Deep Learning as Fold Geometry Classification Tool
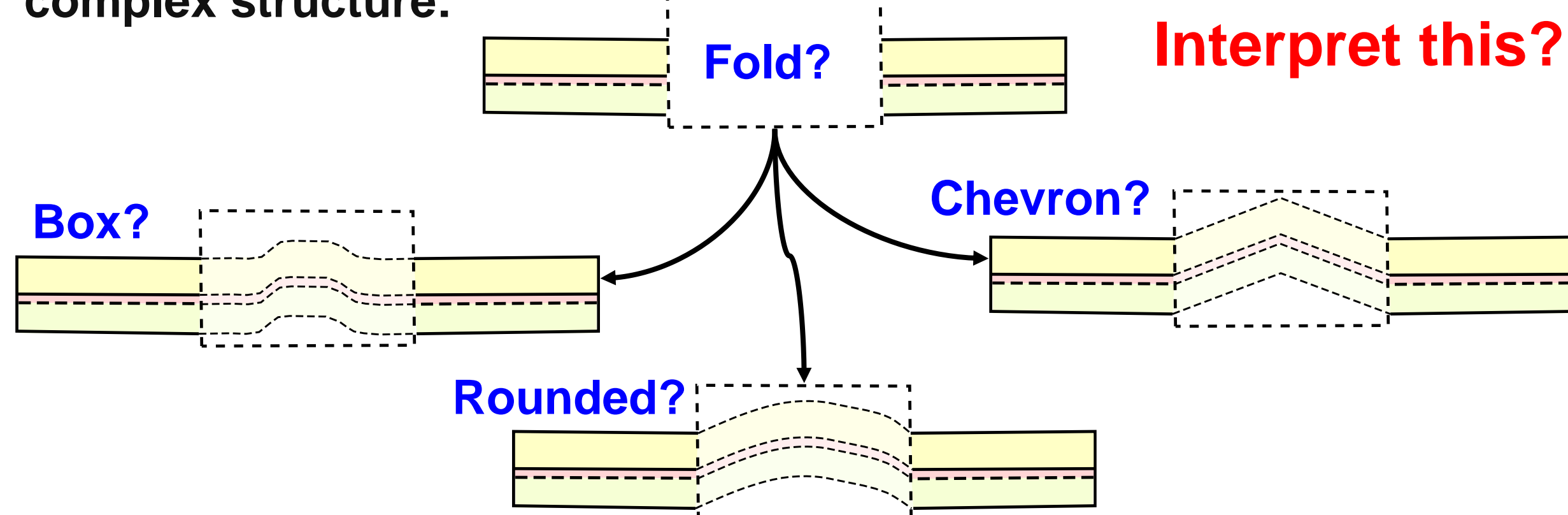
Ramy Abdallah, Clare E. Bond, and Robert W.H. Butler;   *r.abdallah.18@abdn.ac.uk*
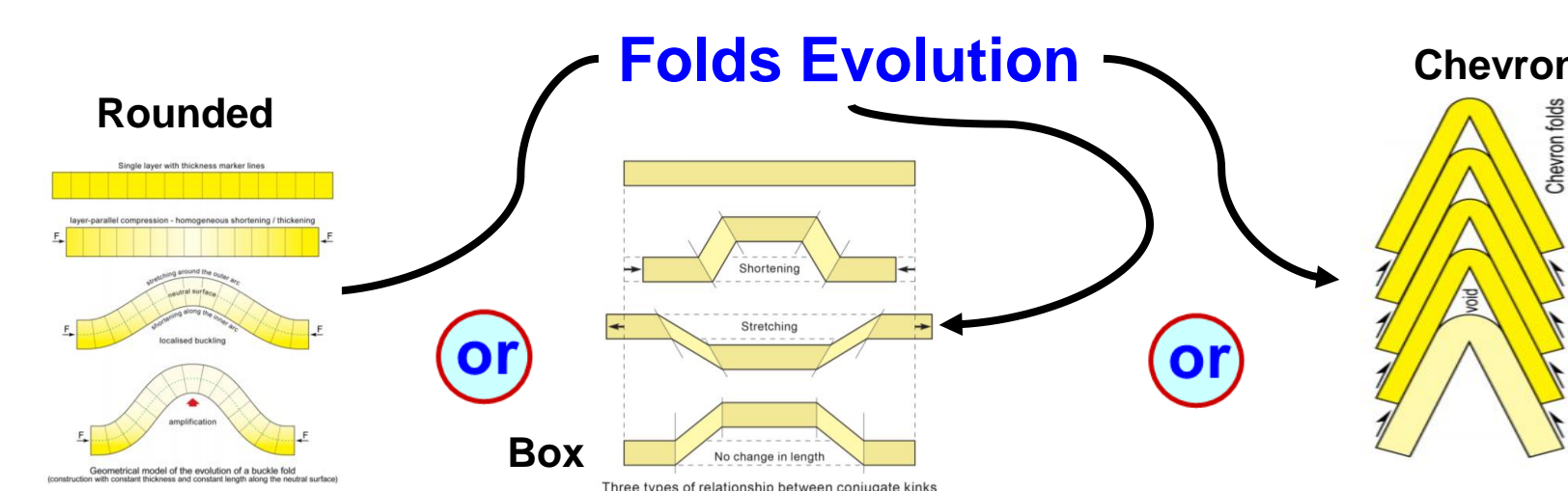
Geology and Geophysics, School of Geosciences, University of Aberdeen, Kings College, Aberdeen, AB24 3UE, UK

UNIVERSITY OF ABERDEEN — 1495

1. Why perform fold classification?

1. Provide additional information and constraints when interpreting complex structure:

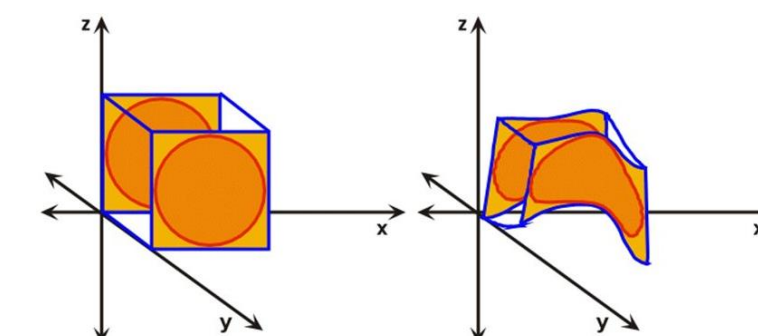Fold? — Box? — Chevron? — Rounded? — *Interpret this?*

2. Helps to understand folding mechanisms and development:

Rounded — Folds Evolution — Chevron — or — Box — or

3. Helps to understand distribution and heterogeneity of strain:

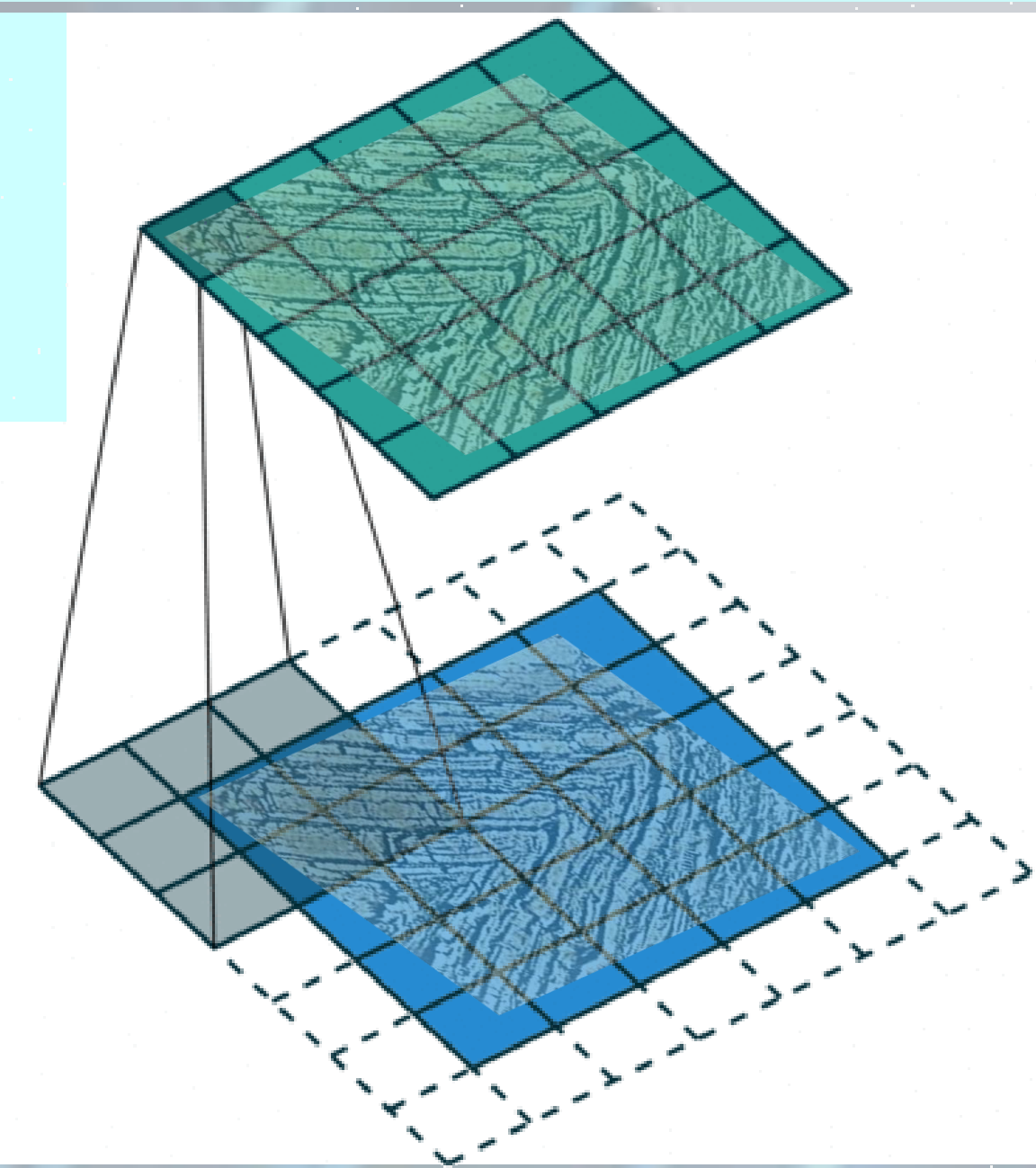Heterogeneous or Inhomogeneous Strain
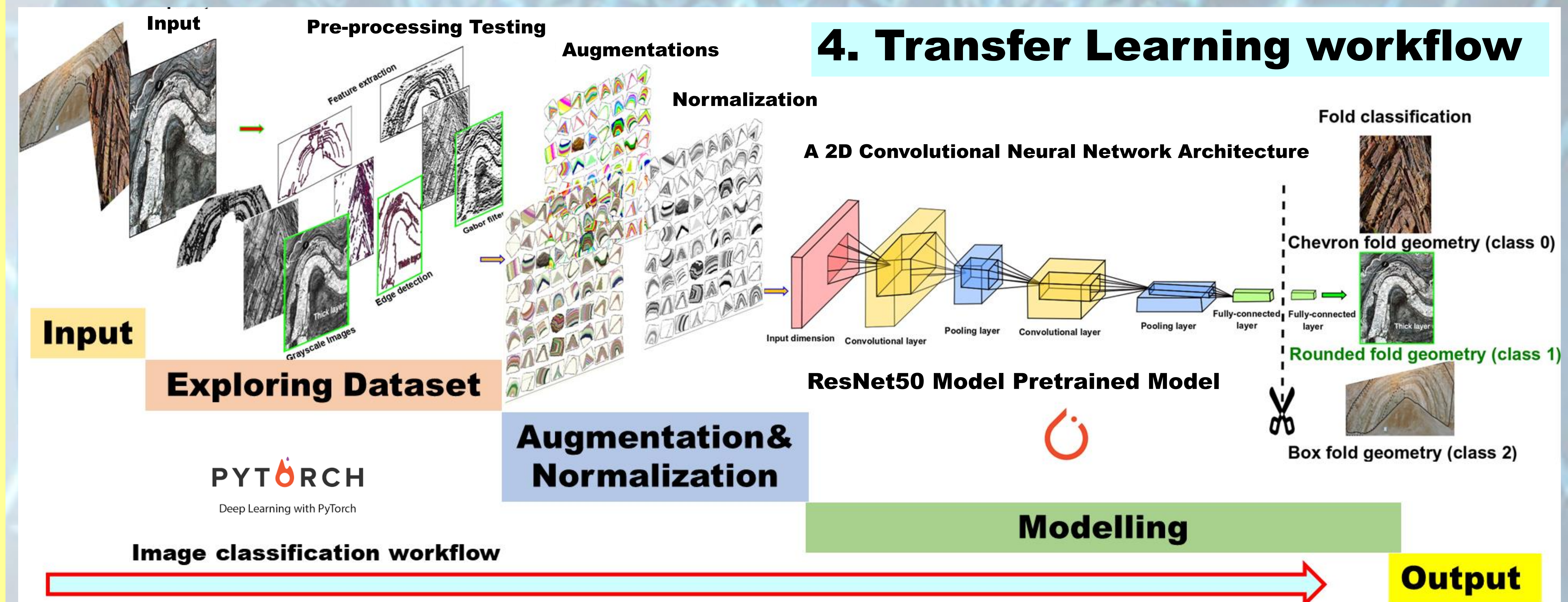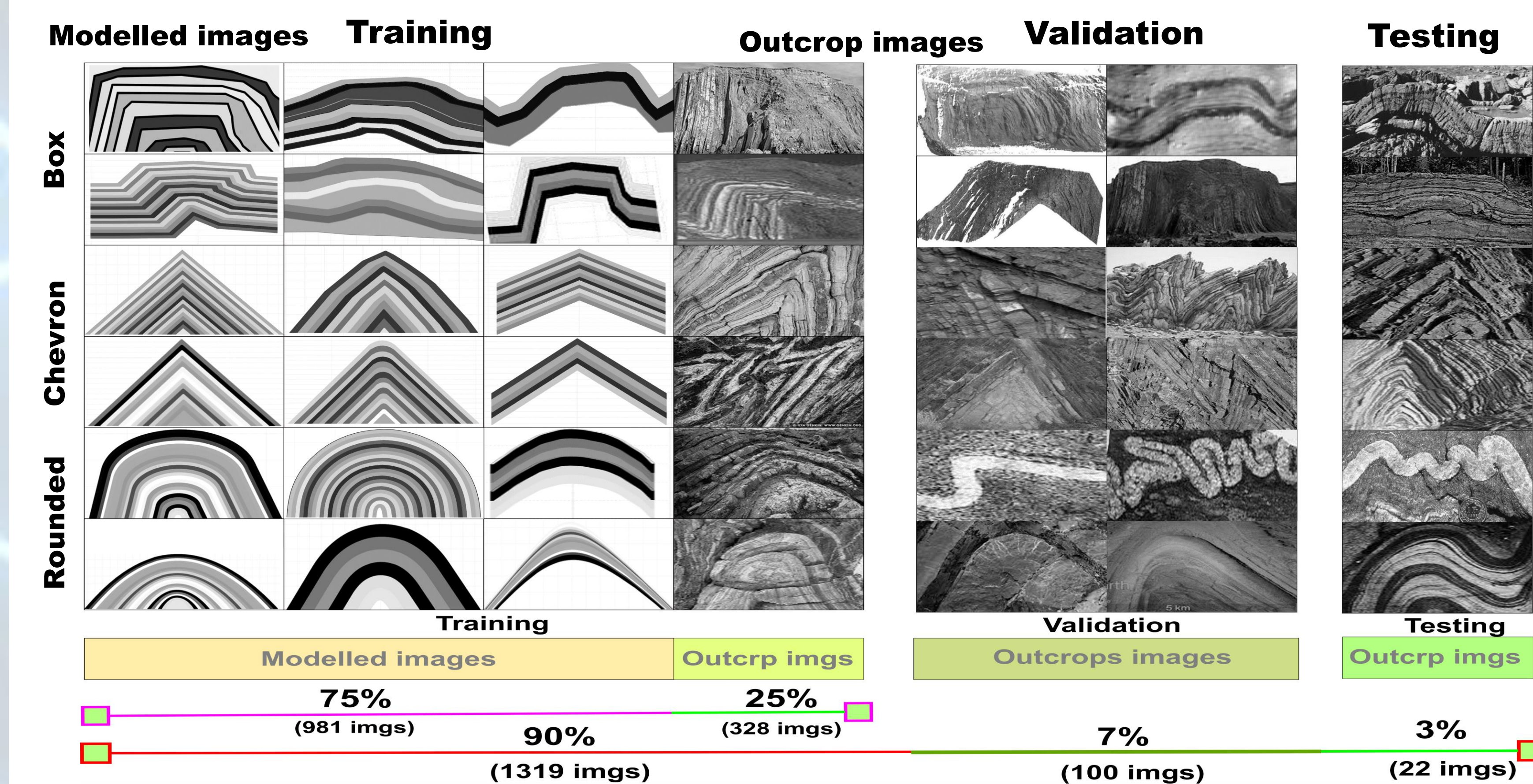
2. Image Classification

Using 2D Convolutional Neural Networks
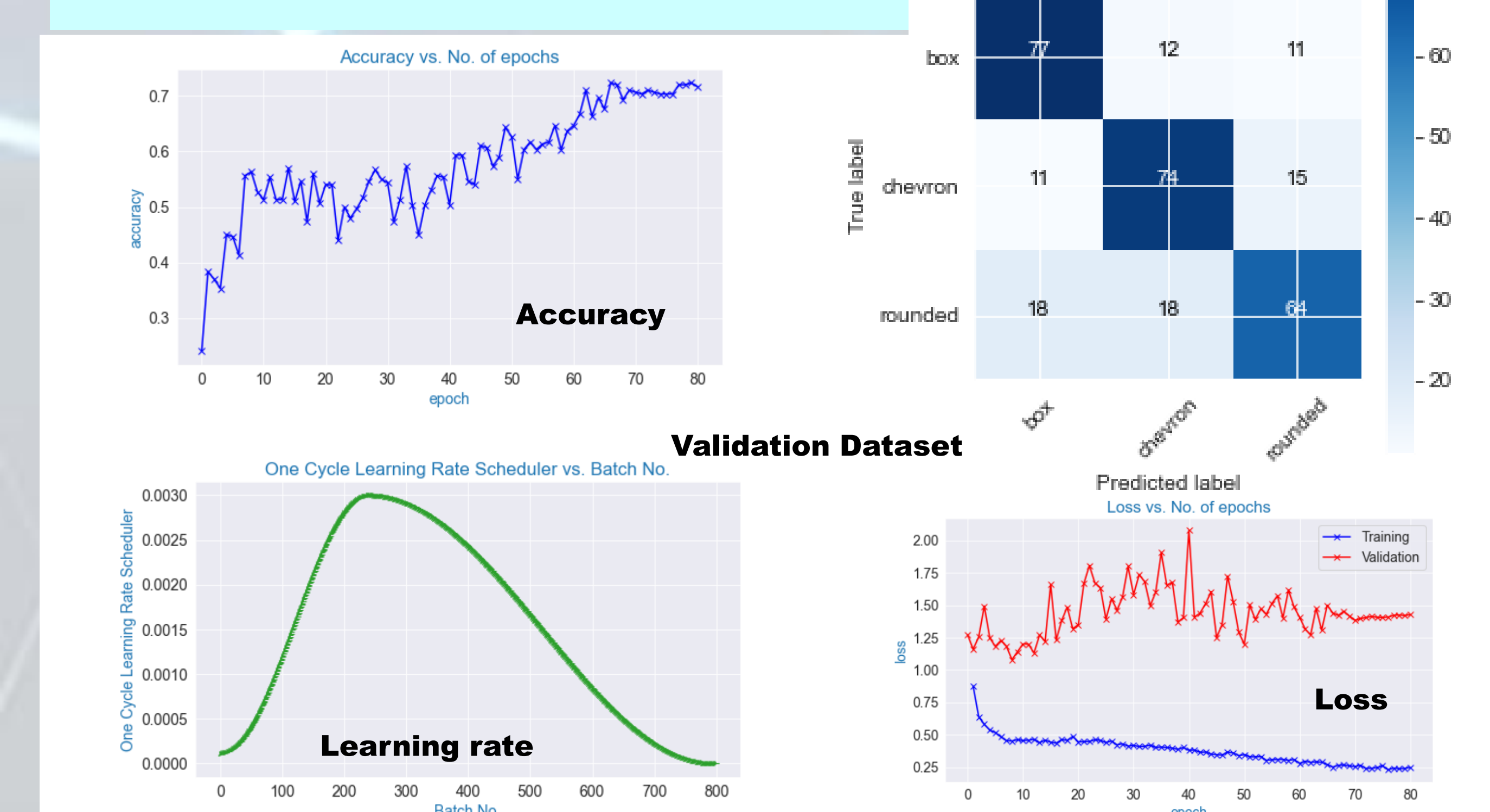
2D convolution operation:
- Start with a kernel, which is simply a small matrix of weights.
- This kernel (slides) over the 2D input data, performing an elementwise multiplication with the part of the input it is currently on.
- Then summing up the results into a single output pixel.

3. Shape of Hinges Dataset

Modelled images — Training — Outcrop images — Validation — Testing

Box — Chevron — Rounded

Training		
Modelled images	Outcrp imgs	
75% (981 imgs)	25% (328 imgs)	

Validation	Testing
Outcrops images	Outcrp imgs

90% (1319 imgs)	7% (100 imgs)	3% (22 imgs)

4. Transfer Learning workflow

Input — Pre-processing Testing — Augmentations — Normalization — Fold classification

Feature extraction — Edge detection — Gabor filter — Greyscale Images — Thick layer

A 2D Convolutional Neural Network Architecture

Input dimension — Convolutional layer — Pooling layer — Convolutional layer — Pooling layer — Fully-connected layer — Fully-connected layer — Thick layer

ResNet50 Model Pretrained Model

Chevron fold geometry (class 0)
Rounded fold geometry (class 1)
Box fold geometry (class 2)

Input — Exploring Dataset — Augmentation & Normalization — Modelling — Output

PYTORCH — Deep Learning with PyTorch

Image classification workflow

5. Model Evaluation

Confusion matrix

Accuracy vs. No. of epochs — Accuracy

One Cycle Learning Rate Scheduler vs. Batch No. — Learning rate

Validation Dataset — Loss vs. No. of epochs — Loss

	box	chevron	rounded
box	77	12	11
chevron	11	74	15
rounded	18	18	64

True label / Predicted label

6. Deep Learning - Results

Learning	Model	Training	Types of Data	Accuracy (%)	Drive (CPU) - time	Colud (GPU) - time
Shallow	Random Forest	Untrained	Test	57	13min 3s	-
			Val	64		
Deep	ResNet9	Untrained Greyscale	Test	61.7	2h 12min 43s	20min 35s
			Val	55.7		
		Untrained Coloured	Test	69	2h 2min 14s	22min 45s
			Val	60		
Transfer	ResNet34	Untrained	Test	42.6	2h 4min 54s	25min 15s
			Val	45		
		Pretrained	Test	63	2h 7min 25s	25min 9s
			Val	60		
	ResNet50	Untrained	Test	33.8	3h 52min 48s	3h 27min 2s
			Val	45		
		Pretrained	Test	70.6	3h 14min 58s	30min 35s
			Val	62		
	ResNet101	Untrained	Test	39.7	3h 8min 15s	32min 11s
			Val	38.9		
		Pretrained	Test	61.7	4h 58min 34s	44min 9s
			Val	61		
	ResNet152	Untrained	Test	33.8	3h 33min 1s	34min 7s
			Val	36.7		
		Pretrained	Test	66.2	6h 1min 48s	58min 6s
			Val	61.3		

ResNet50 Pretrained Model Accuracy over 70%

7. Classification Examples

Label: chevron , Predicted: box
Label: chevron , Predicted: chevron
Label: chevron , Predicted: chevron
Label: chevron , Predicted: chevron
Label: rounded , Predicted: rounded
Label: rounded , Predicted: chevron
Label: rounded , Predicted: box
Label: chevron , Predicted: chevron
Label: chevron , Predicted: chevron
Label: chevron , Predicted: chevron
Label: rounded , Predicted: rounded
Label: rounded , Predicted: box
Label: rounded , Predicted: rounded
Label: box , Predicted: rounded
Label: box , Predicted: box
Label: box , Predicted: chevron
Label: rounded , Predicted: rounded
Label: box , Predicted: box