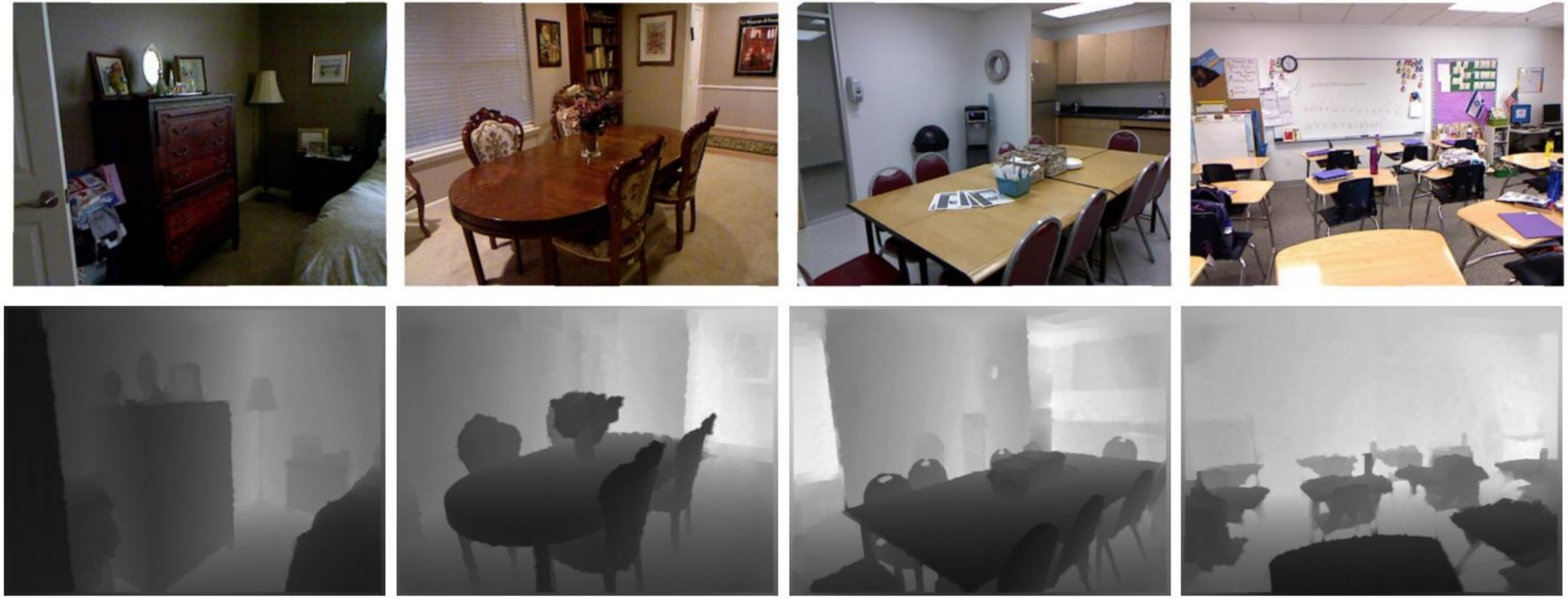
- Simple fault classification
- Generating geophysical synthetic training data
- From classification to segmentation:
  - Fault segmentation
  - Salt body segmentation
  - Channel segmentation
- Tracking geobody in a recurrent style
- **Predicting relative geological time (RGT)**

Predicting relative geological time (RGT)



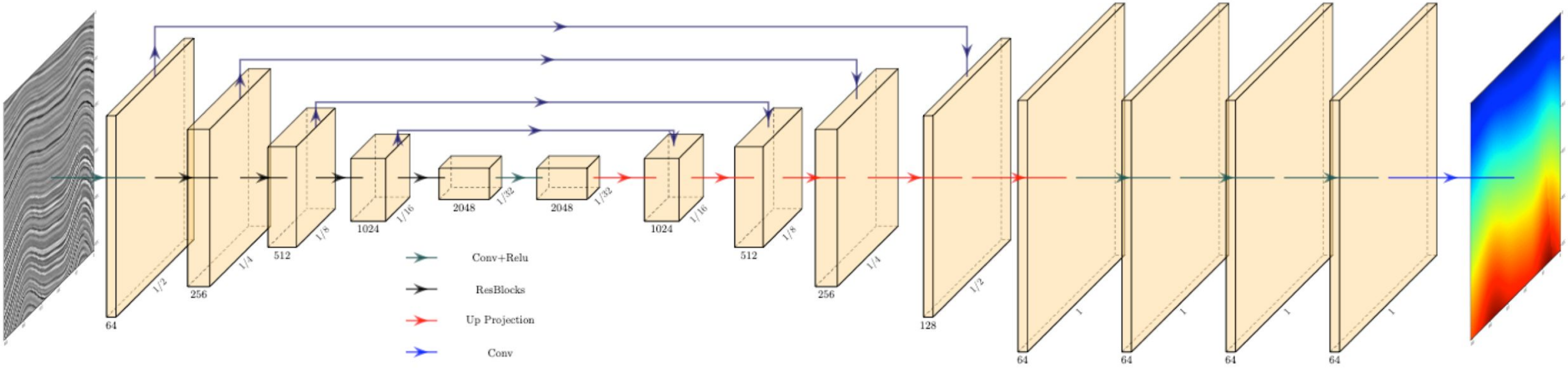
Input seismic image   Ground truth   Estimated RGT

Predicting relative geological time (RGT)

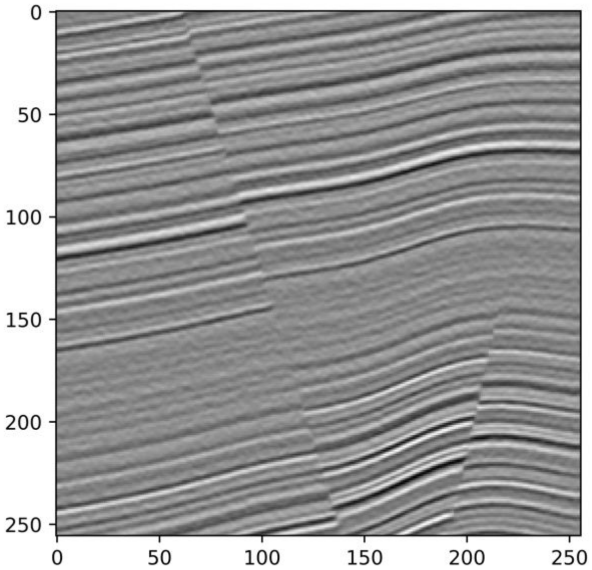


Liu, Fayao, et al., *Deep convolutional neural fields for depth estimation from a single image*, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015.

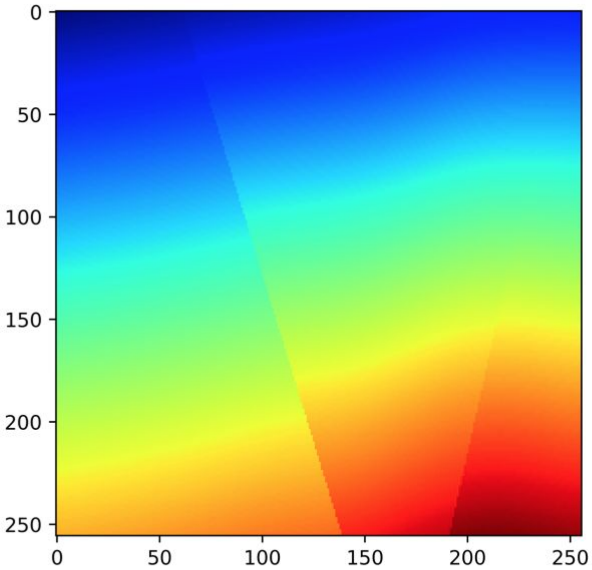
Predicting relative geological time (RGT)



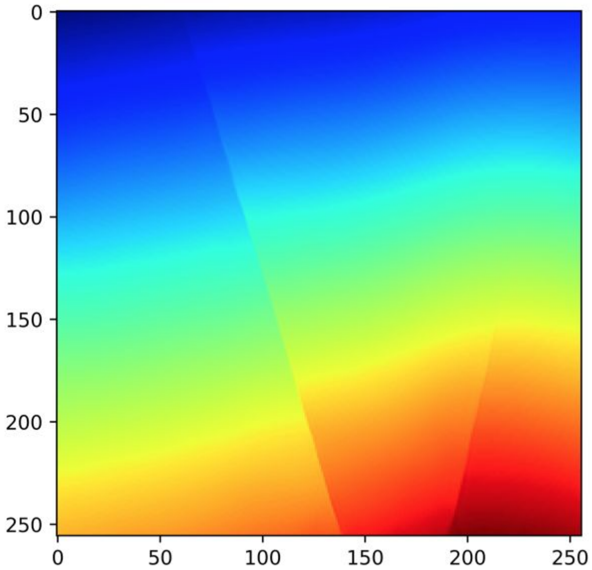
Predicting relative geological time (RGT)



Input seismic image

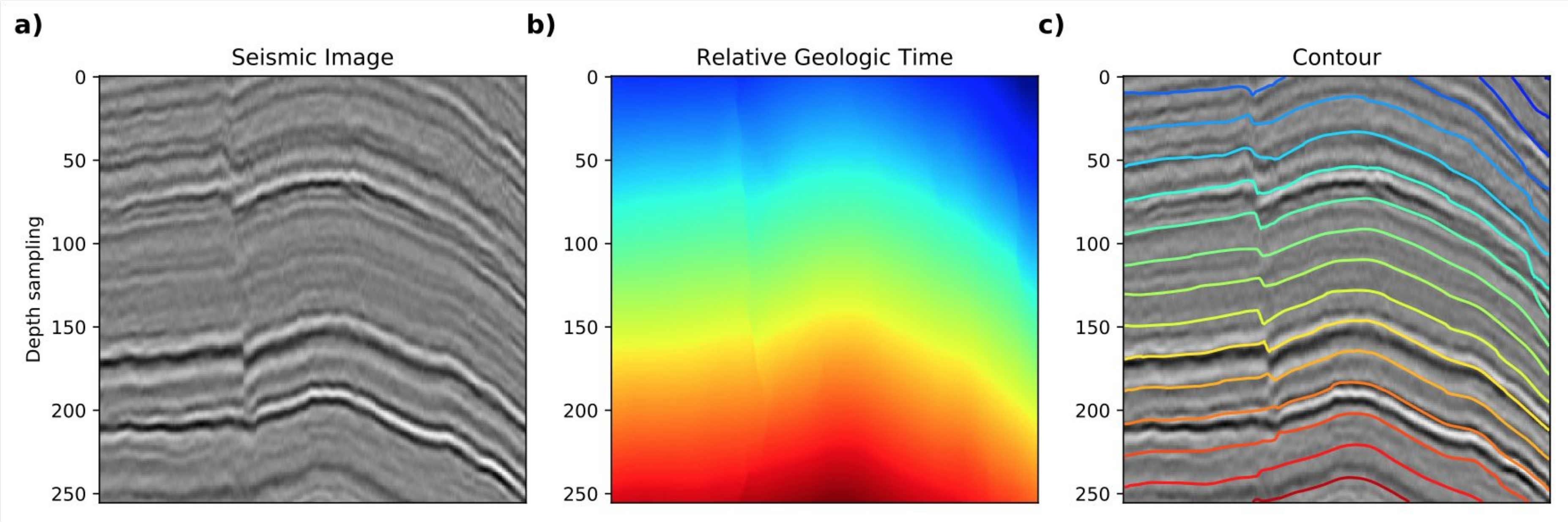


Ground truth

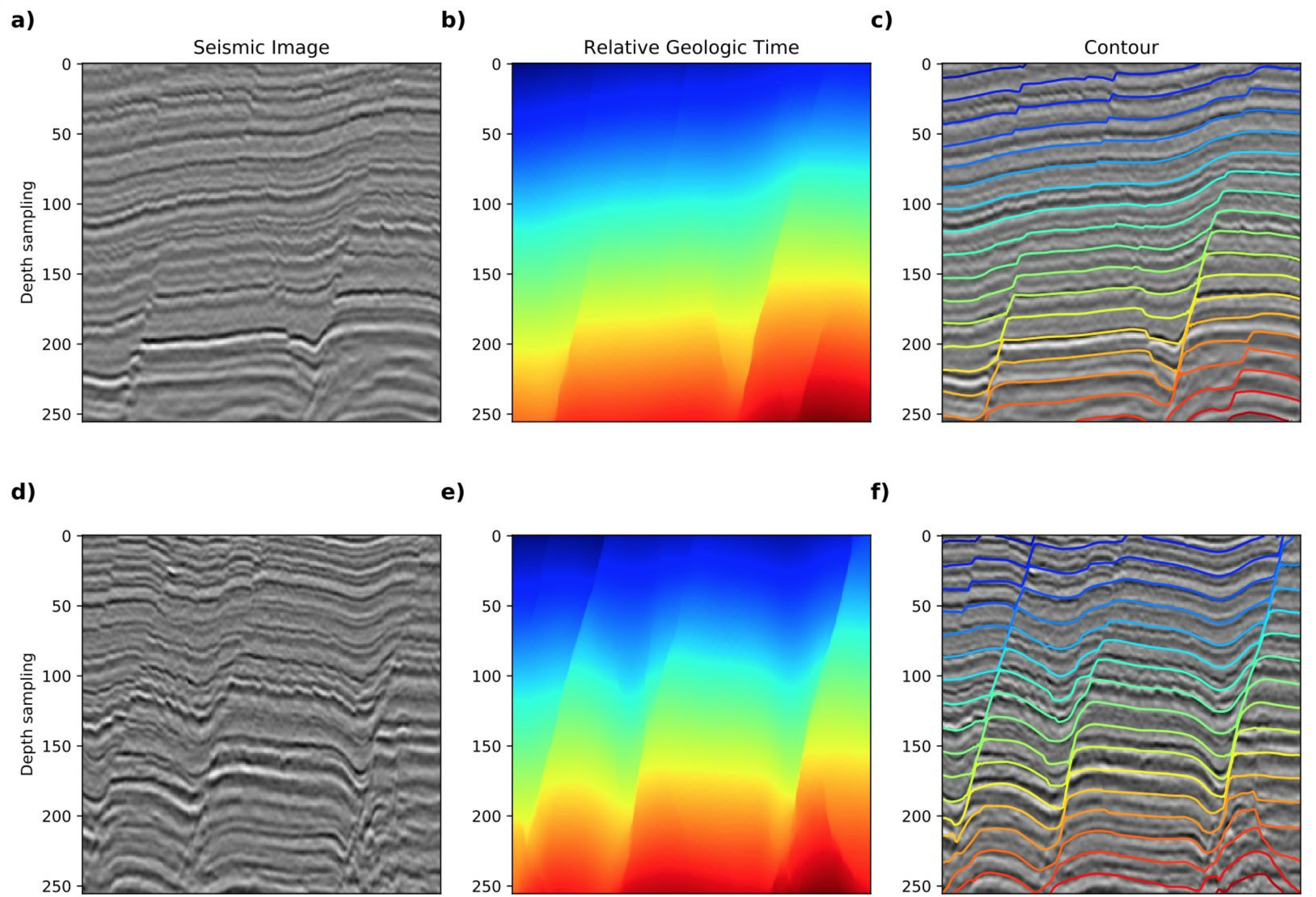


Estimated RGT

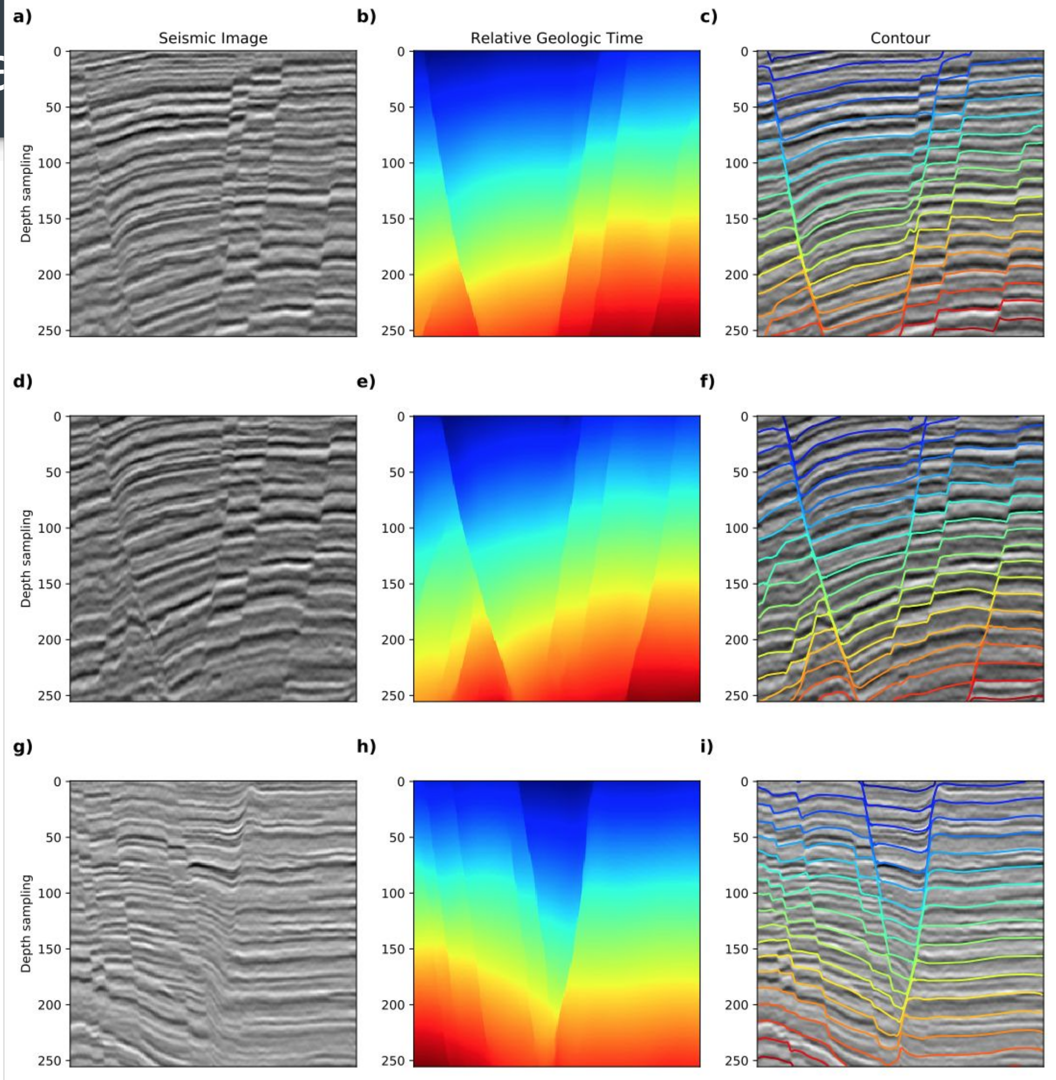
Predicting relative geological time (RGT)



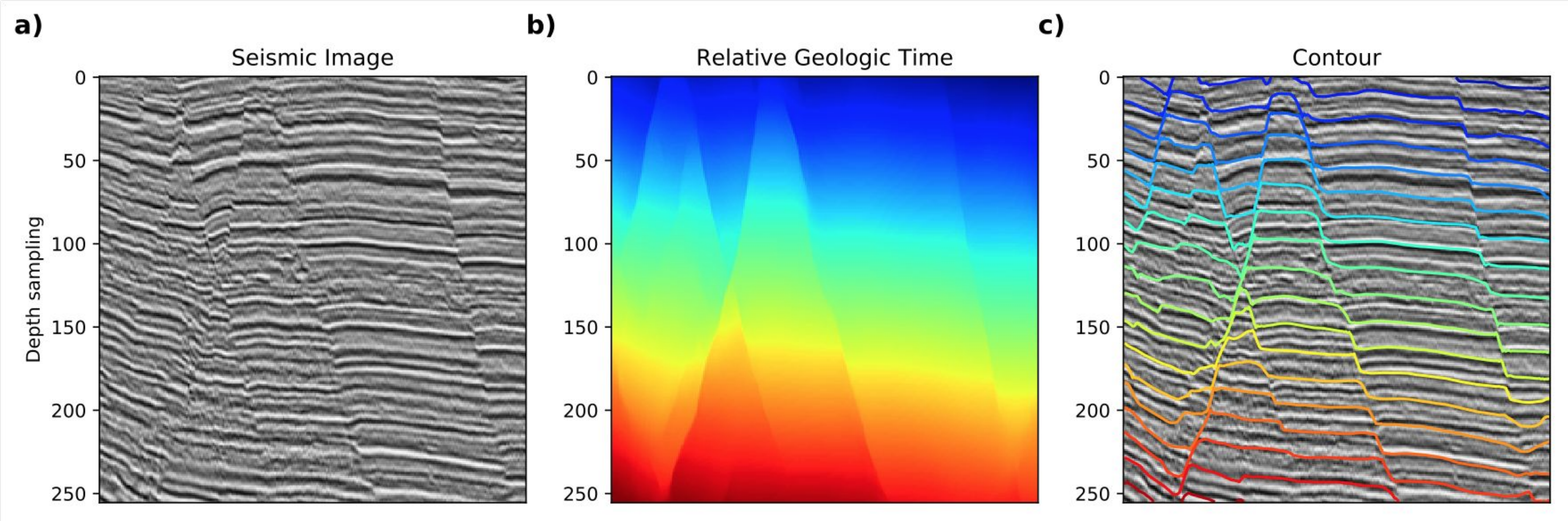
Predicting relative geological time (RGT)



Predicting rela

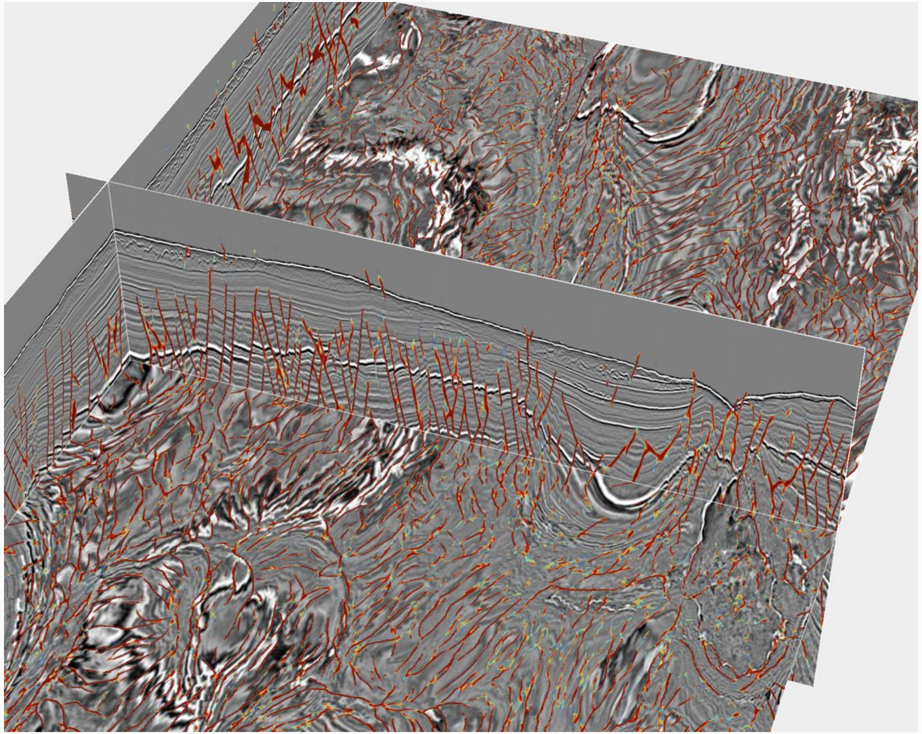


Predicting relative geological time (RGT)

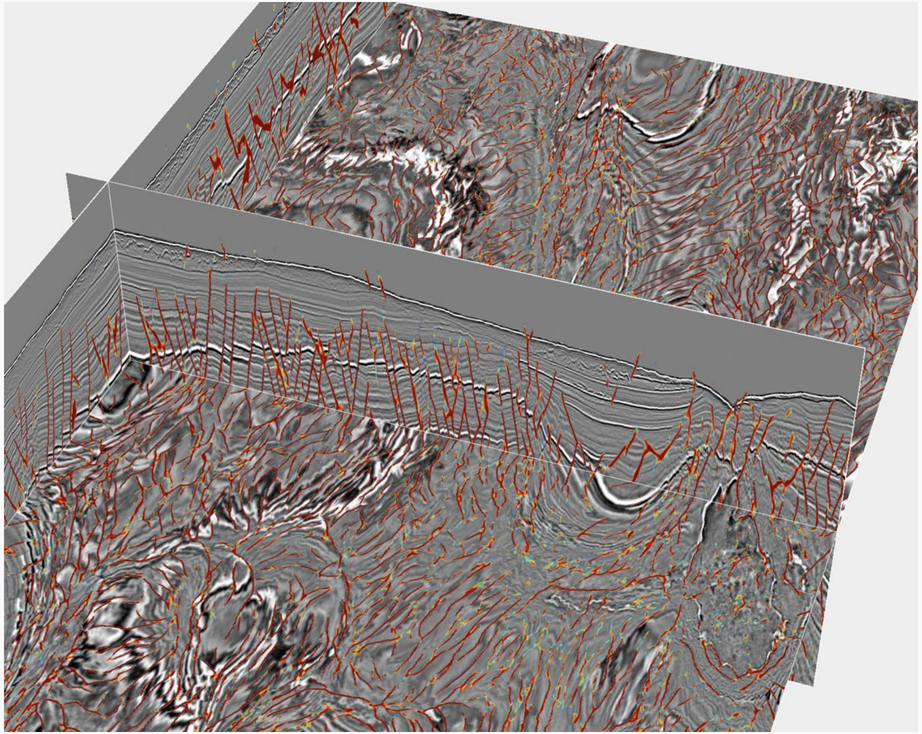


Conclusions

- We discuss what are the new things deep learning can bring into seismic interpretation.
- We show our development of the workflow with this tool designed for seismic images
- **Many can be achieved by using synthetic data already!**



- CNN model can learn from synthetic data
- CNN model can work hard over night
- CNN model can get more experienced with human feedback
- CNN model only performs well on specific tasks
- Can a CNN model follow the logic like a geologist?
- Can a CNN model learn to tell stories from a seismic image?



References

- Wu, Xinming, et al. "Convolutional neural networks for fault interpretation in seismic images." SEG Technical Program Expanded Abstracts 2018.
- Shi, Yunzhi, et al. "Automatic salt-body classification using a deep convolutional neural network." SEG Technical Program Expanded Abstracts 2018.
- Pham, Nam, et al. "Automatic channel detection using deep learning." SEG Technical Program Expanded Abstracts 2018.
- Wu, Xinming, et al. "FaultSeg3D: using synthetic datasets to train an end-to-end convolutional neural network for 3D seismic fault segmentation." Geophysics 84.3 (2019): 1-36.

Acknowledgements

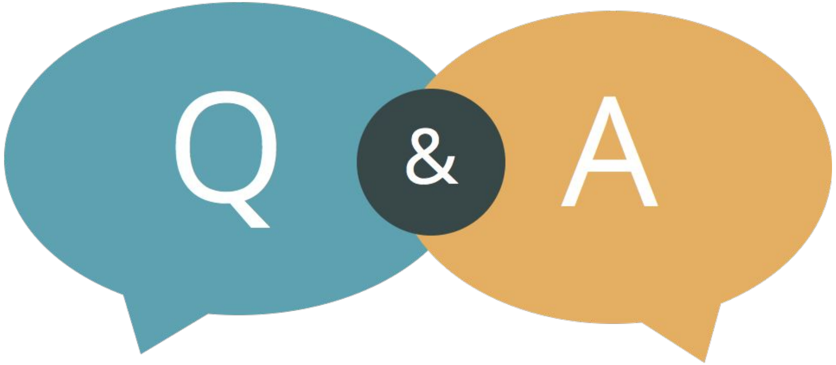
TCCS
Texas Consortium for Computational Seismology

NVIDIA. DEVELOPER NEWS BLOG FORUMS

RTX GAMEWORKS DESIGNWORKS VRWORKS COMP

Home > ComputeWorks > Academic Collaboration > GPU Grant Program

GPU Grant Program


Q
&
A