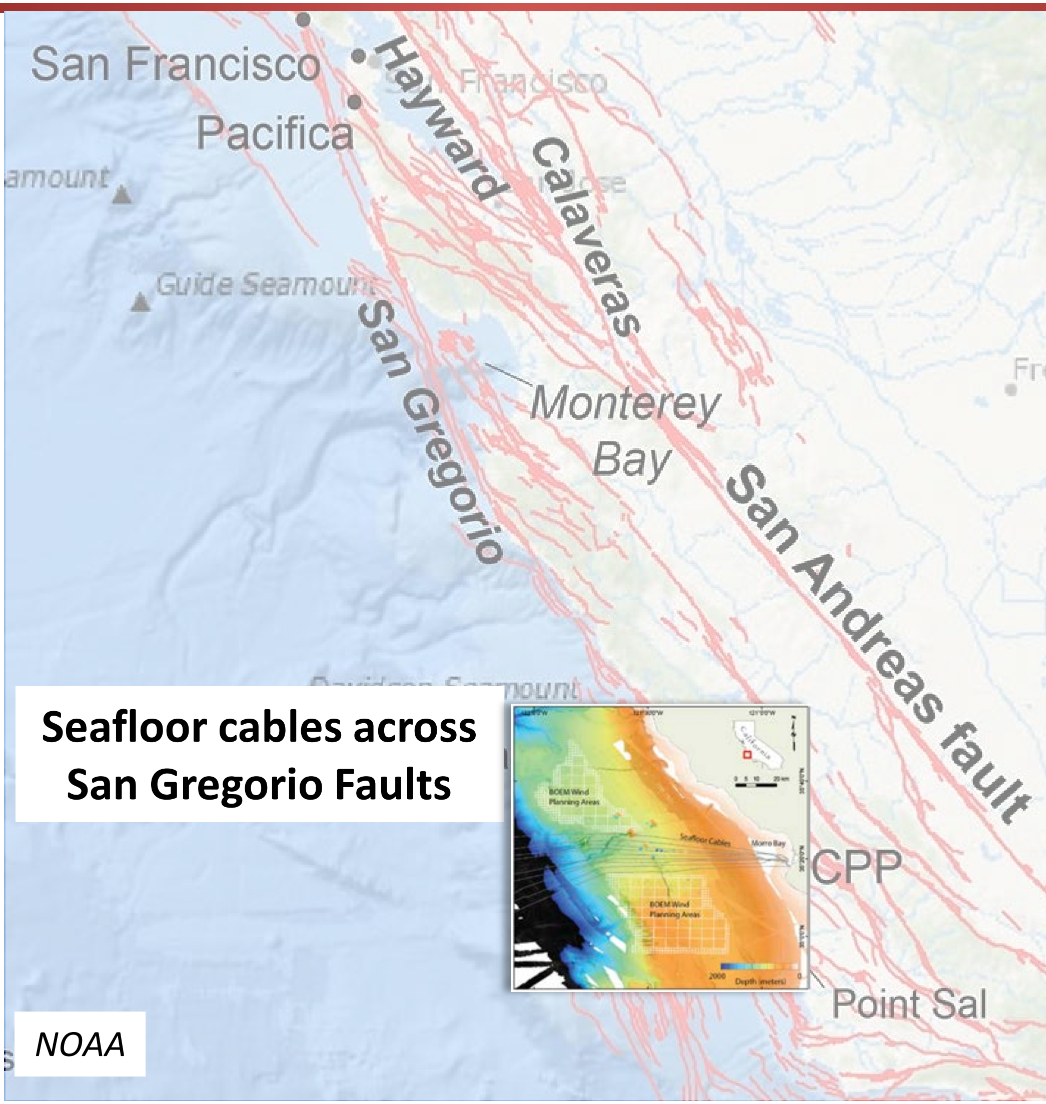
Evaluating Strike-Slip Fault Evolution with laboratory experiments and 2D-CNNs: Identifying the Geohazard Zones?

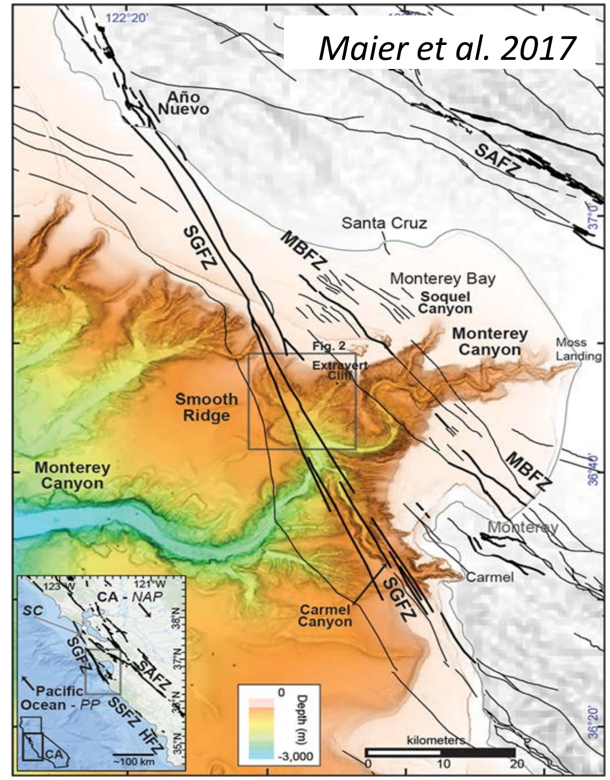
Laainam Chaipornkaew

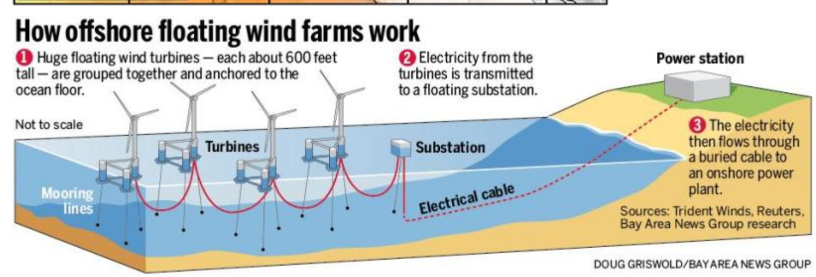
Meeting 2019

Stanford
Rock Physics &
Borehole Geophysics Project

Geohazards Detection







Early detection of **incipient to through-going strike-slip faults**.

- Floating wind farms, offshore floating rigs, etc.

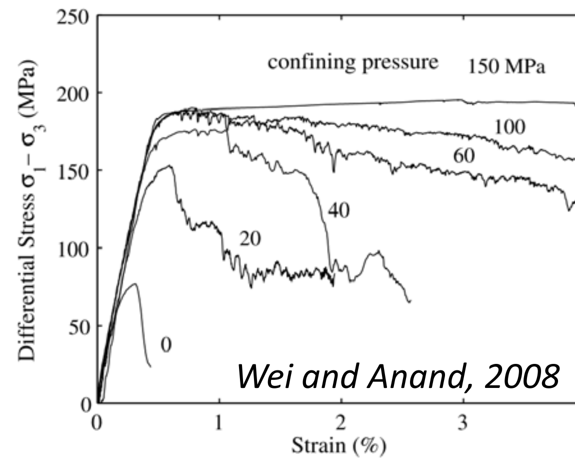
Motivation

Dataset : **Claybox strike-slip experiments**

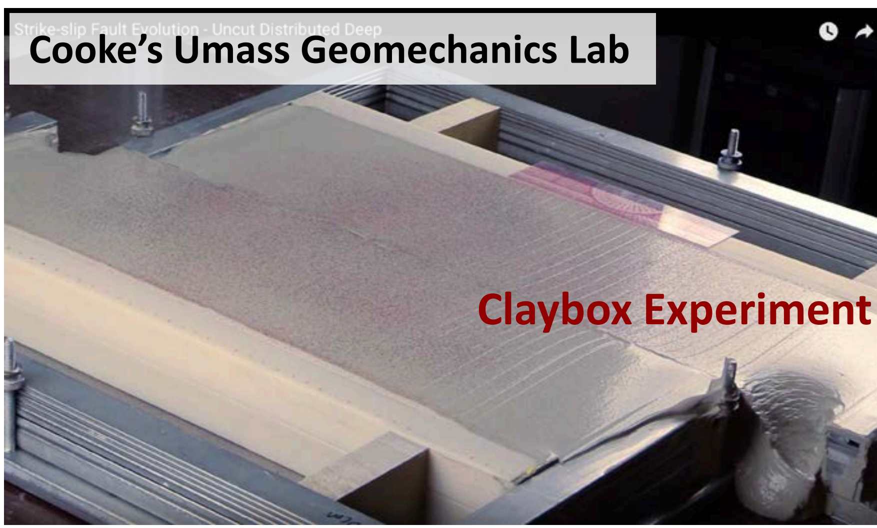
- Experiment: <u>Full range</u> of deformation stages

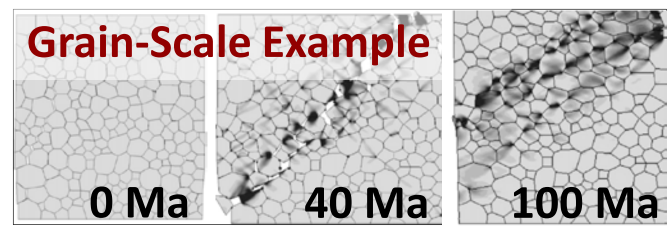
Approach : **Deep learning**

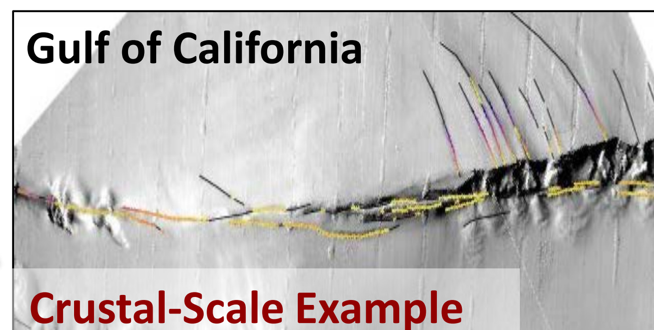
- *Inelastic deformation* behavior is highly non-linear
- Directly predict deformation stages from data without explicitly solve for analytical solution



Application : **Time-lapse displacement data infer mechanical deformation**



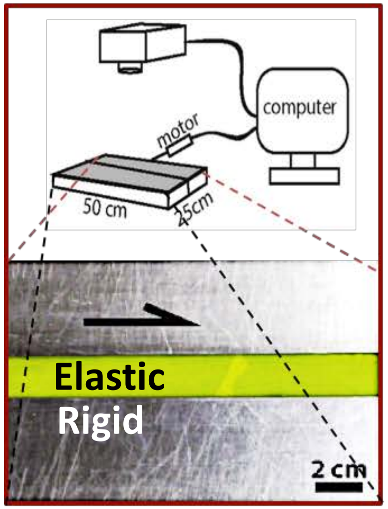




Hilley et al., 2019 (in prep)

UMass Geomechanics – Claybox Experiments

Cooke's Umass Geomechanics Lab



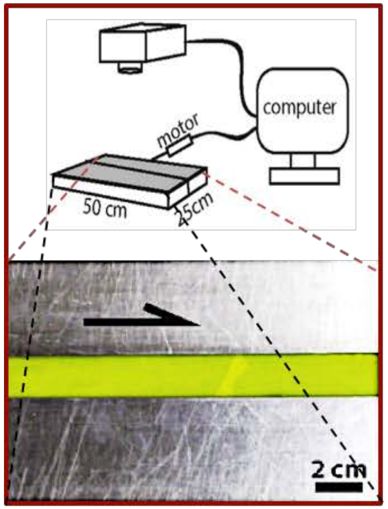
Base Plate

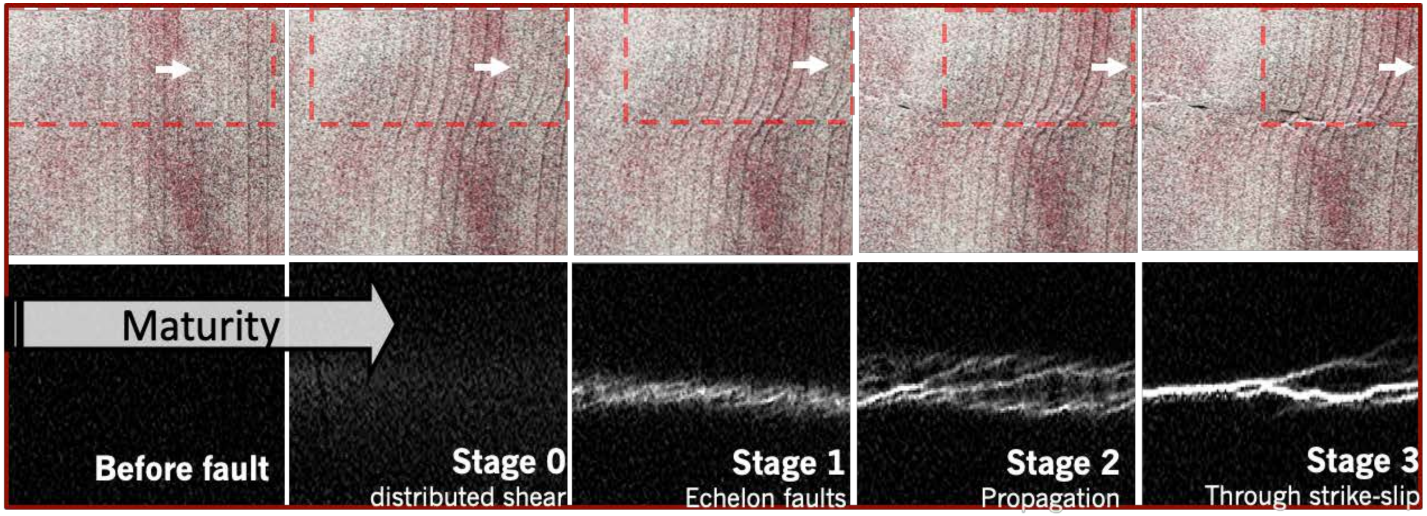
Experiment Variation

Distributed	Deep	Uncut
Localized	Shallow	Cut

Claybox Experiments – full range of deformed stages

Cooke's Umass Geomechanics Lab



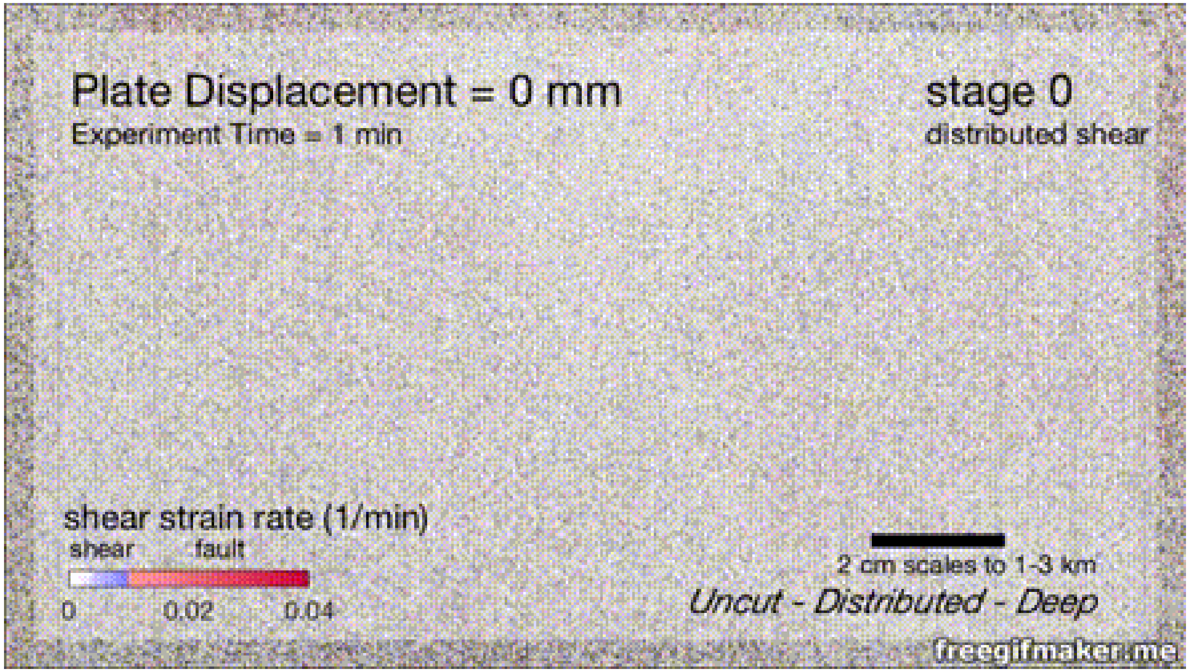


1/3: Stage 0

-- Stage 1 --

1/3: Stage 2

1/3: stage 3



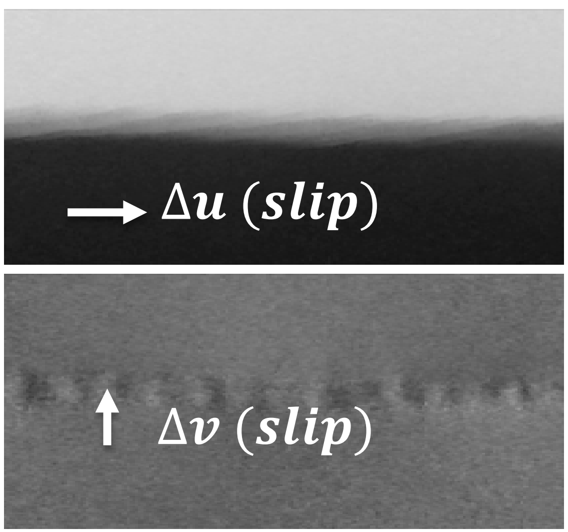
Objective

Goal:

Given a **trained model**, which has already **seen all fault stages** from Claybox experiments

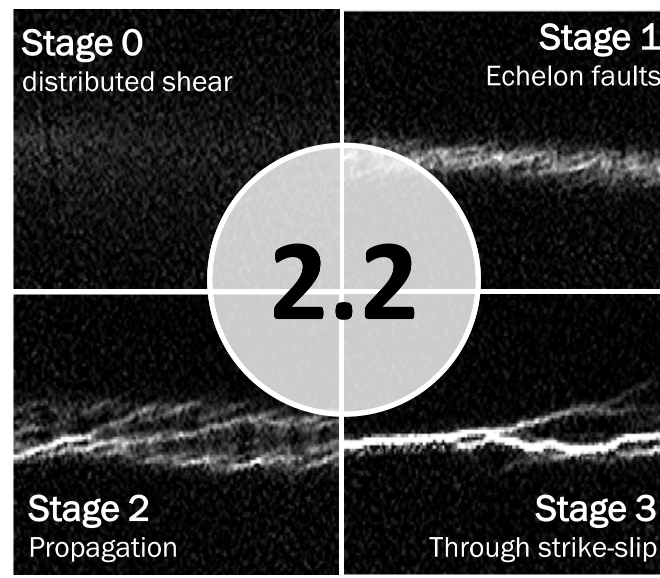
Prediction of fault stage from **a single 'current-day'** surface data.

Input:



Output:

stage prediction (1digit float)



Workflow

Data Pipeline

- Stacked 3 physical values (.mat)
  - Δu, Δv, Strain rate
- Clipped to **128x32** sub-images
- ~7,500 sub-images per experiment
- Train: Dev: Test = **{0.75: 0.15: 0.10}**

Labeling Strategy

- Continuous Label
- Regression + Classification styles

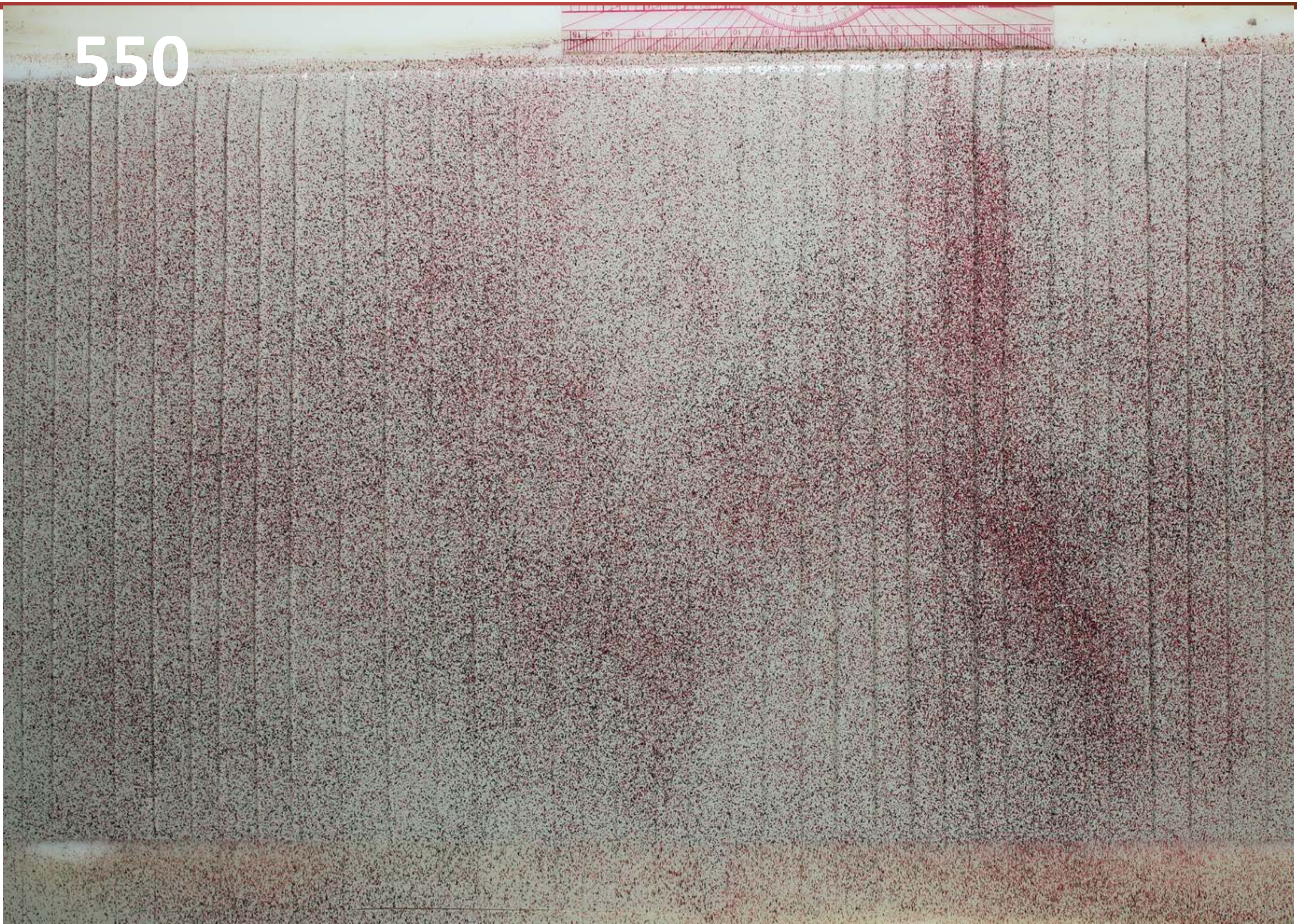
Training

- Custom Loss Function
- Image Augmentation
- Model Architecture and hyperparameter decisions

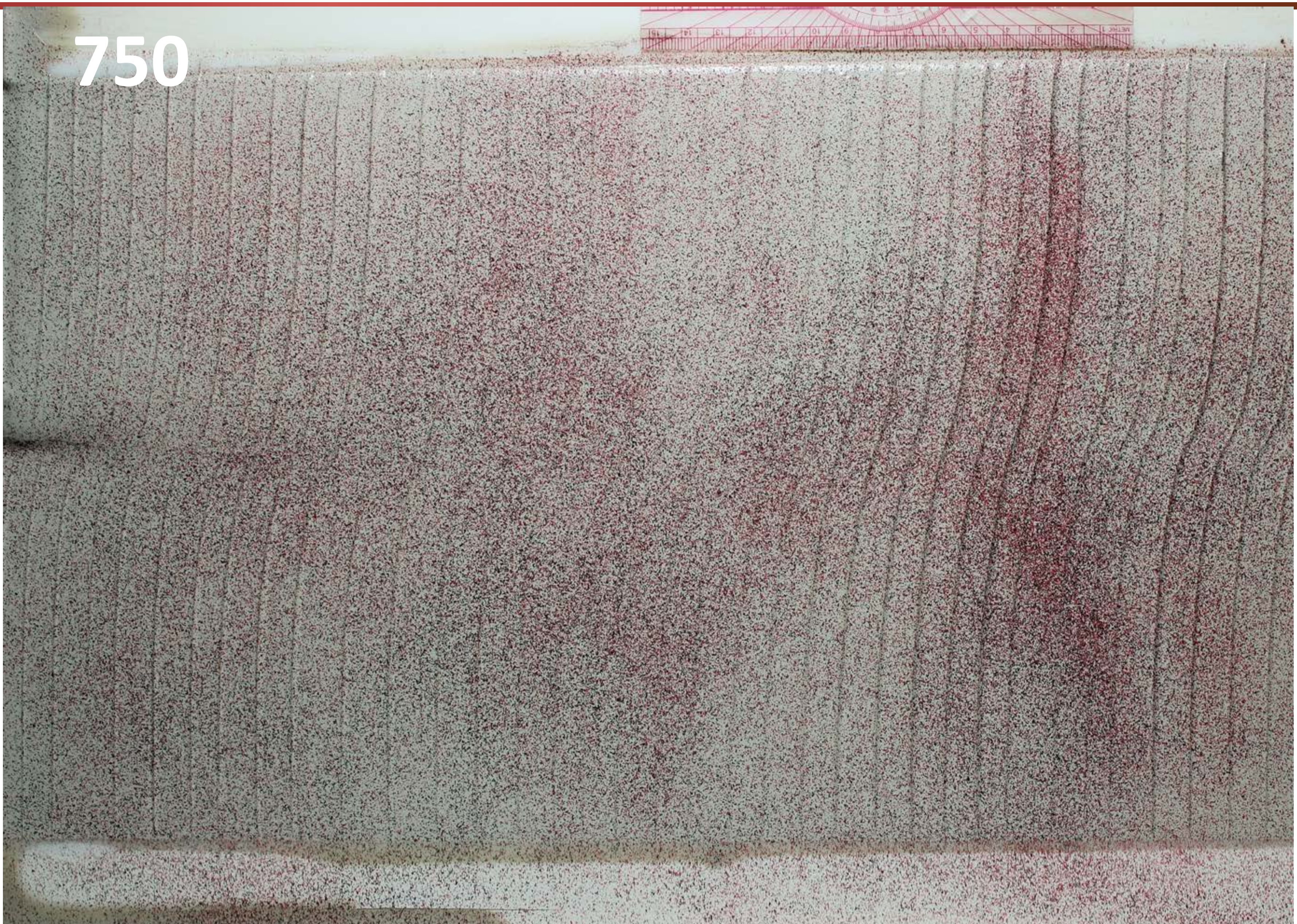
Evaluation

- Custom Loss Metric
- Grad-Cam Visualization

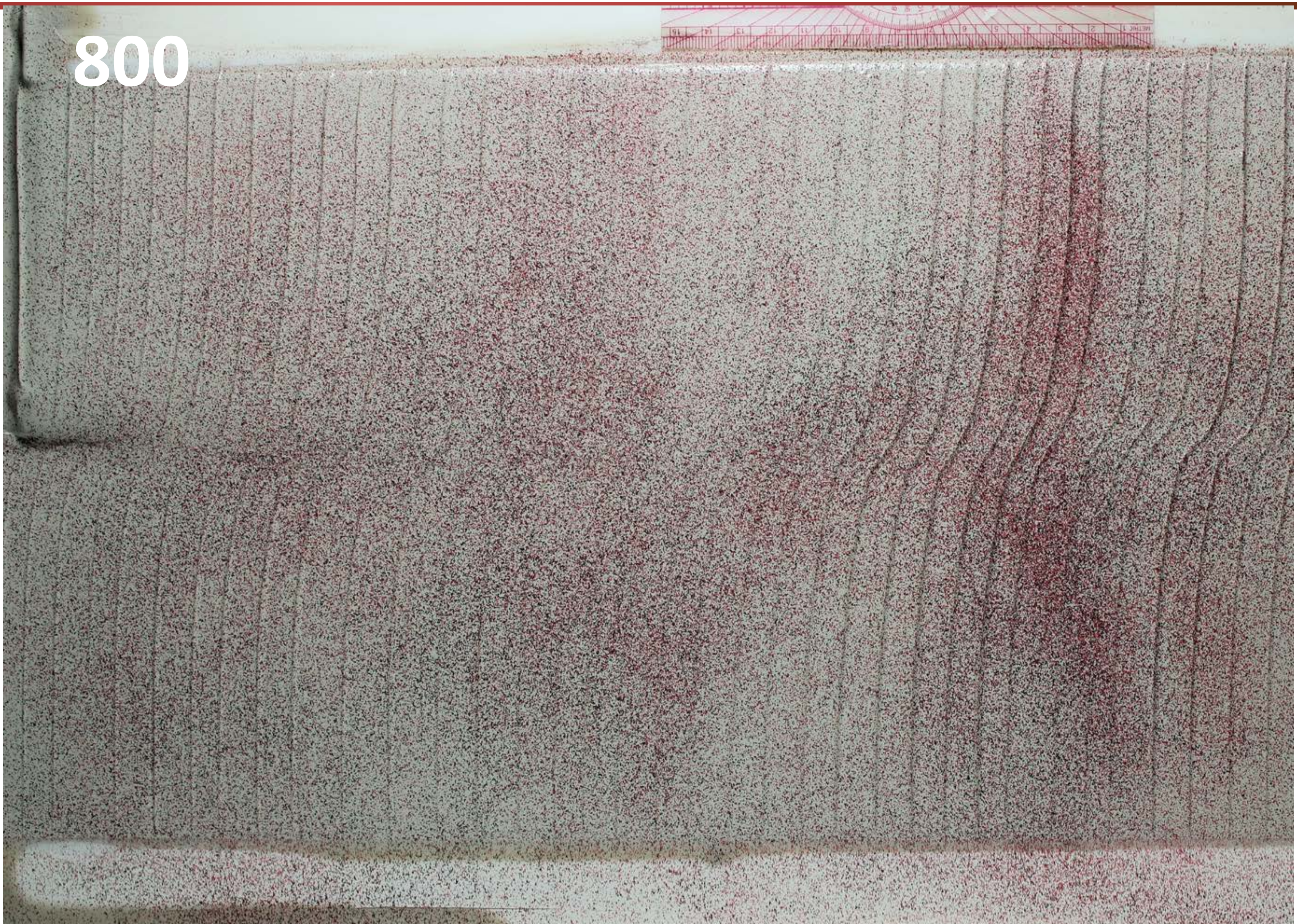
Data Pipeline: Raw Images



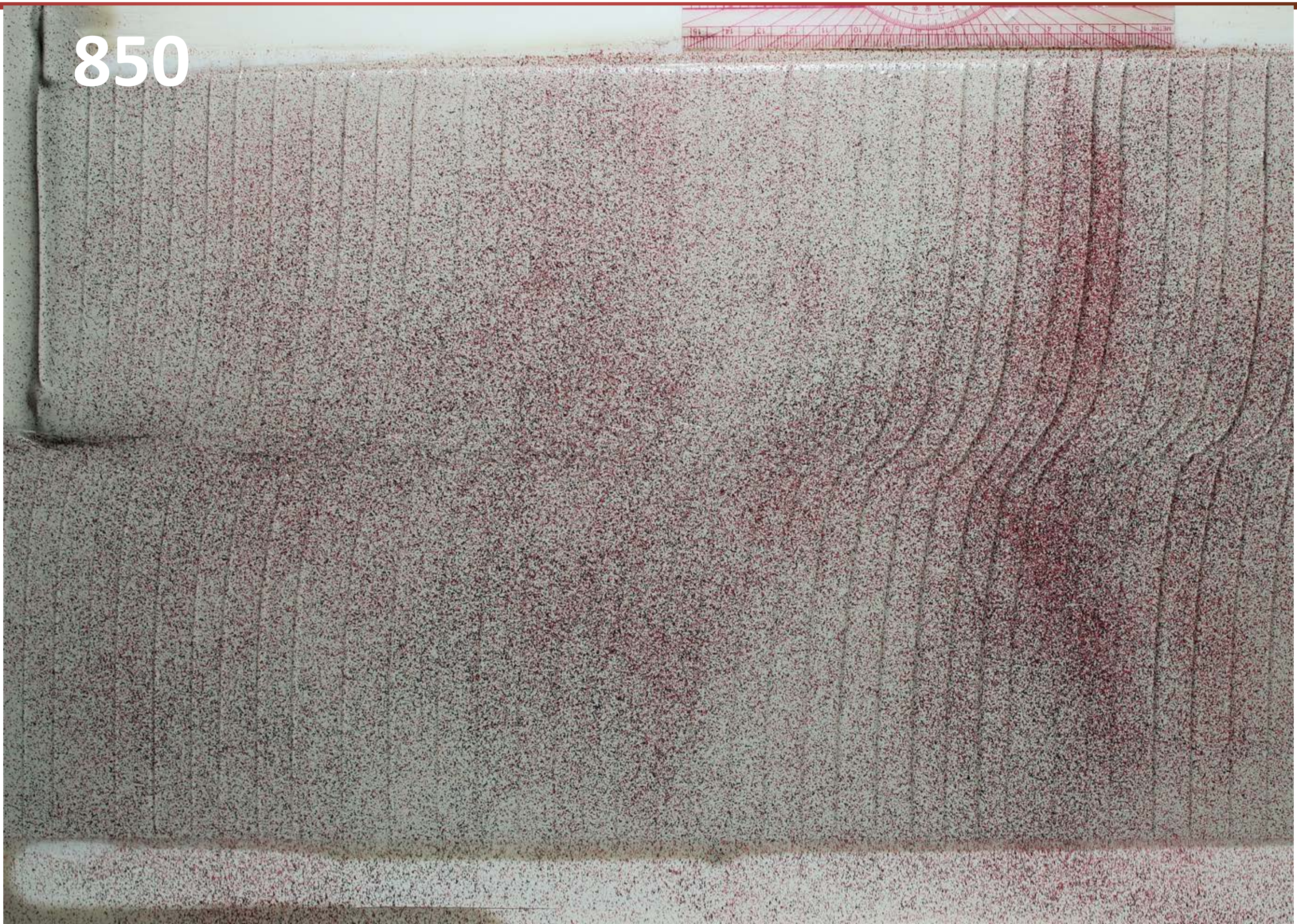
Data Pipeline: Raw Images



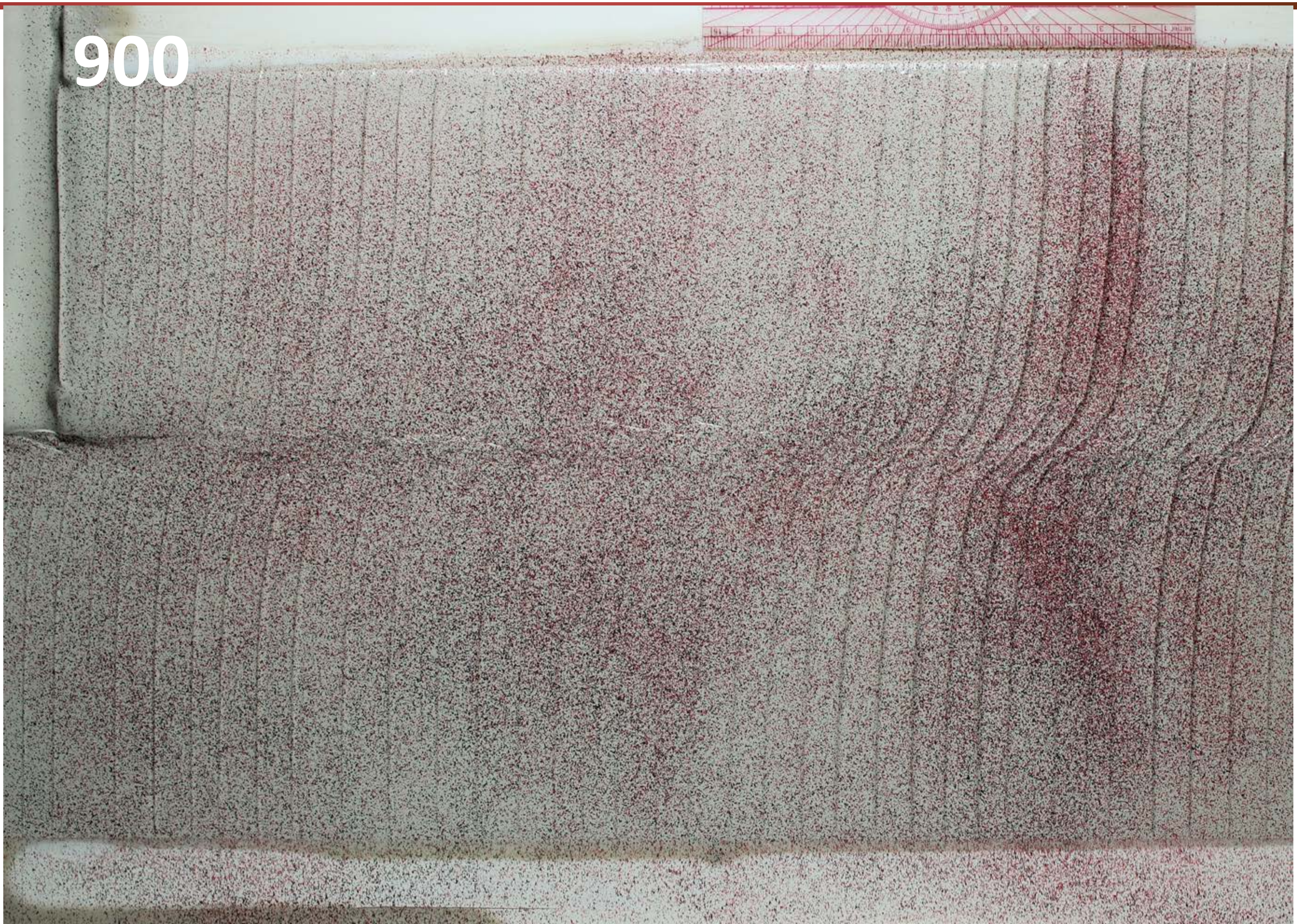
Data Pipeline: Raw Images



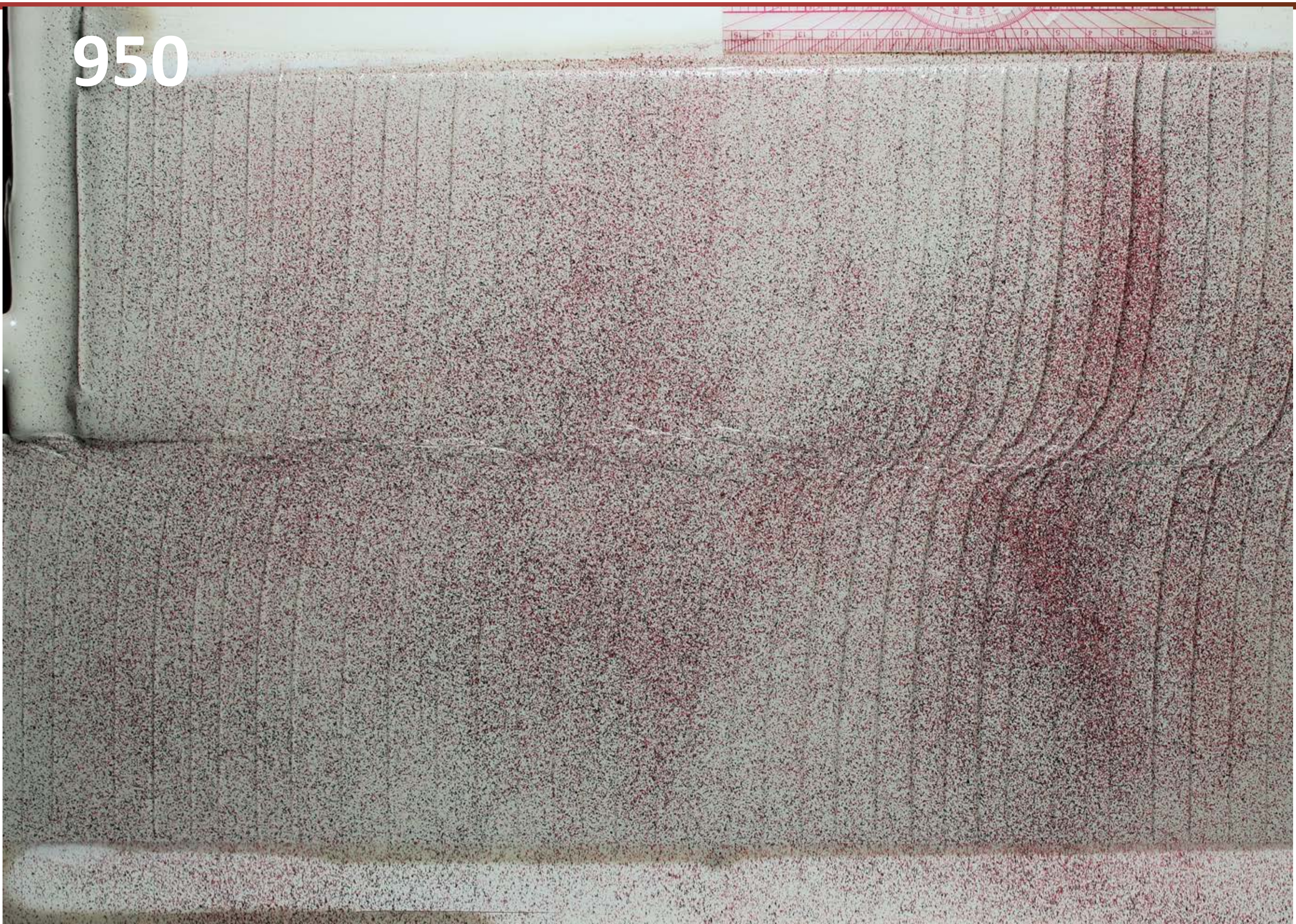
Data Pipeline: Raw Images



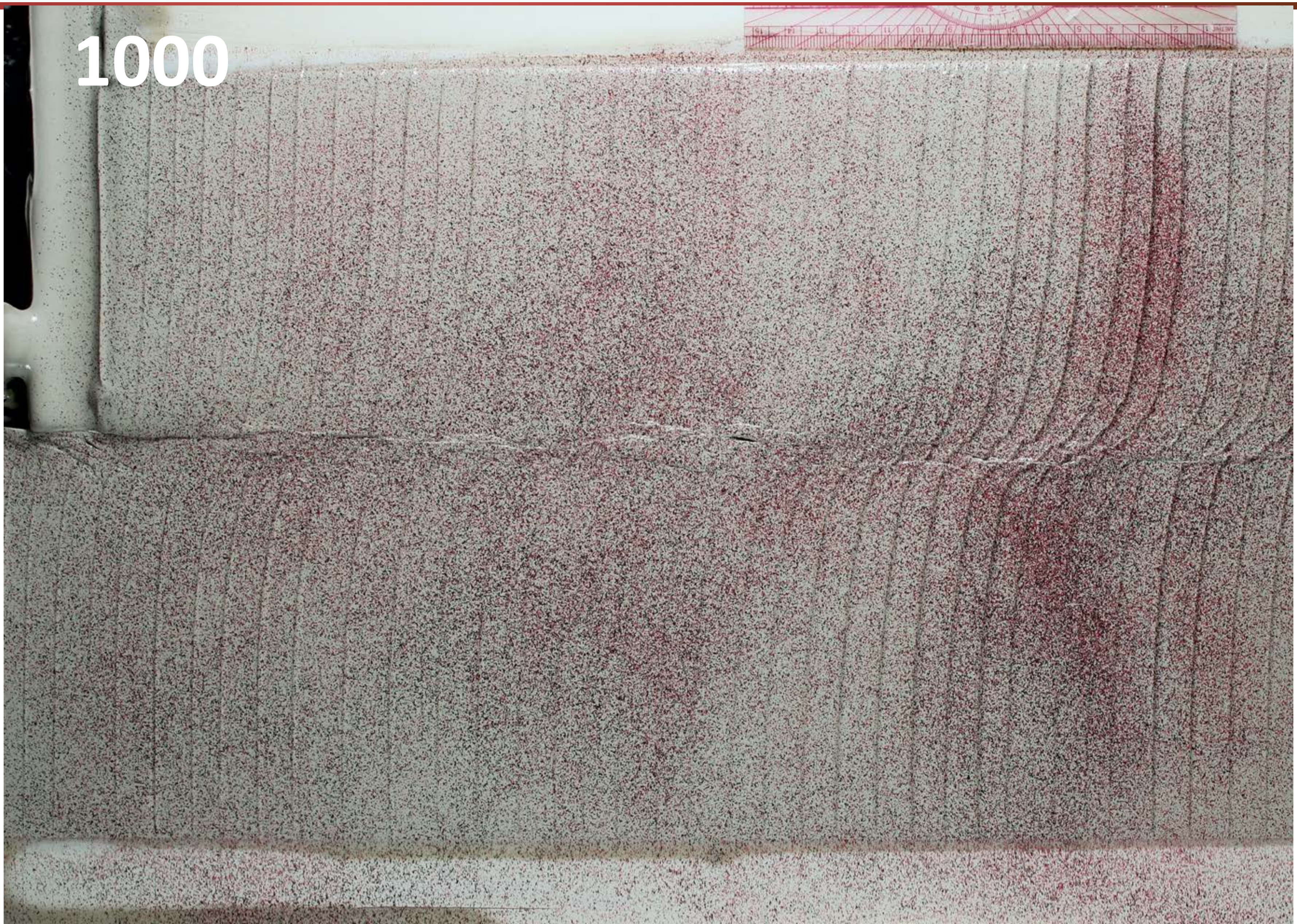
Data Pipeline: Raw Images



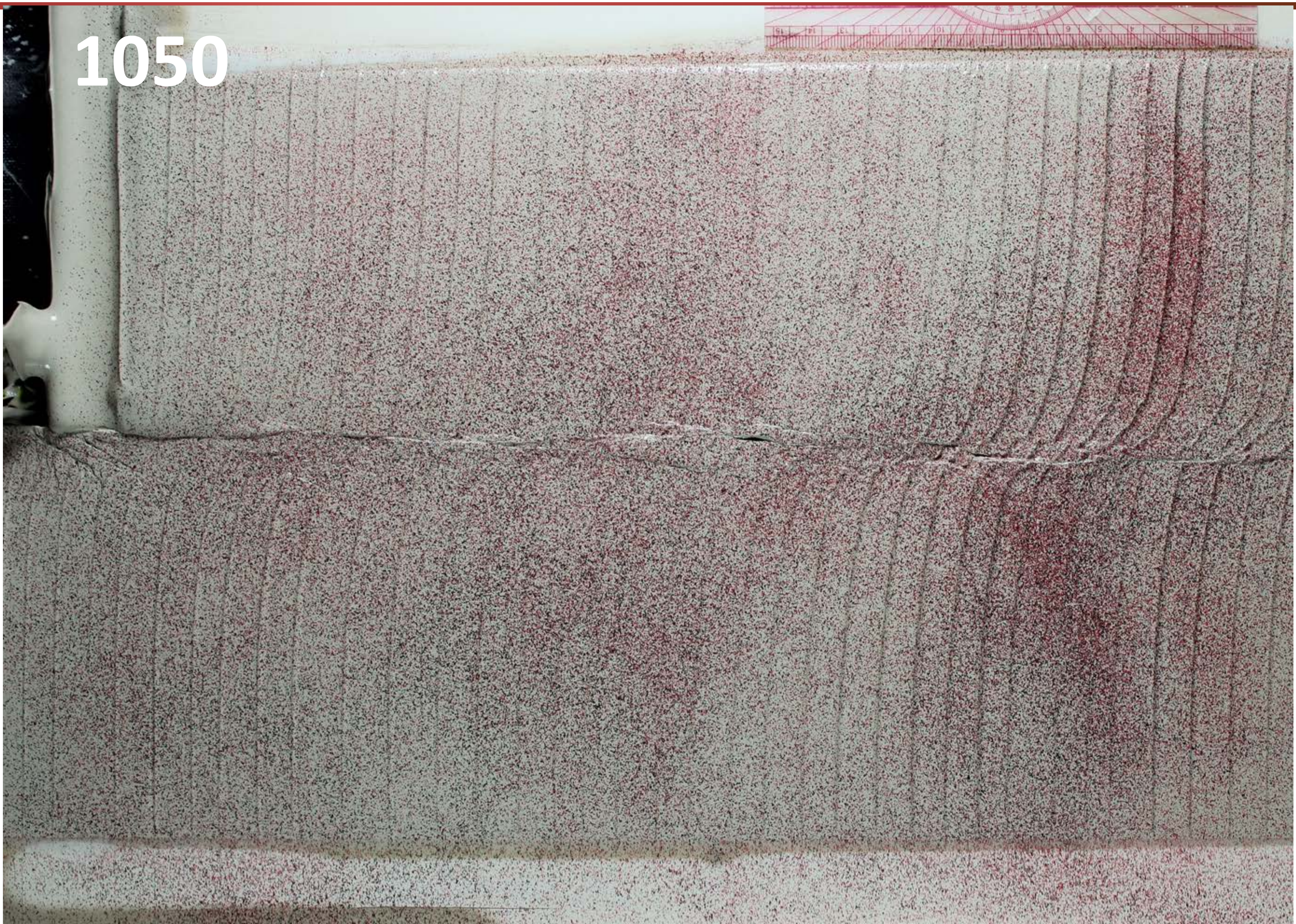
Data Pipeline: Raw Images



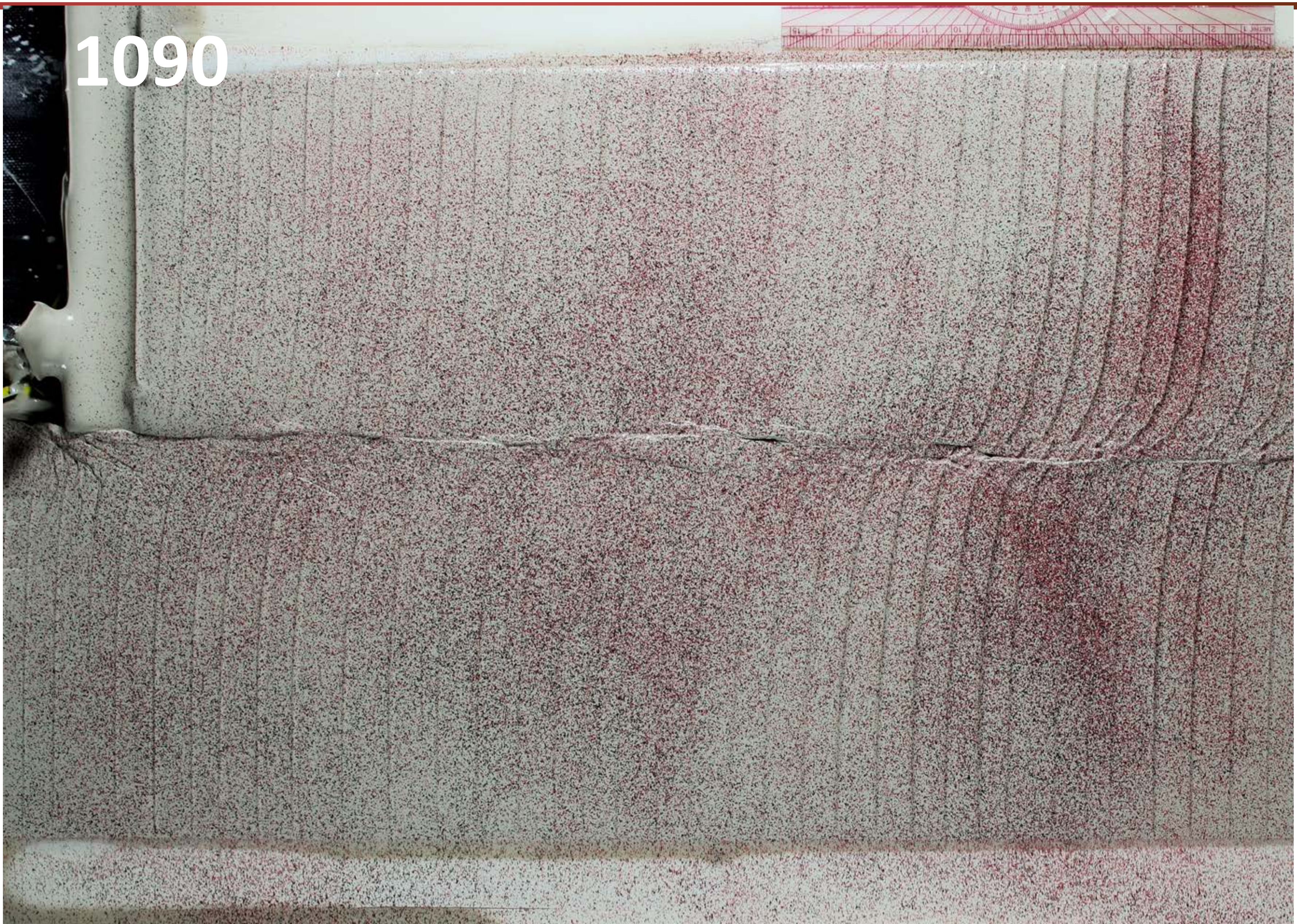
Data Pipeline: Raw Images



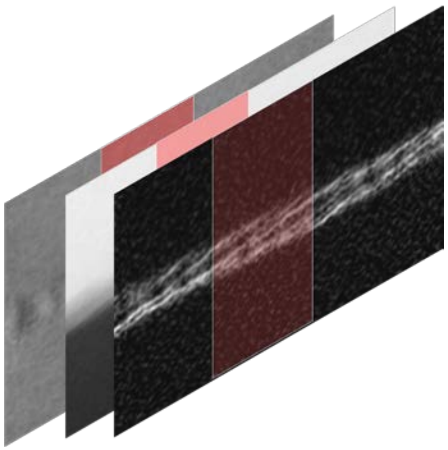
Data Pipeline: Raw Images

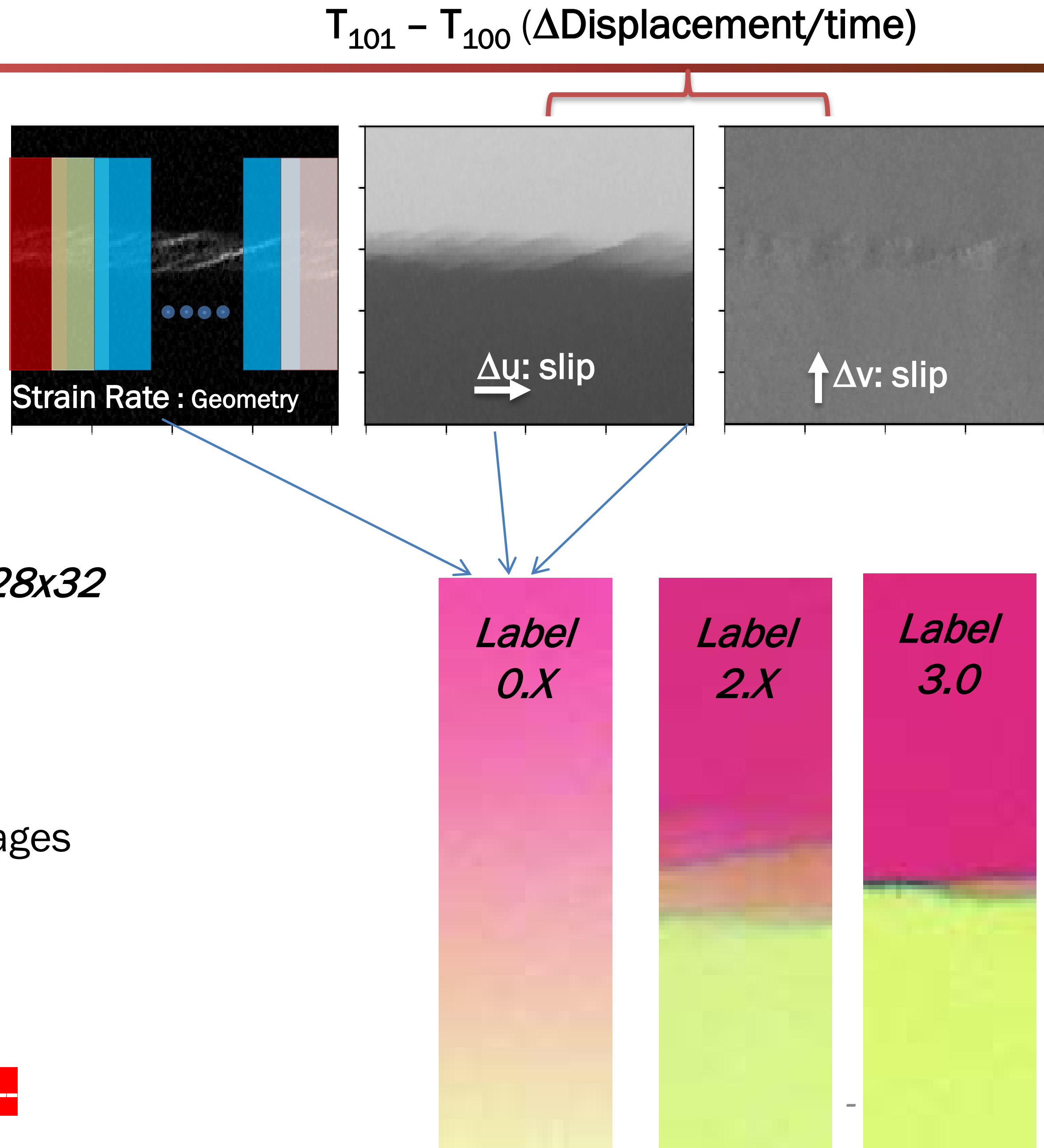


Data Pipeline: Raw Images



Data Pipeline





Stacked input data 128x32 (x3 channels)

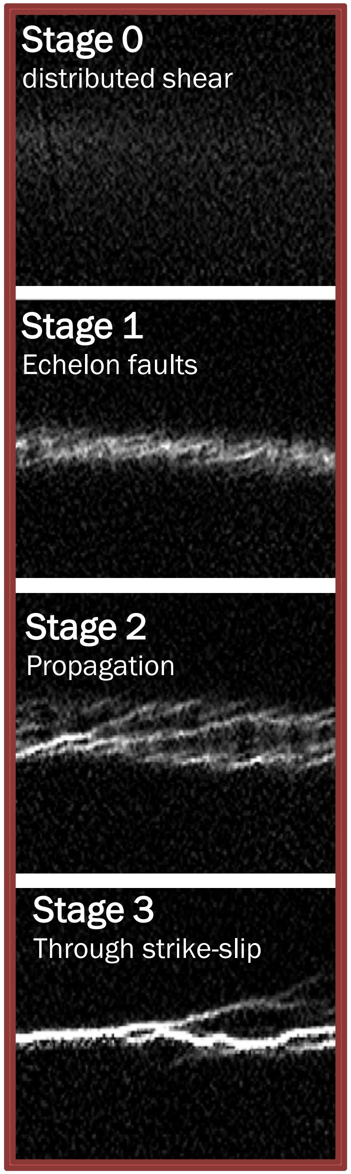
Per Experiment

- 7,500 sub-images
- 2,500 stacked-images

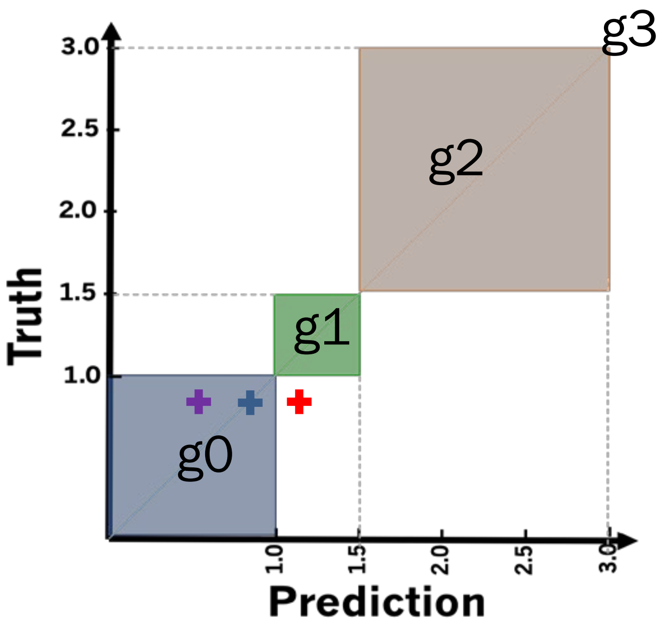
Split {Train: Dev: Test}
{0.75: 0.15: 0.10}

--- Pre-Augmentation ---

Labeling Strategy



Custom 'Bracket' Loss Function and 'Bracket' Accuracy



$$\mathcal{L} = mse + (g0)^2 + (g1)^2 + (g2)^2 + g3$$

$y<1$: $g0 = \max(0, \hat{y} - 1)$

$1 \leq y < 1.5$: $g1 = \max(\max(1 - \hat{y}, 0), \max(0, \hat{y} - 1.5))$

$1.5 \leq y < 3.0$: $g2 = \max(\max(1.5 - \hat{y}, 0), \max(0, \hat{y} - 3.0))$

$y = 3$: $g3 = \max(0, 3 - \hat{y})$

- **MSE** is a reasonable metrics for regression problem
- '**Bracket Loss**' uses MSE with extra penalization to predictions that fall outside their characteristic groups (g0, g1,g2, g3)
- Co-efficient terms / order of magnitude are tuned during training.

Bracket Accuracy :

- **Correct** if prediction is in characteristic group as ground truth
- **Incorrect** otherwise

Model Architecture

- Both shallow and deeper CNNs are explored.

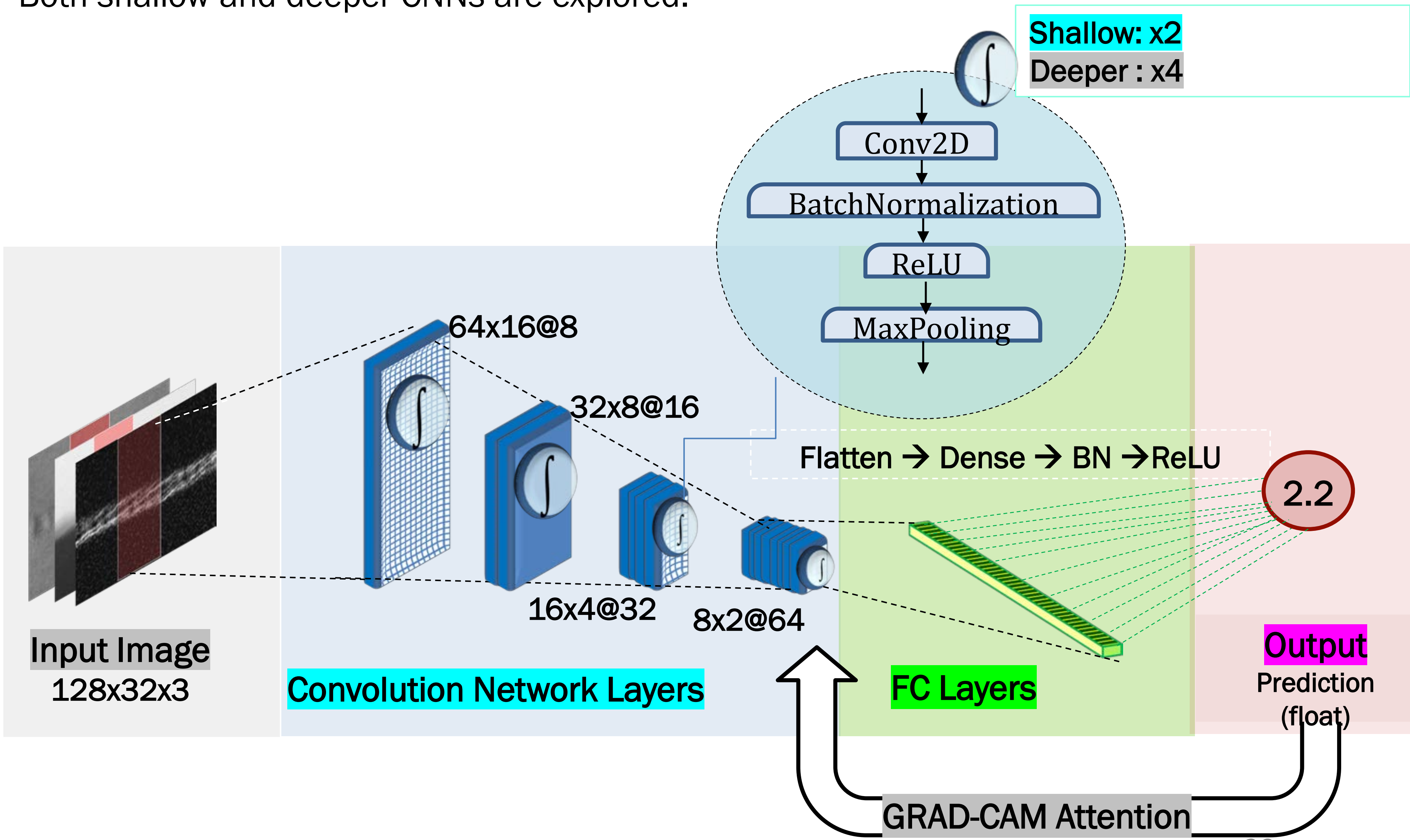
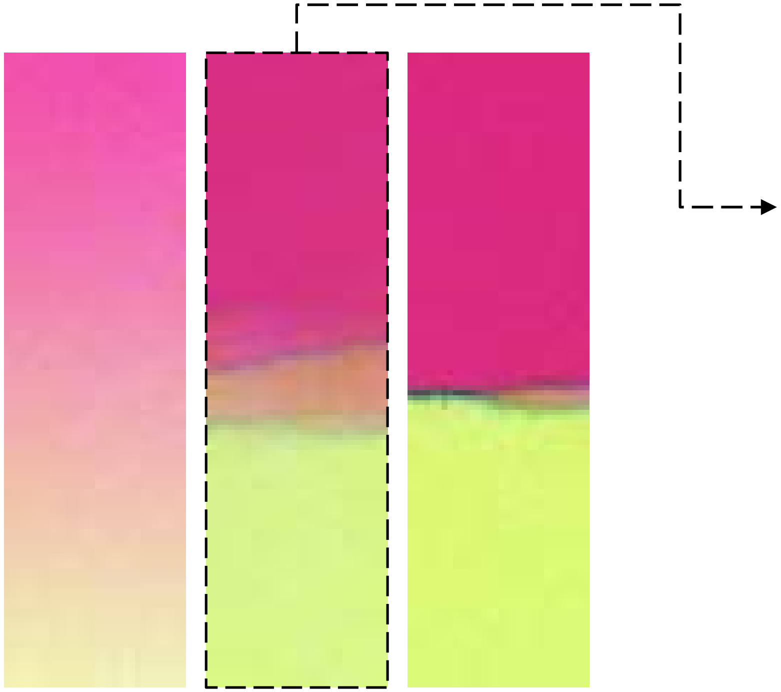
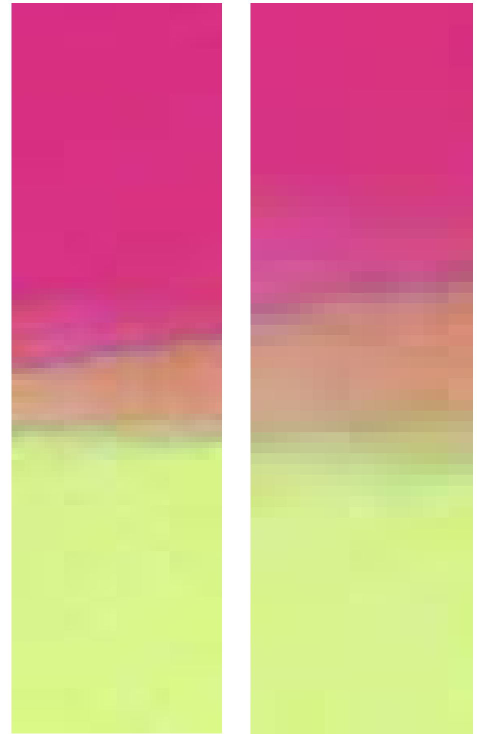


Image Augmentation force CNNs to ‘see’ faults

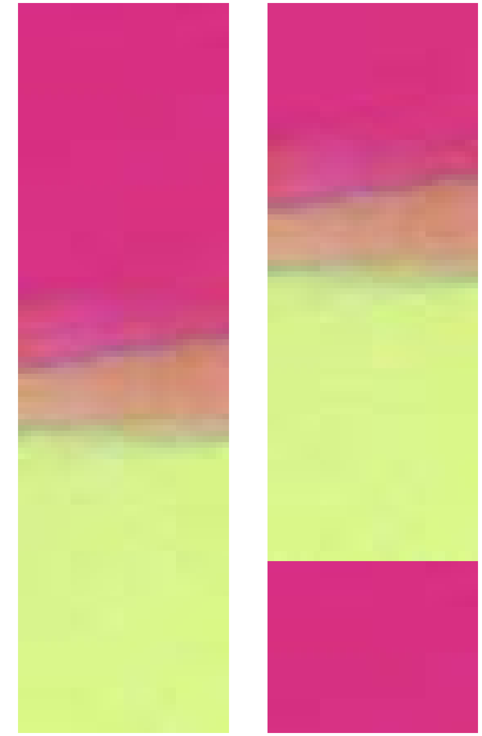


Randomly applied:

- Horizontal Flip
- Vertical Flip
- Brightness
- ** Shift
- ** Zoom



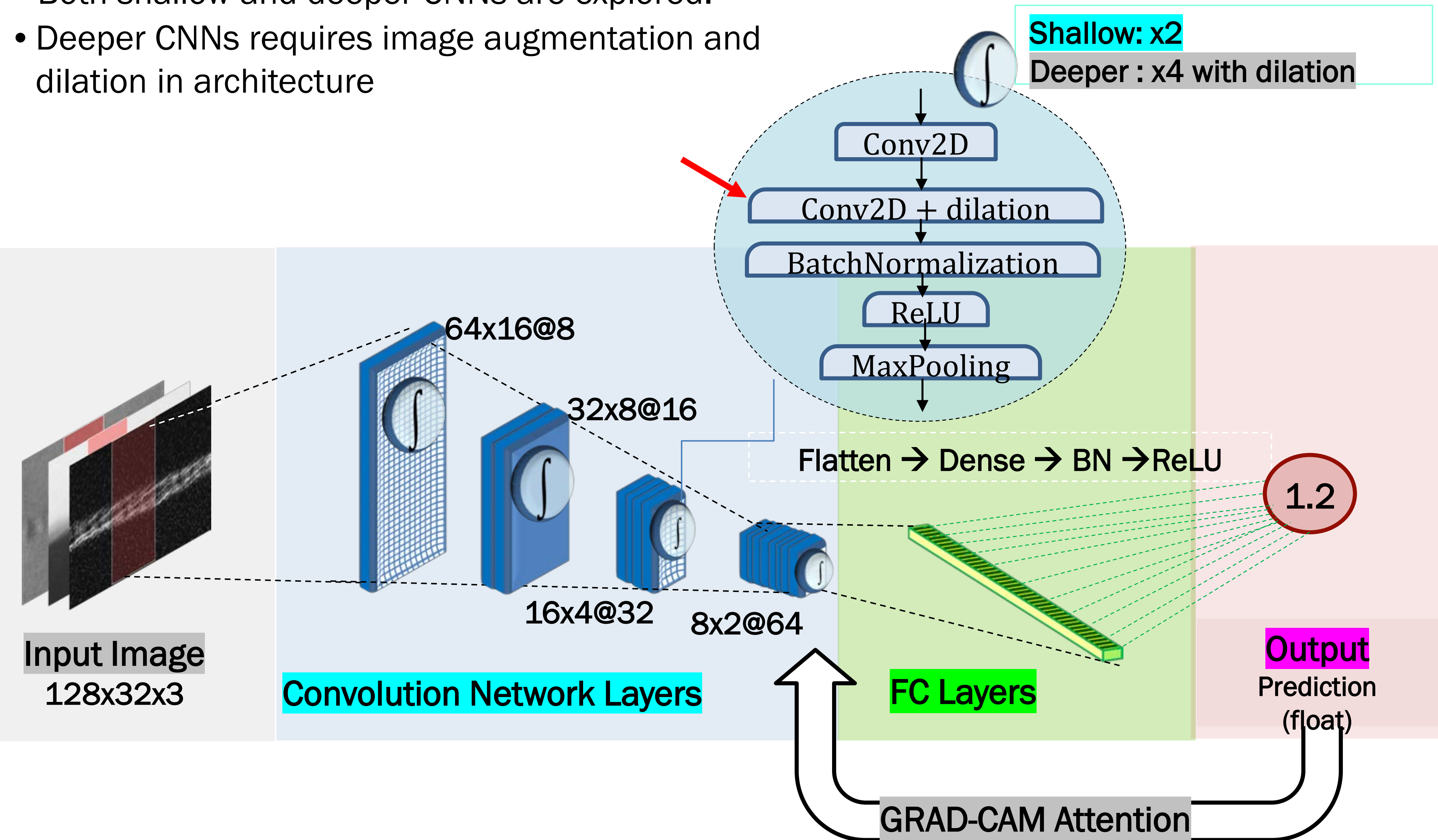
Random Zoom
0- 25%



Random Shift
0- 20%

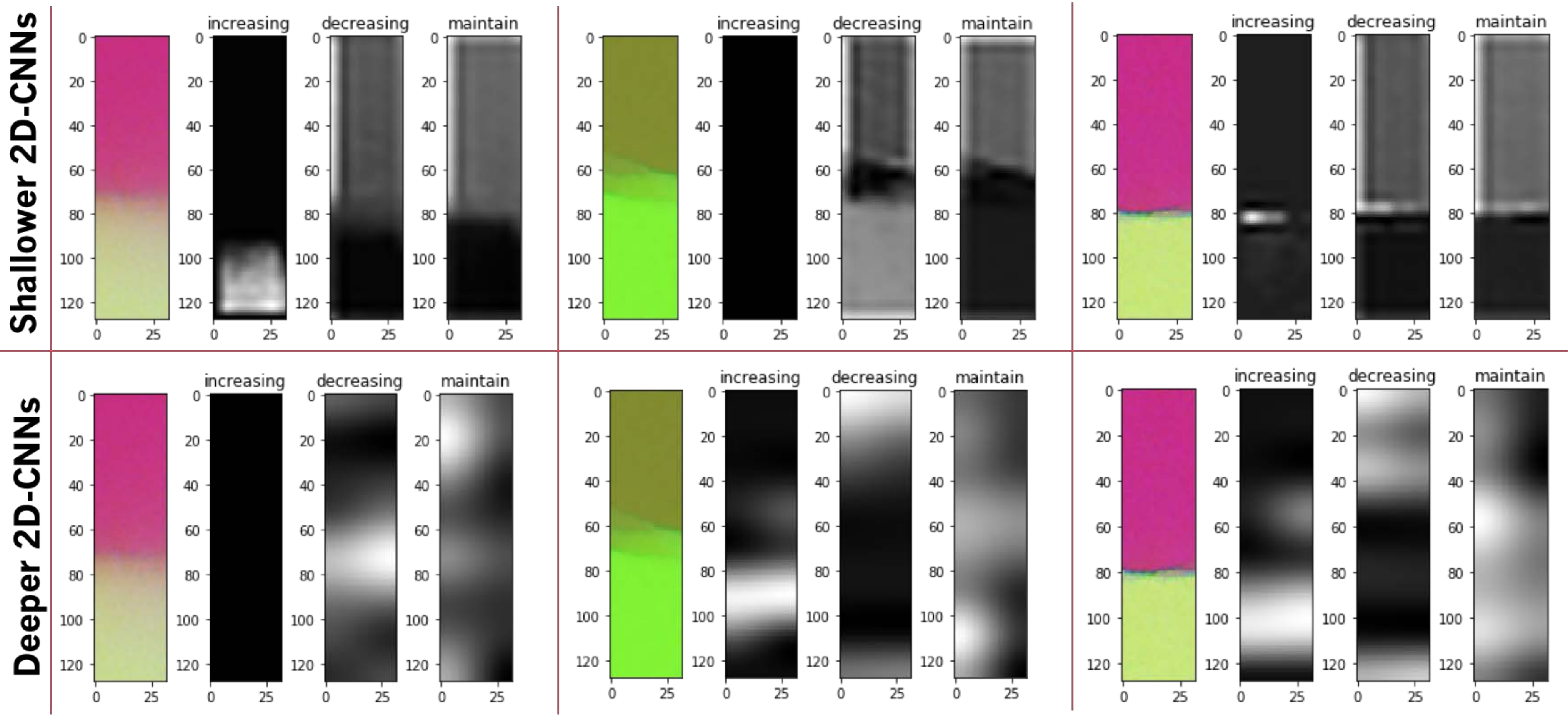
Model Architecture

- Both shallow and deeper CNNs are explored.
- Deeper CNNs requires image augmentation and dilation in architecture



Where CNNs look at while predicting?

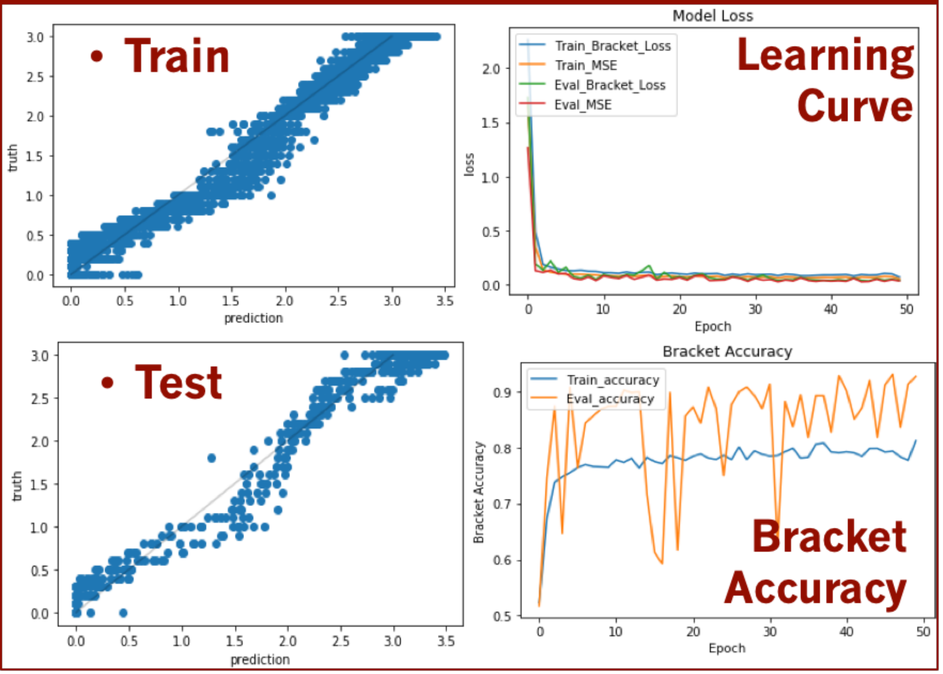
GRAD_CAM Attention Map



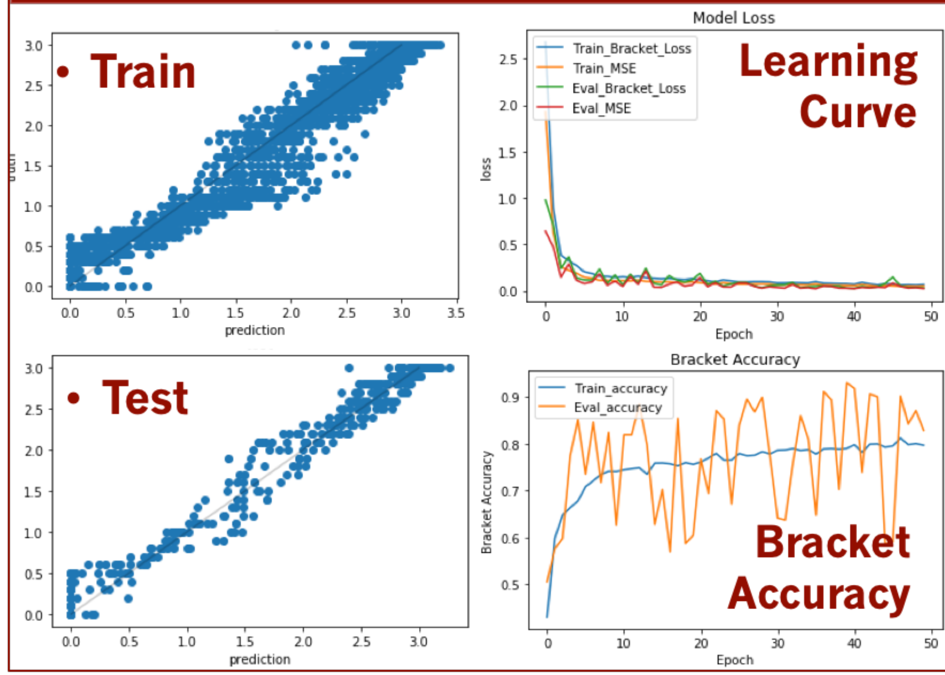
Use both near-field and far-field information to predict fault stages

Training and Evaluation Results

2D-CNN Shallower:



2D-CNN Deeper: Dilation, Augmentation, Bracket Loss



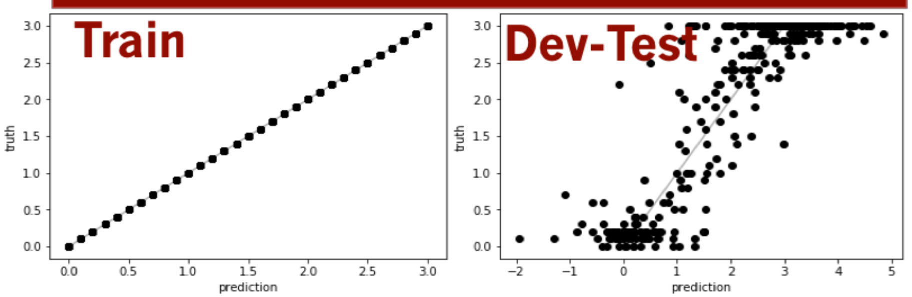
Tested Models	Params	MSE		Bracket Loss		Bracket Accuracy	
		Train	Test	Train	Test	Train	Test
2D CNN : shallower Bracket Loss	AdamOptimizer, Lr = 5e-3 Epoch=50, Batch-size = 32 momentum = 0.8	0.038	0.024	0.029	0.048	81.19%	85.82%
2D CNN : deeper Dilation + Augmentation + Bracket Loss		0.031	0.022	0.036	0.026	87.55%	89.80%
2D CNN : deeper Dilation + Augmentation + MSE		0.027	0.020	0.051	0.028	74.60%	67.55%

Baseline Performance?

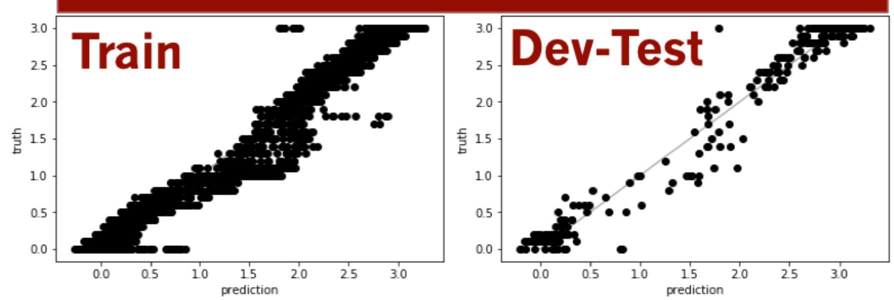
- No existing ML, DL study on fault stage prediction
- Linear Regression show extreme overfitting behaviors
- Slightly improved generalization with added regularization (tuned ridge regression). But model still does not perform on unseen data

	Regularization	Train MSE	Dev-Test MSE	Dev-Test Bracket Accuracy
Linear Regression	None	0.00	0.40	53%
Ridge Regression	$\alpha = 0.8$	0.03	0.04	61%

Linear Regression
Label VS Prediction



Ridge Regression
Label VS Prediction



Conclusions

The CNNs models

- Predict fault stages with accuracy of ~86% (shallower) ,~ 89% (deeper) from CNNs, a significant improvement from linear regression baseline (60%).
- The deeper CNNS outperforms shallower CNNs, but required image augmentation and dilation CNNs filters.
- Custom “Bracket” Loss improves training and test accuracy. Continuous prediction that characterized pretty accurately into 4 groups.

GRAD_CAM Attention

- Help finalizing architecture choices and selecting preferred CNNs among various choices that make identical predictions
- Though, deeper CNNs perform better, the shallower CNNs’ attention maps are more interpretable.

Successful workflow to predict at experimental scale

Time-lapse displacement infer mechanical deformation in different scale?

- Fault detection using modern seafloor topography offshore California?
- Empirical relationship in grain-scale mechanical deformation?

Acknowledgements

- Michele Cooke, Physical Modeling Lab at UMass Amherst
- CS 230 Instructor and TAs
- Tapan Mukerji, Steve Graham, Gary Mavko
- Basin and Petroleum System Modeling Group, Stanford