https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

Towards robust fracture mapping: benchmarking automatic fracture mapping in 2D outcrop imagery

Ayoub Fatihi[1], Jefter Caldeira[1], Tom Beucler[2,3], Samuel T. Thiele[4], and Anindita Samsu[1]

[1]Institute of Earth Sciences, University of Lausanne, 1015 Lausanne, Switzerland
[2]Institute of Earth Surface Dynamics, University of Lausanne, 1015 Lausanne, Switzerland
[3]Expertise Center for Climate Extremes, University of Lausanne, 1015 Lausanne, Switzerland
[4]Helmholtz Institute Freiberg, Helmholtz-Zentrum Dresden-Rossendorf, Chemnitzer Str. 40, 09599 Freiberg, Germany

Correspondence: Ayoub Fatihi (ayoub.fatihi@unil.ch)

Abstract. Extracting consistent and accurate fracture traces from large volumes of high-resolution imagery remains a persistent challenge in structural analysis. We present a harmonised benchmarking dataset, FraXet, for pixel-wise fracture segmentation in high-resolution RGB orthophotos and digital elevation models (DEMs). FraXet curates images from three publicly available datasets, totalling 8953 256×256 RGB+DEM patches spanning diverse lithologies and imaging conditions. We use this dataset to systematically assess traditional image-processing filters (Canny, Sobel, Gabor, Sato, phase congruency) and two deep-learning (DL) models, U-Net and SegFormer, for per-pixel fracture detection. Quantitative comparison using image-quality (e.g., MSE, PSNR), segmentation (e.g., Precision, Recall, F1, IoU) and proposed similarity FracSim metrics suggest that the deep models substantially outperform classical filters (F1 $\approx$ 03 -0.5 vs $\leq$ 0.29), giving smoother, more continuous fracture traces with reduced noise. Training on the combined dataset (M_all) improves cross-site generalisation relative to models trained on the individual sub-datasets. Challenges remain in handling annotation misalignments, illumination artifacts, and thin traces. More importantly, probability maps derived from the DL approaches enable confidence-based triage and visualisation of model uncertainty. This work thus establishes a unified benchmark, curated dataset, and reproducible baseline to support further development of robust automated tools for fracture detection.

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

1 Introduction

Fractures, formed through brittle deformation when stress exceeds rock strength, are important structures that influence both rock stability and fluid flow (Twiss and Moores, 2006). They can reduce cohesion, posing risks in seismically active areas and engineered settings (Cappa et al., 2018). They can also enhance permeability—especially in low-porosity rocks like limestone—by creating pathways for fluid movement (Aydin, 2000). Fractures are therefore key in exploring for and extracting hydrocarbons and geothermal energy, as well as assessing sites for nuclear waste storage.

Researchers often use outcrop analogues to better understand subsurface fracture geometries (Smeraglia et al., 2021). Recent advances in remote sensing have transformed how geological fractures in outcrops are documented. Drone-based imaging now allows the creation of high-resolution, georeferenced orthophotos that capture rock surfaces in remarkable detail. These digital data enable precise mapping, measurement, and analysis of structural features like orientation and spacing, complementing traditional field observations and providing access to outcrops that are difficult or unsafe to reach. However, while data acquisition has become faster and easier, the interpretation of these images remains time-consuming and subjective. The visibility of fractures depends on lighting, surface weathering, vegetation cover, and image resolution, and the process of manual or semi-automated mapping is often influenced by interpreter bias (Bond et al., 2007; Scheiber et al., 2015; Peacock et al., 2019; Andrews et al., 2019). Furthermore, mapping a 10 × 10 m area with 1 cm resolution can still take over half an hour, even with semi-automated tools (Thiele et al., 2017b). These challenges highlight the growing need for faster, more objective, and more reproducible fracture mapping methods.

With the rapid progress of computer vision (CV) and machine learning (ML), automated fracture detection has become an increasingly appealing alternative. Various techniques have been developed, ranging from classical image processing methods—including edge detection, Hough transforms, and color-based segmentation—to more advanced deep learning (DL) approaches (Table 1). Each method offers different benefits depending on data type and imaging conditions. Among these, deep learning has emerged as particularly promising, because it can automatically learn spatial features at multiple scales (LeCun et al., 2015), allowing models to distinguish complex but distinctive fracture geometry from noise and other "edges" commonly found in drone imagery.

Table 1. Summary of methods and their characteristics for fracture mapping

Method	Algorithms / Key studies	Data types	Description	Advantages	Limitations	Automation
Colorimetry-Based Segmentation	Thresholding approaches (Ren and Malik, 2003; Vasuki et al., 2017)	Orthophotos	Segments image features based on pixel intensity. Fractures appear darker due to illumination differences, while the host rock appears lighter.	Simple and intuitive; effective in high-contrast conditions.	Strongly affected by lighting and shadows; not reliable in heterogeneous or low-contrast scenes.	Manual or semi-automatic
DEM/DTM-Based Feature Extraction	Segment Tracing Algorithm (Koike et al., 1995), Segment Tracing and Rotation Transformation (Raghavan et al., 1995), Directional Detection (Raghavan et al., 1993), Line Segment Tracking (Fan and Ni, 2023; Gaikwad et al., 2023)	DEMs, DTMs, satellite imagery	Uses terrain derivatives (curvature, slope, relief) to reveal lineaments. Segment tracing improves feature continuity.	Reveals terrain-related or hidden subsurface structures.	Dependent on resolution and quality of DEMs; computationally demanding.	Semi-automatic or automatic (ML used in some)
Edge Detection (traditional filters)*	Canny Edge Detection (Masoud and Koike, 2011; Soto-Pinto et al., 2013; Adiri et al., 2017), Sobel, Laplacian of Gaussian; combinations such as Canny + Hough Transform (Han et al., 2018)	Orthophotos, satellite imagery, DEMs	Gradient filters highlight intensity changes; often used as preprocessing before other algorithms.	Fast, well established, interpretable.	Sensitive to noise and illumination; lacks context.	Semi-automatic
Phase Symmetry / Phase Congruency*	Phase Symmetry and Congruency (Kovesi, 1999, 2000; Vasuki et al., 2014)	Orthophotos	Uses local phase information to detect edges invariant to brightness or contrast.	Robust under varying illumination; noise-tolerant.	Computationally heavy; more complex to implement.	Semi-automatic
Hough Transform	Classical Hough Transform (Duda and Hart, 1972); combinations such as Canny + Hough Transform (Soto-Pinto et al., 2013; Han et al., 2018)	Binary edge maps (from other detectors)	Converts edge pixels into parameter space and accumulates votes to identify linear structures.	Good for linear fractures; interpretable.	Limited to straight lines; prone to false positives.	Semi-automatic

* evaluated in this study

Table continued on next page

Table 2. Summary of methods and their characteristics for fracture mapping

Method	Algorithms / Key studies	Data types	Description	Advantages	Limitations	Automation
Wavelet and Multi-Scale Transforms	Wavelet Transform (Mallat and Hwang, 1992), Ridgelets and Curvelets (Candès and Guo, 2002; Candès and Donoho, 2005; Do and Vetterli, 2005), Complex Shearlet Transform (Prabhakaran et al., 2019)	Orthophotos, satellite imagery, DEMs	Multi-scale decomposition; directional wavelets capture curvilinear and branching structures.	Multi-scale, effective for curved fractures.	Classical wavelets lack directionality; advanced transforms are complex and computationally intensive.	Semi-automatic
Line Segment Detection / Tracking	Line Segment Detection (Rahnama and Gloaguen, 2014; Masoud and Koike, 2017; Saint Jean Patrick Coulibaly et al., 2021), Line Segment Tracking (Fan and Ni, 2023; Gaikwad et al., 2023)	Satellite imagery, DEMs	Detects line segments directly; tracking improves continuity across scales.	Captures both short and long features; efficient.	Sensitive to parameter tuning; may fragment lineaments.	Automatic (ML used in some)
Deep Learning and Machine Learning Approaches*	CNN U-Net (Chudasama et al., 2024), Geological Adaptive Incremental Network (Zhang et al., 2024), DL + Line Detection (Oliveira et al., 2019), Multi-feature DL integration (Zhang et al., 2024), Graph Attention Network + Fast Line Detection, Photogrammetry + ML (Pola et al., 2024)	Orthophotos, DEMs, satellite imagery, 3D point clouds	Learns hierarchical spatial features for segmentation and fracture extraction.	Robust to noise; captures complex nonlinear spatial patterns; transferable across similar datasets.	Requires large labelled data; prone to false positives (e.g., vegetation, shadows); can be expensive to train.	Fully automatic (DL/ML-based)

* evaluated in this study

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

However, a persisting issue is that automated fracture detection still struggles with false positives and negatives, particularly under natural and uncontrolled imaging conditions. Natural outcrops contain shadows, vegetation, uneven lighting, and erosion, all of which can confuse detection algorithms. Moreover, fractures are an unusual target for semantic segmentation, as they are better represented as linear segments with arbitrary width rather than as continuous areas. Most fracture analyses also require instance-level outputs—such as fracture length, orientation, spacing, and connectivity—which depend directly on the correct topology of each trace. As a result, even minor misclassification can drastically alter fracture connectivity and the derived measurements. Furthermore, the ground-truth labels used for training are often drawn at coarser scales than pixel-level precision, leading to misalignment between the labelled and actual fracture edges (Figure 1). These inconsistencies affect both model learning and evaluation accuracy.

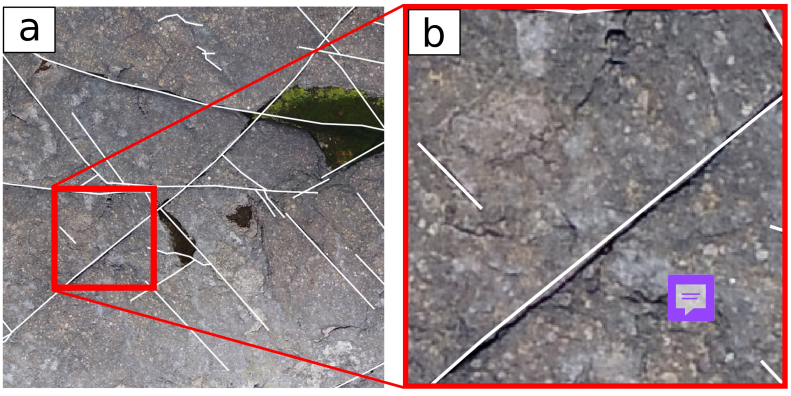

Figure 1. Example of annotation misalignment. (a) Outcrop orthophoto (Nordbäck and Ovaskainen, 2025) overlaid with fracture traces in white from the ground-truth labels (Ovaskainen and Nordbäck, 2022). (b) Zoomed-in view showing how manually drawn fracture lines deviate from the actual visible fracture edges.

For fracture mapping and similar tasks, two main computer-vision approaches are commonly used: semantic segmentation and instance segmentation (Sharma et al., 2022). Semantic segmentation labels each pixel as a fracture or not, producing a binary mask without separating individual fractures. Instance segmentation, on the other hand, identifies and labels each distinct fracture, allowing direct measurement of geometric attributes. This study focuses on semantic segmentation because it is simpler, more robust with current data, and does not require instance-level labels. Instance segmentation remains the ultimate goal for full fracture analysis but is highly challenging in this context and beyond the scope of this work.

Algorithm performance also often drops when applied to new datasets with different lithologies, topographies, or imaging conditions, limiting generalizability. This underscores the need for standardised benchmarking frameworks and shared datasets that allow consistent and fair comparison of methods (An et al., 2023). While initiatives like GeoCrack (Yaqoob et al., 2024) have begun to address this gap, current resources remain limited to RGB imagery and predominantly vertical outcrops and prone to incomplete annotations.

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

Our approach addresses this limitation by curating three heterogeneous benchmark datasets and applying clear and consistent preprocessing, hyperparameter optimisation, and evaluation procedures. Performance is assessed using both image-quality metrics (e.g., MSE, PSNR) and segmentation metrics (e.g., precision, F1-score, IoU), as well as a novel geological similarity metric. The primary goal of this study is to establish a standardised evaluation framework for pixel-wise fracture segmentation from 2D outcrop imagery, incorporating both traditional computer vision and deep learning methods. This framework enables systematic comparison of detection performance across diverse datasets, geological settings, and imaging conditions. By harmonising data processing, parameter tuning, and evaluation metrics, it promotes transparency, reproducibility, and comparability in automated fracture mapping research, while also providing a useful test dataset and benchmark for future developments.

2 Materials and Methods

FraXet (Fatihi et al., 2025a) integrates three annotated outcrop datasets—**Ovaskainen22** (Nordbäck and Ovaskainen, 2025; Ovaskainen and Nordbäck, 2022), **Matteo21** (Mattéo et al., 2020), and **Samsu19** (Samsu et al., 2019)—selected for their combination of high-resolution orthomosaic and digital elevation models (DEMs), diverse lithologies and varied outcrop geometries. From these sources, we compile 8,953 co-registered 256×256 patches, each consisting of an RGB orthomosaic tile, a DEM tile, and a corresponding binary 1-pixel wide fracture label. We standardise and pre-process all patches, then apply a suite of traditional image-processing filters (Canny, Sobel, Gabor, Sato, phase congruency) and deep learning models (U-Net, SegFormer) to generate per-pixel fracture probability maps. All methods are evaluated using a set of image-quality, segmentation and proposed similarity metrics on held-out test splits, with deep models tuned on validation subsets before final testing.

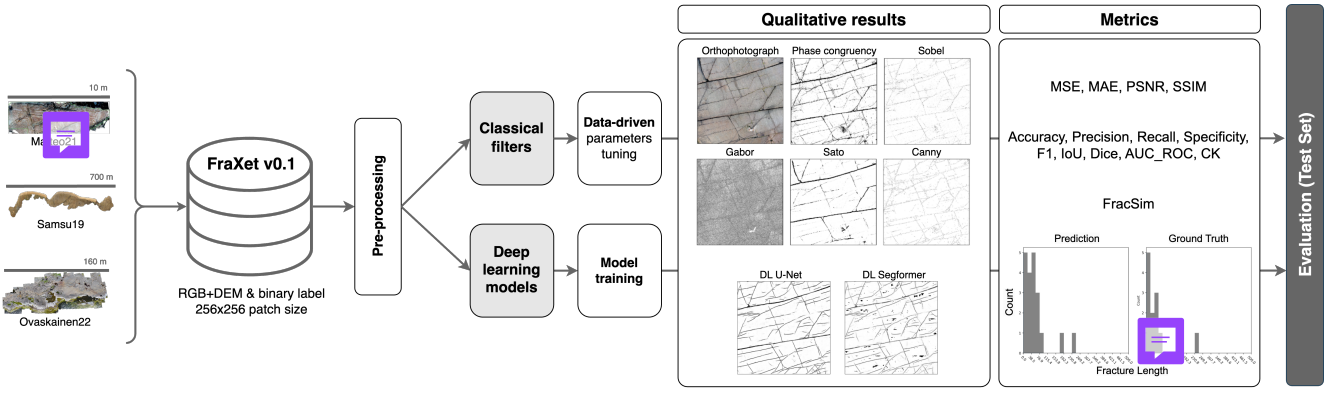

Figure 2. Overview of the benchmarking workflow. The three datasets (Matteo21, Samsu19, Ovaskainen22) are processed into the FraXet v0.1 database, used to train classical filters and deep learning models, the results of which are evaluated on the test set through qualitative visualisation and quantitative metrics and proposed similarity metric.

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

2.1 Data Compilation

2.1.1 Data description

Ovaskainen22 captures fracture networks in the Mesoproterozoic Wiborg Rapakivi Granite Batholith of southeastern Finland, mostly isotropic coarse-grained wiborgite facies granite with K-feldspar megacrystals up to 5–10 cm (Härmä, 2020). Matteo21 (Mattéo et al., 2021) presents data from the Granite Dells, Arizona, an anorogenic granite of similar age (∼1.40 Ga) but with medium-grained, and generally equigranular textures showing weak porphyritic tendencies (Krieger, 1965). Samsu19 (Samsu et al., 2019) map fractures in a very different geological setting: a fluvial–lacustrine syn-rift sediment succession in the Lower Cretaceous Strzelecki Group (Southeastern Australia), comprising mud- and sand-dominated siliciclastic strata with conglomeratic or organic-rich interbeds (Constantine, 2001).

Together, these three datasets cover varied geological contexts (isotropic crystalline to anisotropic sedimentary lithologies) with fractures mapped at different scales, providing a diverse testbed for evaluating model generalisation.

In addition, a single site from the Lapiés di Bou karst surface was included to enable qualitative visual comparison across methods. This locality lies within the Helvetic domain of the Swiss Western Alps, specifically within the Wildhorn Nappe complex (Steck et al., 2001; swisstopo, 2024), where fine limestones to marls, locally oolitic, of the Aptian–Barremian Schrattenkalk Formation crop out (Badoux et al., 1959). Two patches from the Bingie Bingie area were also used for the timing experiment, allowing direct comparison with the manual and semi-automatic mapping workflow of Thiele et al. (2017a).

2.1.2 Test, train and validation splits

To ensure reliable model assessment and minimise spatial autocorrelation—a known issue in geospatial machine learning where spatial overlap inflates accuracy metrics by reducing the statistical independence of training and testing data (Wang et al., 2023)—we have defined spatially disjoint data splits by dividing each of the three datasets into training, validation, and test regions (with no spatial overlap). Specifically, here the Ovaskainen22 dataset consists of Orrengrund (training), Kasaberget (validation), and Kampuslandet (testing) islands off the coast of the Loviisa region; the Samsu19 dataset comprises Eagles Nest (training) and Harmers Haven north (split between validation and testing); and the Matteo21 dataset is split into sites A and B (training), held-out regions of A and B (validation), and site C (testing). Only tiles containing at least one labelled fracture pixel were retained, to reduce challenges with class imbalance and to focus learning on relevant features.

The orthophotos and DEMs were resampled to have matching spatial resolution, normalised, and tiled into non-overlapping 256 × 256 pixel patches, each with four channels (RGB+DEM). This patch size was chosen as a compromise between field of view, feature scale, and computational efficiency—large enough to preserve contextual information while remaining compatible with typical convolutional neural network (CNN) input constraints.

Ground truth annotations for Ovaskainen22 and Samsu19 were provided as vector shapefiles and were subsequently rasterised to the corresponding image resolution, producing binary masks with 1-pixel-wide white fracture lines on a black background. These binary images match the raster-formatted annotations in Matteo21 to ensure consistent label representation.

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

Table 3. Overview of datasets and their metadata

Dataset	Resolution (m)	Train patches	Validation patches	Test patches	Total patches	Total Area of used patches (m^2)
Ovaskainen22	0.005 - 0.006	2808	966	671	4445	7220
Matteo21	0.0005 - 0.0013	2174	323	912	3409	142
Samsu19	0.029 - 0.032	642	228	229	1099	68271
All	-	62.82%	16.94%	20.24%	8953 (100%)	75633

Probability Map Representation. Because our goal is to generate per-pixel probability maps rather than hard binary labels, we designed all preprocessing steps with this output in mind. Probability maps provide a continuous estimate (0–1) of the likelihood that each pixel corresponds to a fracture, capturing confidence and uncertainty rather than forcing discrete predictions. This representation is well suited to fracture mapping, where annotations are often noisy or slightly misaligned, and where soft boundaries reflect the inherent ambiguity of thin linear features.

It also enables flexible post-processing: thresholds can be adjusted to balance precision and recall, high-confidence regions can be skeletonised to extract fracture traces, and outputs can be integrated into human-in-the-loop workflows or combined with traditional filters. Moreover, probability maps support richer evaluation metrics (e.g., ROC analysis), and improve robustness when using ensembles or multi-scale fusion. For all these reasons, probability maps serve as the most informative and practical output of the methods.

2.2 Pre-processing

Several preprocessing and augmentation steps were applied prior to model training and evaluation, to standardise inputs, refine ground-truth annotations, and improve model generalisation. All preprocessing scripts and procedures are publicly released as the FraXtex2D github repository (https://github.com/ayoubft/fractex2D.pt) to ensure reproducibility. These procedures are summarised below.

Depth. For all deep learning experiments, we used the full RGB imagery together with the corresponding DEM channel, which was normalised at the patch level using min–max normalisation. This scaling ensures that elevation values within each patch fall between 0 and 1, reducing the influence of absolute elevation differences across sites and improving numerical stability during training.

Channel selection. Most traditional edge-detection filters operate on a single image band. Therefore, each band was first z-standardised (zero mean, unit variance) on the patch-level to reduce sensitivity to illumination differences and overall brightness. PCA (principal component analysis) was then applied to the normalised four-band data to obtain a single-band representation containing a maximal amount of variance.

Label refinement. We implemented two complementary label-processing techniques to correct inconsistencies in the original fracture annotations and enhance their spatial accuracy: wide-fracture expansion and multi-scale label smoothing.

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

Wide-fracture expansion. Preliminary experiments revealed that the models underperformed on wide and visually prominent fractures. Many such fractures were annotated as 1-pixel-wide lines, underrepresenting their true width. To address this, we developed a region-growing algorithm that widens the single-pixel annotations to fill thick fracture zones. This algorithm (Figure 3c) fills dark linear regions (in the green channel) by applying a maximum filter followed by morphological dilation and then merging the resulting (dark and connected) regions with the original binary mask. Only expanded areas connected to existing labelled pixels are retained to prevent false positives. This operation improved annotation fidelity for wide fractures.

Multi-scale label smoothing. Manual fracture tracing also often leads to small misalignments between annotations and the actual fracture pixels, especially in high-resolution orthophotos. We applied a multi-scale label-smoothing procedure to mitigate this issue, using successive dilations with decreasing weights to produce a graded boundary around each fracture (Figure 3d). This approach smooths the input binary classification labels, accounts for positional uncertainty, and helps stabilise the model's learning process.

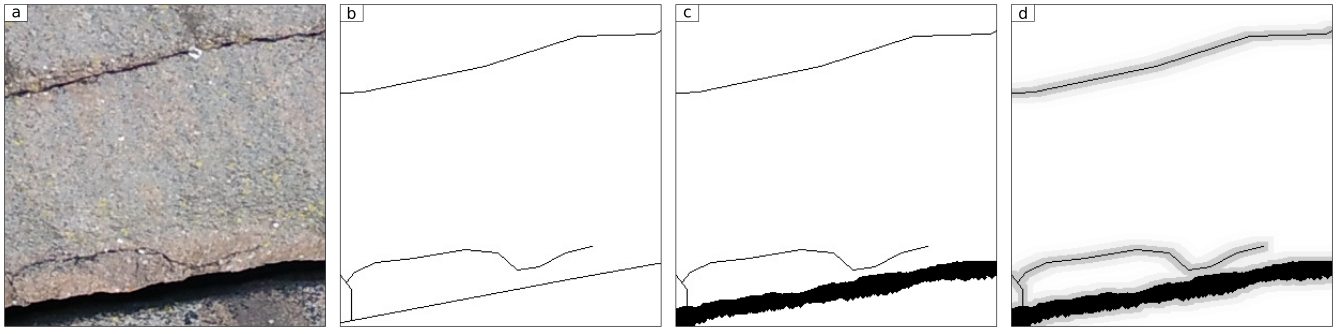

Figure 3. Label refinement steps. (a) Original RGB image; (b) manually traced ground truth with 1-pixel-wide fractures; (c) widened annotations after region-growing; (d) smoothed probabilistic labels produced by multi-scale dilation.

Together, these two steps improved correspondence between fracture labels and actual image structures, increasing the effective proportion of positive (fracture) pixels and slightly reducing class imbalance (Table A1 in the Appendix).

2.3 Data augmentation

To further enhance model robustness and generalisation, we applied the following data transformations randomly to the input patches (Figure 4) during each training epoch:

(1) Flipping horizontally and vertically. These transformations are applied to both the RGB+D images and the ground truth labels to encourage geometric invariance, helping the model detect fractures regardless of their orientation.

(2) Gaussian blurring, brightness, contrast, and saturation adjustment. These are applied only to the RGB channels to promote photometric invariance, improving robustness to changes in lighting and appearance.

Table B1 in the Appendix summarises the applied augmentations and their parameters.

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

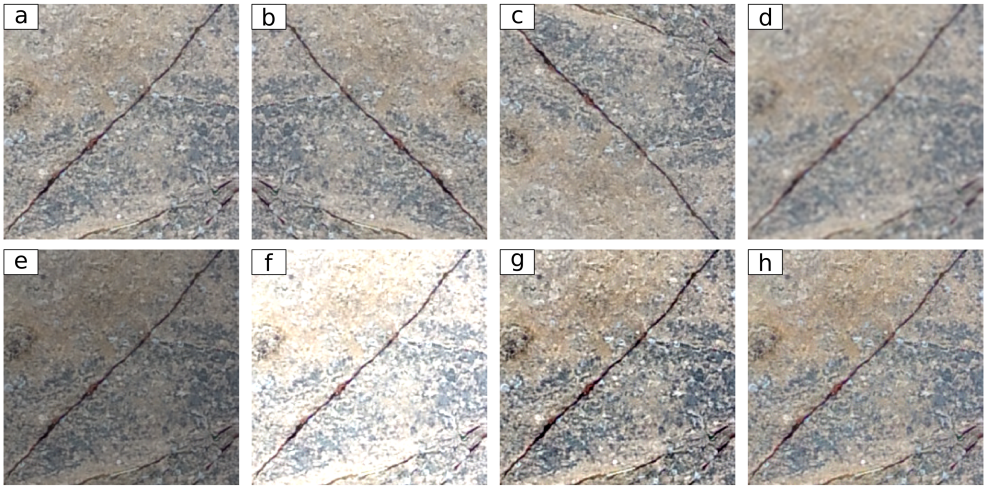

Figure 4. Data augmentations applied during training: (a) original, (b) horizontal flip, (c) vertical flip, (d) gaussian blur, (e) brightness decrease, (f) brightness increase, (g) contrast increase, (h) saturation increase.

2.4 Image filter-based approaches

The following section outlines our implementation of several established image processing filters, as a baseline to which we can compare the deep learning results. These methods do not require training data and operate directly on the image, offering fast, interpretable, and reproducible outputs. While not specifically designed for fracture detection, they are widely used in geoscientific and remote sensing applications to extract features with similar morphology. All filters were applied to the single channel resulting from the PCA reduction.

For the classical filters, which do not produce probabilistic outputs, we applied the same label-dilation step (Step 2 in preprocessing, 2.3) to their binary masks. This ensures that all methods are evaluated against labels that better reflect the actual fracture width and account for the misalignment between annotations and true pixel-level fracture geometry observed earlier.

The filters are grouped into three main categories, also see Table 1:

- Edge detection (Sobel and Canny): Computes spatial intensity gradients to highlight abrupt changes. These filters are simple and efficient but sensitive to noise and lighting, and some require thresholding to generate binary edges.
- Ridge detection (Gabor and Sato): Identifies elongated, linear structures by analyzing second-order derivatives (curvature). These filters can capture fracture-like ridges at tuned scales but may also respond to non-fracture textures.
- Phase-based filters (Congruency): Use Fourier phase information to detect edges. These methods handle varying illumination more robustly, but are computationally intensive and involve more tunable parameters.

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

Each of these methods relies on user-specified parameters (Table 4; §2.4.1) to suppress background clutter and isolate relevant target features (fractures in our case).

Table 4. Summary of parameters adjusted for each algorithm and their descriptions

Algorithm		Parameter	Parameter's description
Edge detectors	Sobel	-	-
	Canny	sigma	Standard deviation for Gaussian smoothing
		low_threshold	Lower bound for edge linking (hysteresis)
		high_threshold	Upper bound for edge linking
Ridge detectors	Gabor	frequency	Spatial frequency of the harmonic function
	Sato	sigmas	Tuple/list of Gaussian scales to apply
Phase Congruency		nscale	Number of wavelet scales
		norient	Number of orientations used
		minWaveLength	Wavelength of the smallest filter
		mult	Factor to scale filter size across scales
		sigmaOnf	Bandwidth of the log-Gabor filter
		k	Multiplier for noise threshold
		cutOff	Frequency spread threshold
		g	Controls sharpness of sigmoid weighting
		noiseMethod	-1=median, -2=mode, >=0=fixed threshold

2.4.1 Parameter Selection

To explore the behavior of each filter across different datasets, we first conducted a manual visual tuning phase. This provided insight into which parameter settings produced plausible results and helped define sensible parameter ranges. We then applied a data-driven parameter search to select the optimal settings for each filter. Specifically, we performed a grid search with cross-validation, which is justified here due to the low dimensionality of the parameter space and the relatively small number of candidate values per method. For each configuration within the predefined parameter ranges , the filter response was dilated (Step 2 in preprocessing), thresholded and evaluated against the refined ground truth using the F1 score (definition in Table 5). The parameter set yielding the highest score was selected as optimal. This grid search standardises filter hyperparameters across datasets and reduces subjectivity in parameter selection.

2.5 Deep Learning-Based Models

Two deep learning architectures are evaluated in this work: a convolutional U-Net (Ronneberger et al., 2015) and a transformer-based SegFormer (Xie et al., 2021) initialised with pretrained ResNet34 weights as the backbone. U-Net is widely used for

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

semantic segmentation due to its encoder–decoder architecture and skip connections, while SegFormer leverages attention mechanisms combined with the ResNet34 backbone for long-range context modelling and robustness to geometric variability.

Both models were trained with architecture-specific hyperparameters selected by cross-validation and sensitivity tests. For the U-Net, we used the Adam optimiser with an initial learning rate of 0.05, while the SegFormer employed RMSProp with an initial learning rate of 0.01. The input to each model consisted of 256 × 256 px patches with four channels (three RGB bands and one DEM band). We used Huber (smooth L1) loss for per-pixel probability targets because it proved robust to label noise, spatial misalignment, and class imbalance compared with binary cross-entropy or Dice loss (popular for segmentation tasks). Early stopping based on validation loss was used in both cases to prevent overfitting.

The U-Net and SegFormer architectures are depicted in Figure 5 and Figure 6, respectively.

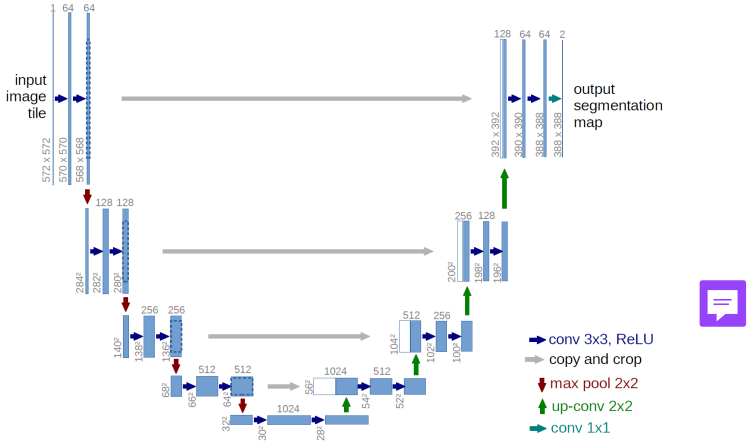

Figure 5. Overview of the U-Net encoder–decoder architecture with skip connections (Ronneberger et al., 2015)

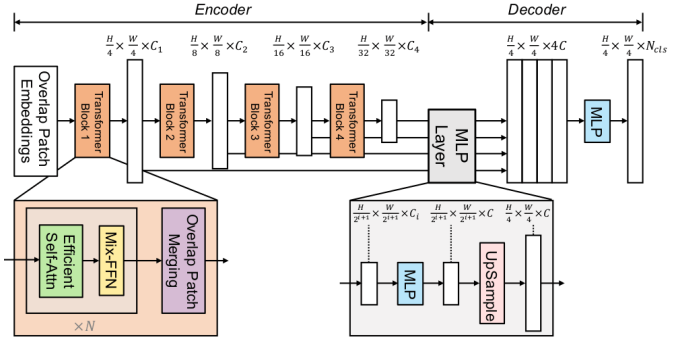

Figure 6. Overview of the SegFormer architecture combining a transformer encoder with a lightweight decoder (Xie et al., 2021)

Both models produce per-pixel probability maps used for visual comparison and were thresholded to generate binary segmentation masks, which were then compared to the ground truth using the metrics described below.

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

2.6 Benchmark metrics

To evaluate segmentation performance, we used three complementary groups of metrics:

1. image-quality metrics that quantify how closely the predicted probability maps match the annotated targets,
2. pixel-wise classification and segmentation metrics that measure detection performance and spatial overlap, and
3. a proposed fracture-level similarity metric, FracSim, designed to compare the geometry of predicted fracture traces with those in the ground truth.

Together, these metrics (Table 5) capture prediction quality at the pixel, mask, and structural levels. Higher values generally indicate better performance, except for error metrics (MSE, AE), where lower values are preferable.

We include the background class in our evaluation to capture each model's full performance across the entire image. Because fractures represent only a very small fraction of pixels, excluding the background can artificially inflate metrics such as precision and recall. It also tends to make poorly calibrated models appear better, as ignoring the dominant class often results in models overpredicting fractures. Including the background therefore provides a more realistic assessment of false positives, calibration, and overall segmentation integrity. Notably, many previous studies do not explicitly report whether the background class is included in the computation of evaluation metrics.

Among the various metrics computed, we suggest that the Dice coefficient (equivalent to the F1 score), recall, specificity, and intersection over union (IoU) are the most relevant for the fracture segmentation task. In contrast, accuracy is the least informative metric here, as a model that predicts only background can achieve deceptively high scores due to class imbalance. Dice and IoU measure spatial overlap, which is critical for evaluating thin, linear features like fractures. Dice balances false positives and false negatives, while IoU penalises both over- and under-segmentation, making it a stricter metric. Recall ensures that fractures are not missed, while specificity quantifies how well the model avoids false positives in the dominant background class. Given the extreme class imbalance—where background pixels can account for over 99% of the image—reporting recall alone may be misleading. Specificity completes the picture by measuring the number of false detections in non-fracture regions.

We also include ROC-AUC, which evaluates the model's ability to distinguish between fracture and background pixels across thresholds. As a threshold-independent metric, ROC-AUC is particularly useful for probabilistic outputs. Similarly, Cohen's Kappa measures agreement between prediction and ground truth while accounting for chance, offering a more meaningful alternative to raw accuracy in imbalanced settings.

In addition to segmentation metrics, we report image-quality metrics—Mean Squared Error (MSE), Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM), and Absolute Error (AE)—to assess how well the predicted probability maps match the annotations before thresholding. These metrics capture prediction fidelity in both intensity and structure. MSE and AE quantify raw pixel-wise error (lower is better), while PSNR measures signal degradation in a logarithmic scale (higher is better). SSIM accounts for human visual perception by evaluating structural and contrast similarity, which is especially valuable when preserving subtle linear patterns like fractures.

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

Table 5. Overview of metrics for evaluating image quality and segmentation

Metric	Formula	Definition / Use
MSE (Mean Squared Error)	$\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$	Average squared pixel error; lower is better.
PSNR (Peak Signal-to-Noise Ratio)	$10\log_{10}\left(\frac{L^2}{\text{MSE}}\right)$	Log ratio of max intensity L to error; higher is better.
SSIM (Structural Similarity Index)	$\frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$	Perceptual similarity using luminance, contrast, and structure.
AE (Absolute Error)	$\frac{1}{N}\sum_{i=1}^{N}\lvert y_i - \hat{y}_i\rvert$	Absolute pixel-wise error; lower is better.
Accuracy (Acc)	$\frac{TP+TN}{TP+TN+FP+FN}$	Correctly classified proportion; sensitive to imbalance.
Precision (Prec)	$\frac{TP}{TP+FP}$	Correctness of predicted positives.
Recall / Sensitivity (Rec)	$\frac{TP}{TP+FN}$	Fraction of true positives detected.
Specificity (Spec)	$\frac{TN}{TN+FP}$	Fraction of negatives correctly identified.
F1 Score	$\frac{2\,\text{Prec}\cdot\text{Rec}}{\text{Prec}+\text{Rec}}$	Harmonic mean of precision and recall.
Dice Coefficient	$\frac{2\lvert A\cap B\rvert}{\lvert A\rvert+\lvert B\rvert}$	Spatial overlap measure; equivalent to F1 in binary segmentation.
IoU (Intersection over Union)	$\frac{\lvert A\cap B\rvert}{\lvert A\cup B\rvert}$	Mask overlap ratio.
AUC_ROC	–	Area under ROC curve; reflects the probability that a model assigns a higher score to a positive instance than to a negative instance.
Average Precision (AP)	$\int_0^1 \text{Prec}(r)\,dr$	Area under precision–recall curve.
Cohen's Kappa (CK)	$\kappa = \frac{p_o - p_e}{1 - p_e}$	Agreement corrected for chance.

Symbols: N = number of pixels; y_i = reference pixel value; $\hat{y}_i$ = predicted pixel value; L = maximum possible pixel intensity; μ_x, μ_y = means of image patches x and y; σ_x^2, σ_y^2 = variances; σ_{xy} = covariance; C_1, C_2 = SSIM stability constants; TP, TN, FP, FN = true/false positives/negatives; A, B = predicted and reference binary masks; r = recall; p_o = observed agreement; p_e = expected chance agreement; ROC: receiver operating characteristic, is a graphical plot which illustrates the performance of a binary classifier system as its discrimination threshold is varied.

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

The different scenarios in Figure 7 highlight how each metric responds to specific segmentation behaviors. Predicting only background (no segmentation) yields high accuracy and specificity but zero recall, precision, and overlap-based scores, illustrating how class imbalance can inflate accuracy despite complete failure to detect fractures. Conversely, full segmentation maximises recall while collapsing specificity and precision, resulting in low Dice and Intersection over Union values. Random segmentation produces near-chance values across most metrics, with Dice, Intersection over Union, Cohen's kappa, and receiver operating characteristic scores reflecting the lack of spatial agreement. The Gabor filter segmentation shows intermediate performance, improving overlap, precision, and specificity by capturing some fracture structure, but still suffering from fragmentation and false positives. The U-Net prediction achieves the most balanced and consistently high scores across all metrics, with strong overlap, agreement, and classification performance, closely approaching the ground truth.

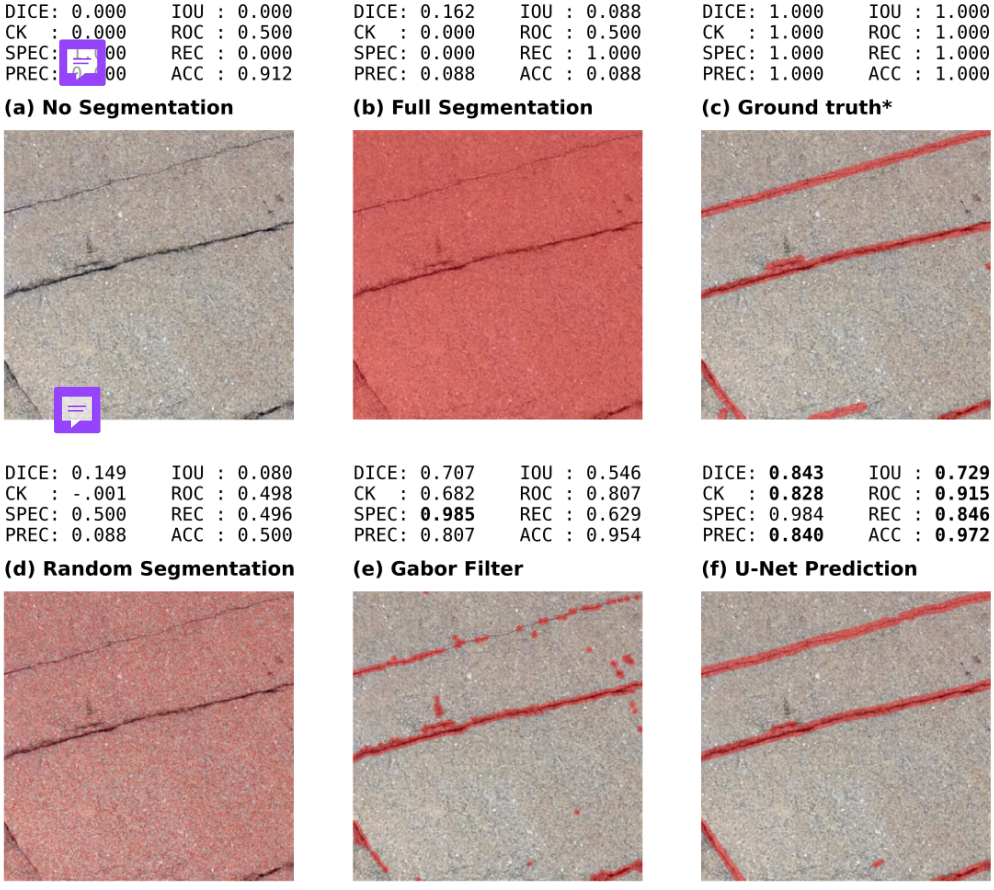

Figure 7. Illustration of metric behavior under different segmentation scenarios: (a) No segmentation (predicts all background), (b) Full segmentation (predicts all fracture), (c) Ground truth, (d) Random segmentation, (e) Gabor filter segmentation, (f) U-Net segmentation. Values in bold showing highest values discarding the first row.

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

2.7 *FracSim*: a fracture-level similarity metric

Because one of the key characteristics relevant to fracture mapping is fracture length, we developed a domain-specific similarity metric, termed FracSim, which compares the distribution of extracted fracture segment lengths between a predicted mask and the ground truth.

For each prediction and label, we:

(1) convert the probability map to a binary mask (user threshold, default: 0.1);

(2) skeletonise the binary masks to a one-pixel wide trace;

(3) remove short noisy junctions using a small neighborhood filter (to avoid counting cross-overs and thick junction blobs as segments);

(4) label connected skeletal segments and compute their pixel lengths;

(5) construct comparable histograms of segment lengths (same bin edges for prediction and ground truth); and

(6) compute a symmetric chi-square distance between the two histograms (Equation 1).

$$d_{\chi^2}(h,g) = \frac{1}{2}\sum_{i=1}^{N}\frac{(h_i - g_i)^2}{h_i + g_i} \tag{1}$$

where

- $d_{\chi^2}(h,g)$ is the symmetric chi-square distance between the two histograms,
- h and g are the two histograms being compared,
- h_i and g_i denote the values of the i-th bin of histograms h and g, respectively,
- N is the total number of bins.

Lower χ^2 indicates closer agreement in the fracture length distribution. We compute *FracSim* only on patches that contain a high fracture-pixel count in the ground truth; empty or nearly empty patches are excluded because they lack a meaningful fracture-length distribution to compare, and even small prediction artifacts in these patches can otherwise produce misleading metric values.

Together, these metrics (§2.6 and §2.7) provide a balanced framework for evaluating both hard segmentation performance and the quality of soft probabilistic outputs.

2.8 Generalisation experiments

To evaluate the generalisation capability of a single model trained across diverse geological settings, we compared a multi-site model (trained on all datasets combined, M_all) against individual models trained separately on each dataset (M_ovas, M_sams, and M_matt). All models employed an identical architecture and training configuration, with no dataset-specific hyperparameter tuning conducted. Each dataset-specific model was evaluated only on its corresponding test set, while M_all

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

was tested independently on each dataset. This experiment aims to assess whether a generalist model can match or exceed the performance of models trained under dataset-specific conditions, offering insights into the feasibility of cross-domain fracture segmentation.

2.9 Deployment

As a first step toward practical deployment, we developed a lightweight web-based prototype that enables users to upload paired RGB images and DEMs and obtain deep-learning–based fracture predictions via the hassle-free Gradio interface (Abid et al., 2019). The tool, hosted on Hugging Face Spaces https://huggingface.co/spaces/ayoubft/fractex2D_tuto, illustrates how the trained model can be made accessible beyond a purely research-oriented setting. While still preliminary, this prototype highlights the potential of interactive platforms to broaden the adoption of automated fracture detection within the geoscience community and offers a concrete example of integrating deep-learning outputs into field and laboratory workflows. The interface additionally allows users to explore a range of computer-vision filters and metrics calculation.

3 Results

To assess model performance, we evaluated both traditional computer vision (CV) filters and deep learning (DL) models on the test dataset, using the image quality, segmentation and similarity metrics described earlier. The following sections present these results, qualitatively and quantitatively.

3.1 Traditional Computer Vision–Based Algorithms

The optimal hyperparameters and thresholds that produced the best results, identified through grid search and manual visual inspection, are listed in Table C1.

3.1.1 Quantitative Results

For completeness, the full numerical results from both the grid-searched settings are provided in Table D1. The radar chart in Figure 8 summarises how the traditional filters perform across image quality, segmentation, and similarity metrics. Overall, the filters cluster near the center of the plot, indicating uniformly limited fracture-detection ability. Canny, Sobel, Sato, and Phase Congruency exhibit similar profiles, with modest accuracy, precision, F1, and IoU, and only small variations in recall, specificity, and ROC. For the structural similarity metric, they achieve an SSIM value of zero, indicating no structural similarity. In contrast, the error-based measures (MSE, AE, and loss) remain low, which is favorable. Gabor is the only method that diverges sharply from this pattern: it attains perfect recall but extremely low precision, yielding a highly unbalanced shape on the radar plot. Because it labels nearly all pixels as background class, the remaining metrics collapse to very low values.

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

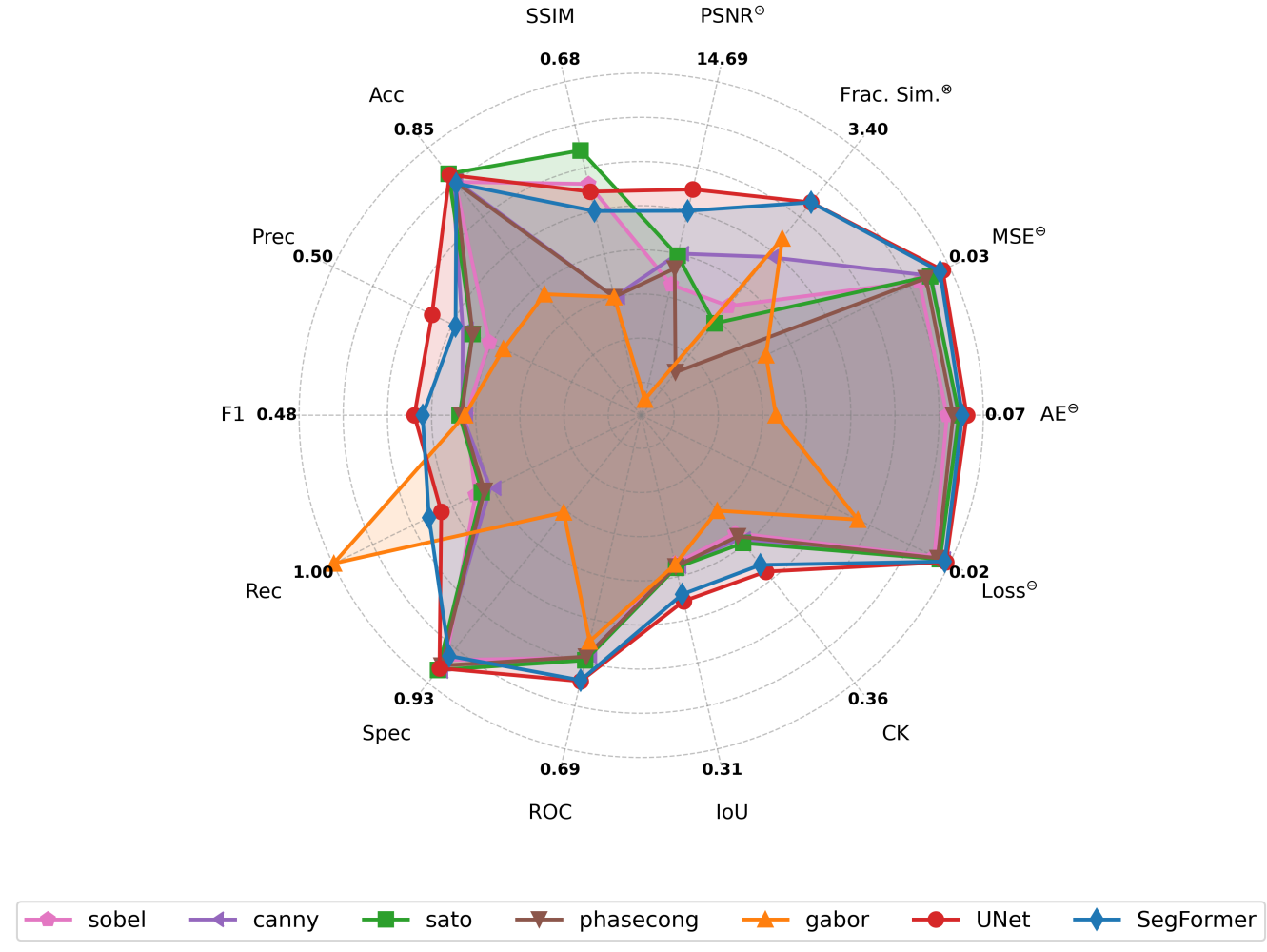

Figure 8. Radar chart summarizing the performance of the classical filters across all evaluation metrics. All metrics originally lie between 0 and 1, except PSNR$^{\odot}$ and FracSim$^{\otimes}$. And metrics where lower values indicate better performance (AE$^{\ominus}$, MSE$^{\ominus}$, Loss$^{\ominus}$ and FracSim$^{\otimes}$) were flipped by computing (1 – value) so that the chart consistently displays better performance toward the outer radius. FracSim$^{\otimes}$ was calculated on the modified test set.

3.1.2 Qualitative Results

Using grid-searched parameters across all test sites, the classical filters produced somewhat useful but generally sparse outputs (Figure 9). Phase Congruency and Canny often highlighted many true fracture segments, but their responses included substantial noise and many thin lines. Sobel and Sato tended to be very sparse, picking up only the strongest edges or ridge cores, while Gabor largely failed to isolate fractures and instead responded to texture, producing noisy, non-fracture patterns. In short,

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

the grid search showed that some filters can detect fracture signals, but the universal parameter sets are not optimal for every site and yield noisy or incomplete maps.

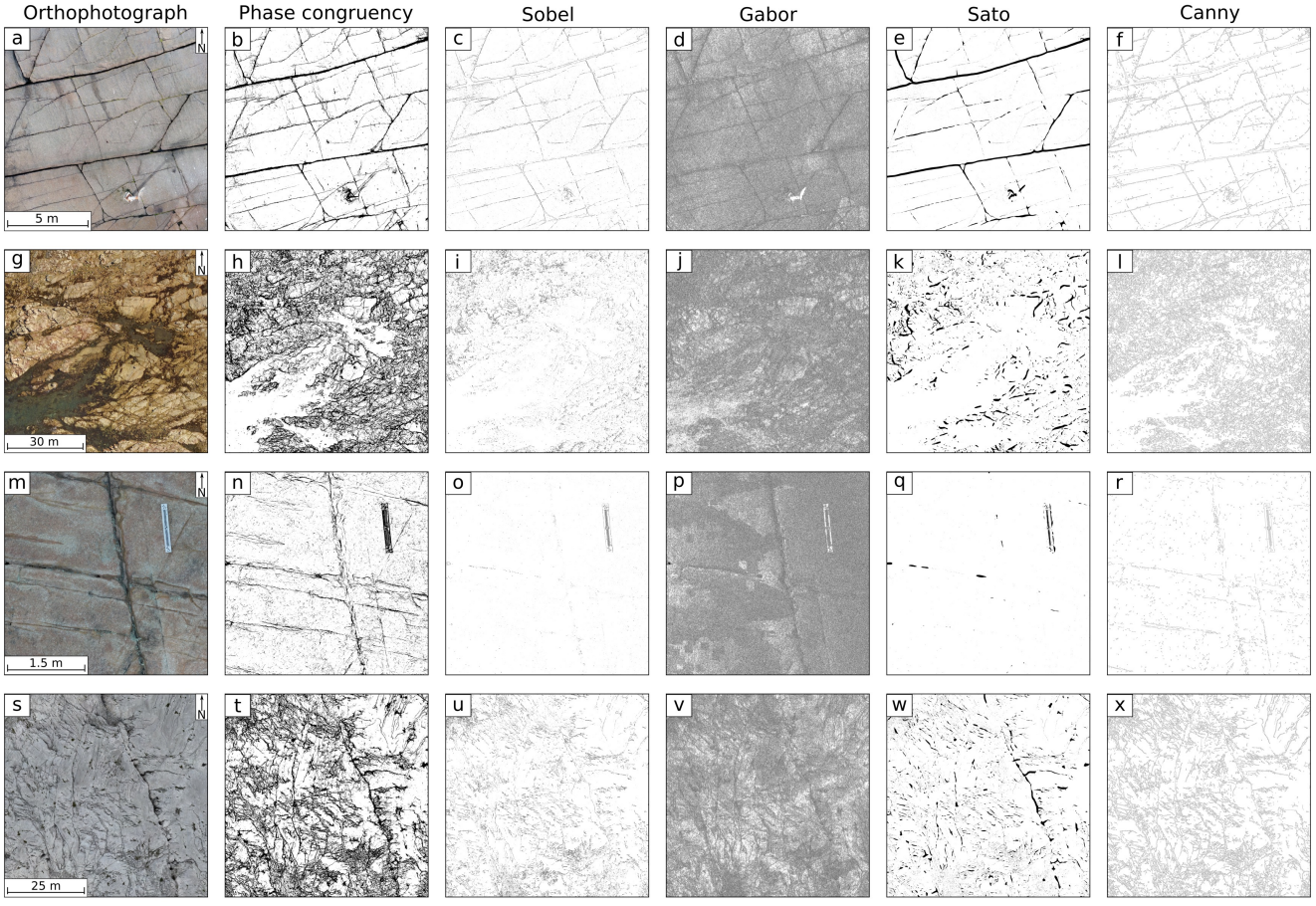

Figure 9. Comparison of traditional filters applied to four representative orthophotographs, using parameters selected by grid search. Each row shows an original orthophotograph (a, g, m, s) alongside results from Phase Congruency (b, h, n, t), Sobel (c, i, o, u), Gabor (d, j, p, v), Sato (e, k, q, w), and Canny (f, l, r, x) filters.

Using manually tuned parameters on each test site separately, some filters were able to highlight fracture segments more clearly (Figure 10). Sobel and Canny captured more pronounced edges along major traces, while Sato produced thicker and more continuous lines. However, the outputs also included numerous spurious detections of non-fracture features.

While manually tuned parameters can produce visually plausible outputs on individual test sites, this performance does not generalise well across other locations—or even across different regions within the same image. Traditional filters are sensitive to local contrast, scale, and texture, requiring site-specific adjustments to capture relevant features. These tuned parameters, although effective in isolated cases, often fail when applied to heterogeneous datasets.

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

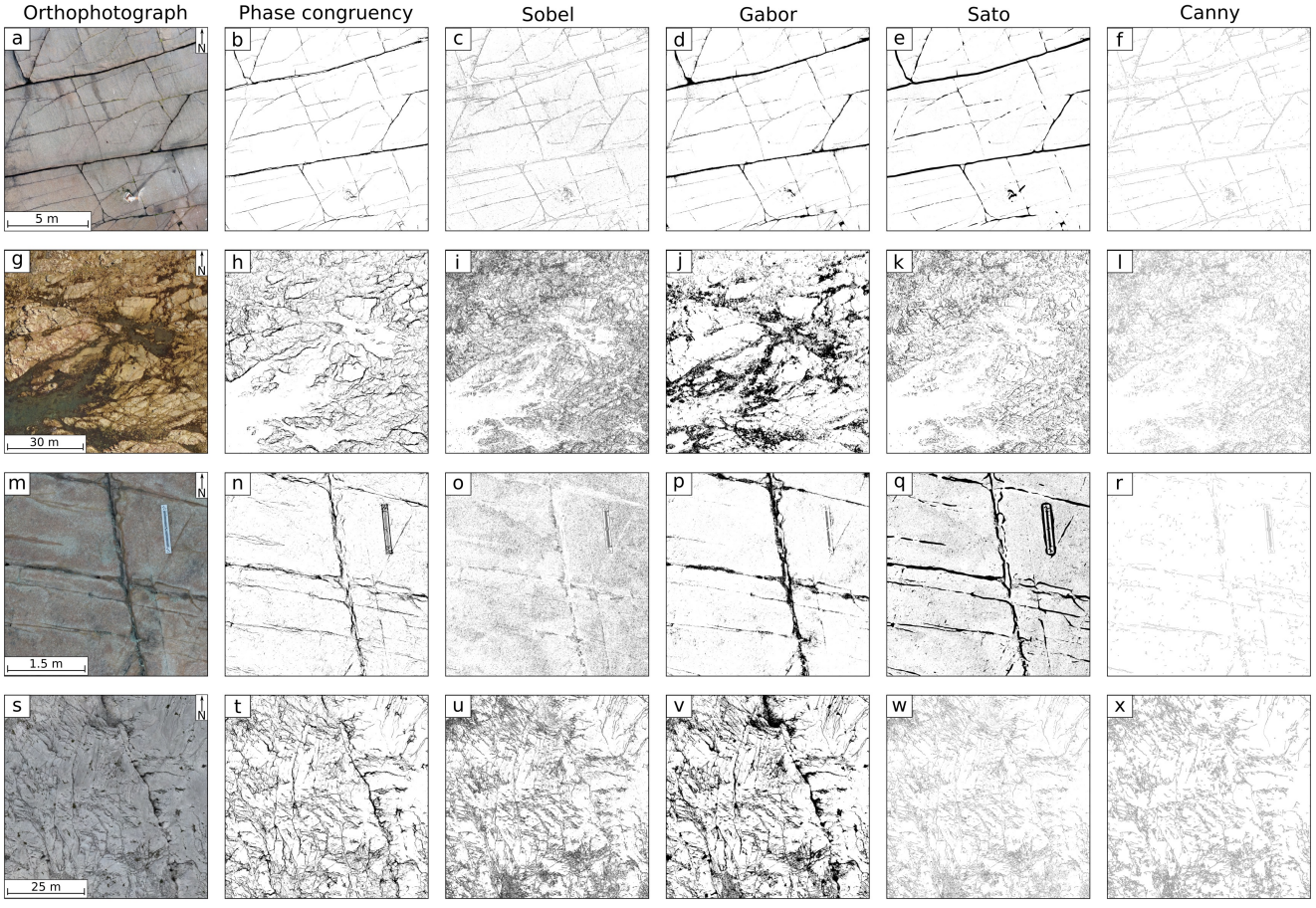

Figure 10. Comparison of traditional filters applied to four representative orthophotographs, using parameters selected manually for each site. Each row shows an original orthophotograph (a, g, m, s) alongside results from Phase Congruency (b, h, n, t), Sobel (c, i, o, u), Gabor (d, j, p, v), Sato (e, k, q, w), and Canny (f, l, r, x) filters.

The limitations described above expose a core issue: both fixed-rule and parameter-tuned methods on traditional filters lack the flexibility to handle the variability of complex, real-world geology. Thus, they motivate the adoption of deep learning approaches, which do not rely on hand-crafted features or globally fixed parameters. Instead, they learn hierarchical representations directly from the data, enabling better adaptation to diverse fracture patterns and geological contexts. Deep networks can, in principle, generalise beyond specific sites by capturing shared structures across datasets.

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

3.2 Deep Learning Models

3.2.1 Training Dynamics

For model and hyperparameter selection, cross-validation was performed using the training and validation subsets. Figure 11 shows a representative training run for both architectures using the fixed hyperparameters mentioned in the Methods section and a single random seed (42), which ensures that weight initialization and any other stochastic processes are reproducible and consistent across runs. The U-Net exhibits a smooth decrease in training loss and a validation curve that stabilises around epoch 43. The gap between the two curves remains small, with only mild overfitting toward the end of training. The checkpoint with the lowest validation loss was used for evaluation. SegFormer follows a similar downward trend but with higher loss values and more fluctuation, particularly during the first half of training. Its slower convergence suggests that the model adapts less effectively to this dataset than the U-Net, or alternatively, that it requires additional training data and longer training to achieve a comparable fit.

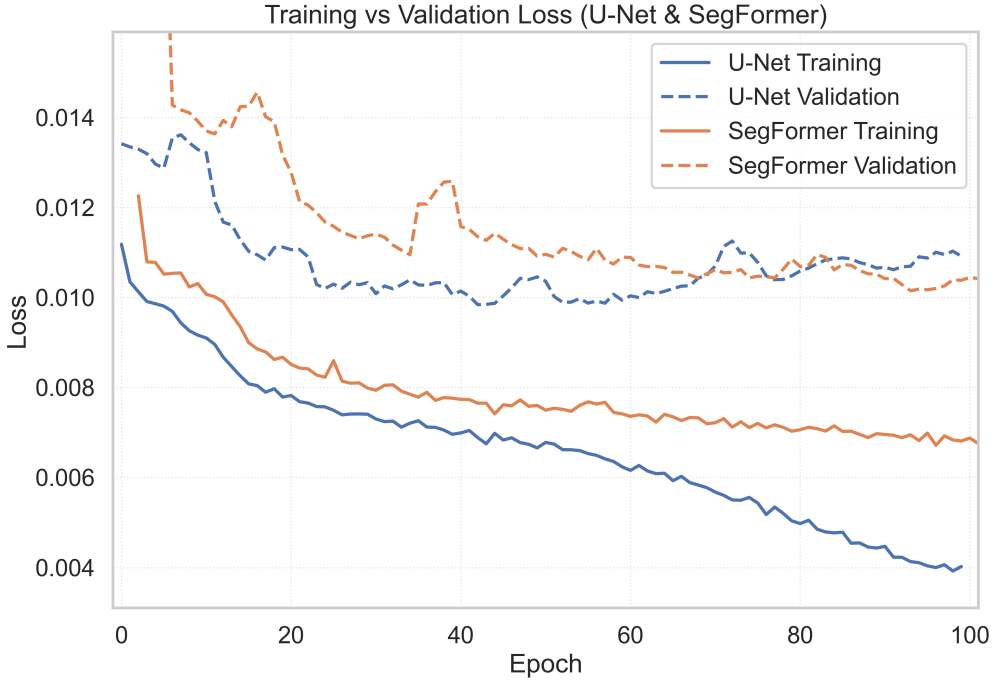

Figure 11. Training and validation loss curves for U-Net and SegFormer (range: [0, 1])

3.2.2 Quantitative Results

As shown in the spider diagram (Figure 8) and in Table D1, the U-Net model achieved a mean F1-score of 0.48 and IoU of 0.31, substantially higher than any traditional baseline (F1 < 0.29, IoU < 0.17). Precision reached 0.50, indicating that half of

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

the predicted pixels presenting fractures corresponded to true positives, while recall remained moderate (0.45), showing that some pixels of the fractures were missed. SSIM (0.49) and PSNR (14.7 dB) confirm that the reconstructed fracture masks deviate notably from the ground truth in terms of image similarity, but the segmentation metrics demonstrate reliable detection of fracture structures.

The SegFormer model was trained using the same data splits and evaluation protocol as the U-Net. While SegFormer produced competitive results, its F1 (0.44) and IoU (0.28) were consistently lower than those of the U-Net. Standard deviations across runs were similarly low, indicating stable but suboptimal convergence. Although the U-Net and SegFormer models clearly outperform filter-based methods, their absolute performance remains well below standard deep learning studies (F1 > 0.8, IoU > 0.6) (Guo, 2023; Kuş and Aydin, 2024; Sumi et al., 2024), emphasising the challenging nature of this task and relatively limited available training data.

Notably, our experiments show that the DL (U-Net and SegFormer) models achieve substantially lower FracSim (better agreement of length distributions) than classical filters. This demonstrates that DL predictions not only overlap ground truth spatially (IoU, F1) but also better reproduce fracture segment length statistics.

3.2.3 Qualitative Results

Example outputs are shown in Figure 12. Across the different datasets, the U-Net CNN consistently produced fracture maps that were smoother, more continuous, and closer to the ground-truth annotations than those of traditional filters. The model often reconnected segments that filters left broken, yielding fracture networks that were both cleaner and more interpretable. These improvements were observed even under varying lithologies and illumination conditions, confirming the CNN's ability to generalise across a wide range of datasets. Remaining errors included omission of faint fractures and occasional short false positives in textured or shadowed areas. These qualitative results reinforce the quantitative findings: deep learning methods provide more reliable fracture maps than traditional filters, with notable improvements in continuity and interpretability across diverse geological settings.

When comparing the two deep learning models, the SegFormer model generated continuous predictions (Figure 12), in some cases capturing long-range structures more effectively than the CNN. However, it introduced more false positives in complex textures, occasionally produced random hot spots, and overall its outputs were less confident and less precise compared to the U-Net model across the test sites.

3.2.4 General vs. Dataset-Specific Models

The comparison of models within each site (Table 6) shows that the combined model (M_all) provides clear advantages over dataset-specific models, particularly in terms of F1 and IoU.

For the SegFormer architecture on the D_ovas dataset, M_all and M_ovas exhibit comparable performance. M_all achieves an F1-score of 0.70 and an IoU of 0.54, closely matching M_ovas (F1 = 0.67, IoU = 0.51). This indicates that training on multiple datasets does not degrade performance on the Ovaskainen dataset, where fractures are relatively clear and annotations are well defined.

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

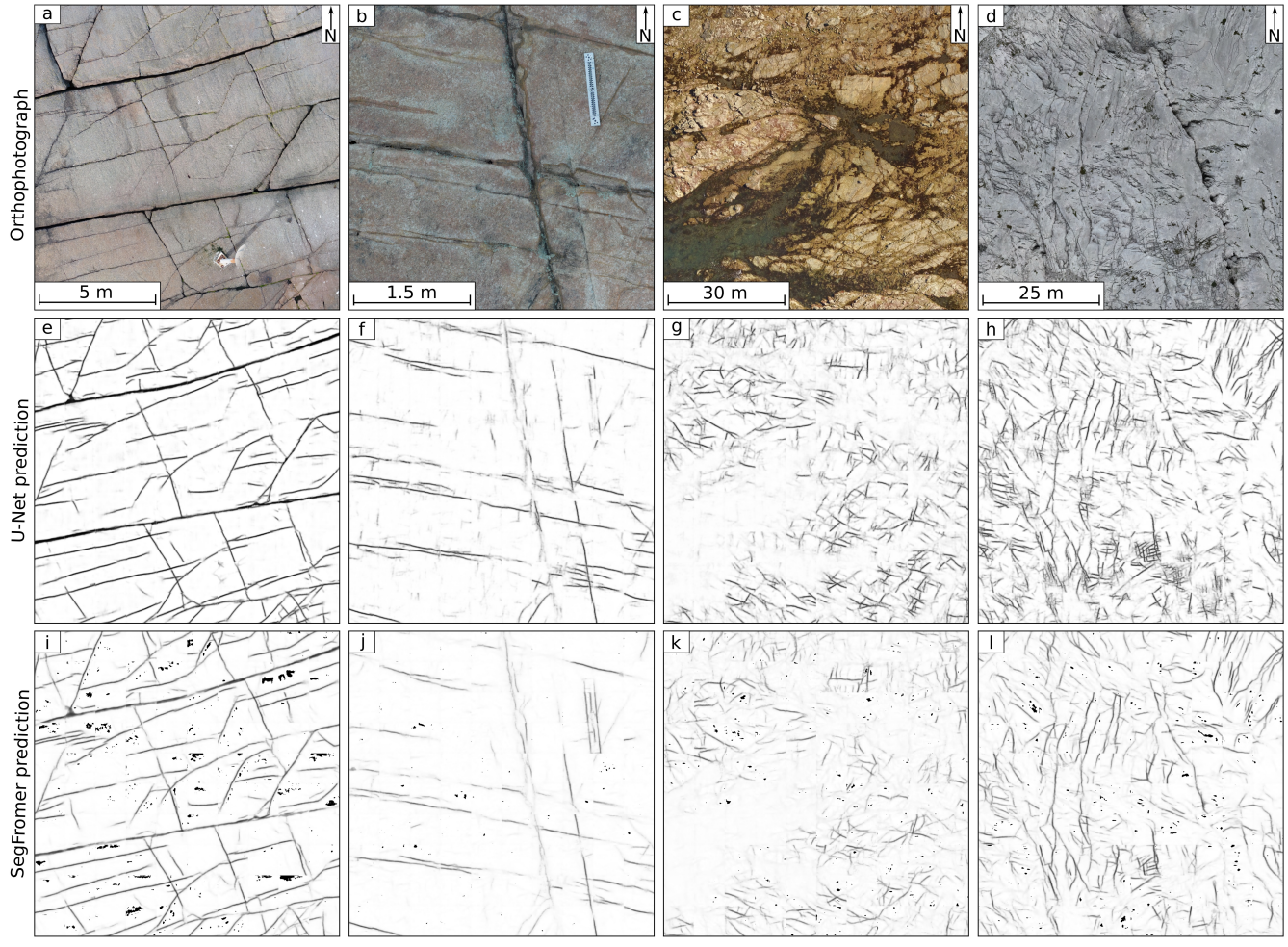

Figure 12. Deep-learning fracture-trace predictions for four representative orthophotographs. Panels (a–d) show the original orthophotographs. Panels (e–h) present corresponding U-Net predictions, and panels (i–l) show SegFormer predictions.

The advantages of M_all become more pronounced on the more challenging sites. On D_sams, the site-specific model (M_sams) performs poorly, achieving very low scores (F1 = 0.11, IoU = 0.06 for U-Net, and zero for SegFormer), despite high specificity, indicating weak fracture identification relative to the ground truth. In contrast, M_all substantially improves performance, reaching F1-scores of 0.34/0.33 and IoU values of 0.21/0.20. This represents more than a twofold increase in overlap metrics, suggesting that the increased variability in the combined training set enables better generalization to the more complex fracture patterns present in the Samsu19 dataset.

A similar trend is observed for D_matt. The site-specific model M_matt attains only F1 = 0.24 and IoU = 0.14 with U-Net, and F1 = 0.06 and IoU = 0.03 with SegFormer. In contrast, M_all improves performance to F1 = 0.39/0.41 and IoU = 0.24/0.25.

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

Table 6. Comparison between multi-site model (M_all) and dataset-specific models (M_ovas, M_sams, M_matt)

Arch	Sub-dataset	Model	AE↓	MSE↓	SSIM↑	PSNR ↑	Acc↑	Prec ↑	F1 ↑	Rec ↑	Spec ↑	ROC ↑	IoU ↑	CK ↑	FracSim∗ ↓	Loss ↓
U-Net	D_ovas	M_all	**0.05**	0.02	**0.64**	**16.51**	**0.93**	**0.77**	**0.73**	0.68	**0.97**	0.83	**0.57**	**0.69**	**2.99**	0.01
		M_ovas	0.06	0.02	0.60	16.03	0.91	0.63	0.67	**0.72**	0.93	0.83	0.51	0.62	3.04	0.01
	D_sams	M_all	0.06	0.03	**0.58**	15.48	0.86	0.32	**0.34**	**0.38**	0.92	**0.65**	**0.21**	**0.27**	7.54	0.01
		M_sams	**0.05**	0.03	0.57	**15.99**	**0.91**	**0.58**	0.11	0.06	**1.00**	0.53	0.06	0.09	**3.26**	0.01
	D_matt	M_all	0.08	0.04	0.57	14.27	0.80	0.45	**0.39**	**0.34**	0.90	**0.62**	**0.24**	**0.27**	**4.01**	0.02
		M_matt	**0.06**	**0.02**	**0.61**	**16.53**	**0.81**	**0.47**	0.24	0.17	**0.96**	0.56	0.14	0.16	4.78	**0.01**
SegFormer	D_ovas	M_all	0.06	0.02	0.53	**16.34**	0.93	0.77	0.70	**0.64**	0.97	**0.80**	0.54	0.66	**2.94**	0.01
		M_ovas	**0.05**	0.02	**0.58**	16.12	0.93	**0.81**	0.70	0.61	**0.98**	0.79	0.54	0.66	3.16	0.01
	D_sams	M_all	0.06	0.03	**0.39**	**15.96**	0.85	**0.29**	**0.33**	**0.39**	0.90	**0.65**	**0.20**	**0.25**	7.85	0.01
		M_sams	**0.05**	0.03	0.35	15.83	**0.91**	0.00	0.00	0.00	**1.00**	0.50	0.00	-0.00	**1.65**	0.01
	D_matt	M_all	0.08	0.03	0.38	15.37	0.81	0.48	**0.41**	**0.35**	0.91	**0.63**	**0.25**	**0.29**	**4.47**	0.01
		M_matt	**0.06**	0.03	**0.67**	**16.01**	0.81	**0.51**	0.06	0.03	**0.99**	0.51	0.03	0.04	5.55	0.01

∗ run only on modified test set

This near doubling of overlap metrics demonstrates that M_all is considerably better suited to handling the lithological and textural variability of the Matteo21 dataset.

Overall, these results indicate that while dataset-specific models can capture local patterns, they are constrained by limited training data and are prone to overfitting, resulting in reduced robustness.

3.2.5 Error Case Analysis

We note that despite its stronger overall performance, the U-Net still produced errors (Figure 13). These fall into two categories: (1) cases where the model was highly confident yet disagreed with the annotations, and (2) cases where the model was uncertain, assigning low probabilities and yielding fragmented or unstable predictions under challenging visual conditions.

High-confidence errors. In several cases, the CNN produced confident predictions that did not match the annotations. A common source was annotation misalignment, where fractures in the ground truth were traced coarsely or offset from the true pixel locations. In these cases, the CNN model is arguably more accurate than the ground truth. This misalignment was particularly evident in the Samsu19 dataset, where manual traces did not precisely follow pixel-level fracture geometry. Another high-confidence error type was input data artifacts, especially in Matteo21, where corrupted pixel values (NaNs which PyTorch converts to 0, rendering them black in RGB) in the RGB channels led to localised false positives or spurious losses unrelated to actual geological structures.

Low-confidence errors. Other errors occurred when the model was less certain and produced scattered or incomplete predictions. Confounding visual features such as vegetation, weathering textures, or shadows were occasionally misclassified as fractures, but with lower confidence values in the probability maps. Additionally, dataset outliers and preprocessing failures

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

(e.g., unusual lithologies, lighting conditions, or mislabelled patches) caused the model to generate unstable or inconsistent outputs. These low-confidence cases reflected the difficulty of generalizing across heterogeneous field conditions not fully represented in training data.

Taken together, this analysis suggests that many errors stemmed not from the CNN architecture itself but from limitations in the data: annotation quality, artifacts in inputs, and domain heterogeneity. Probability maps proved useful here, as they allowed us to distinguish between errors where the model was confident but the labels were unreliable, and those where the model itself was uncertain.

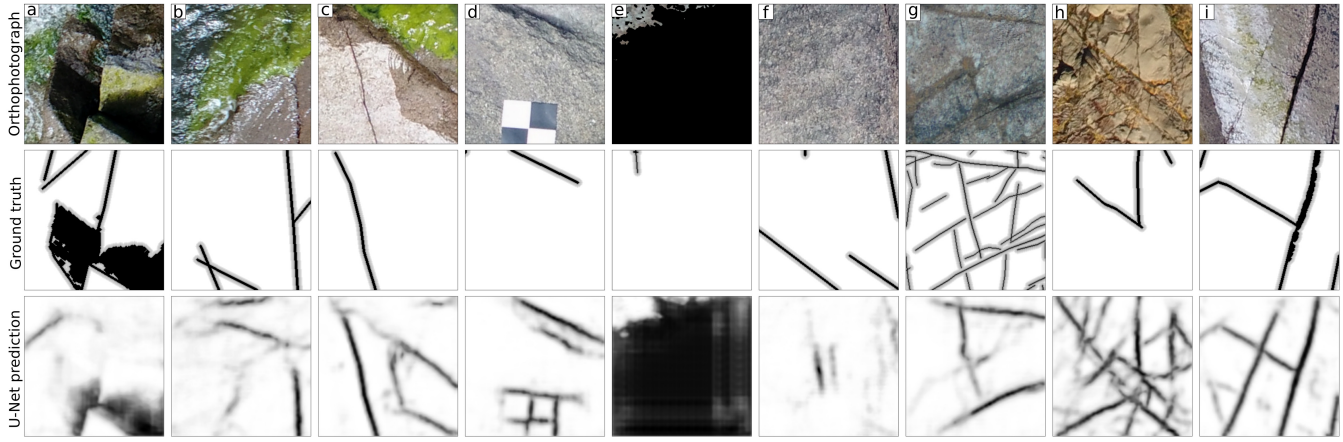

Figure 13. Error case analysis of U-Net predictions

Top row: Orthophotographs (a–c) with water and large shadows, (d) with a black-and-white scale panel, (e) containing missing pixel data, (f) showing no visible fractures, and (g–i) displaying fractures of varying thickness, orientation, and illumination

Middle row: Ground-truth annotations. Some overrepresent shadows due to preprocessing (a), contain labelling errors (c), have over-simplified traces (h), or show omissions (i)

Bottom row: U-Net probability maps. Faux responses appear in (a–e) due to input artifacts or preprocessing issues, while the model captures more realistic fracture geometry than the annotations in (f–i)

3.2.6 Time Required

We compared the time required for fracture mapping across manual, assisted, and automatic approaches (Figure 14). Timing data for manual and assisted interpretation are taken from Thiele et al. (2017b), where mapping fractures in a 10 × 10 m area at 1 cm/pixel resolution took 54–57 minutes manually and 35–37 minutes using semi-automatic tools. In contrast, our automatic deep learning approach generated probability maps in 6 seconds (on free tier huggingface space (2vCPU and 16GB RAM)). While post-processing is still required to derive a fracture map (instance segmentation and vectorisation), this speed-up could enable a substantial reduction in effort and time during large-scale fracture interpretation.

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

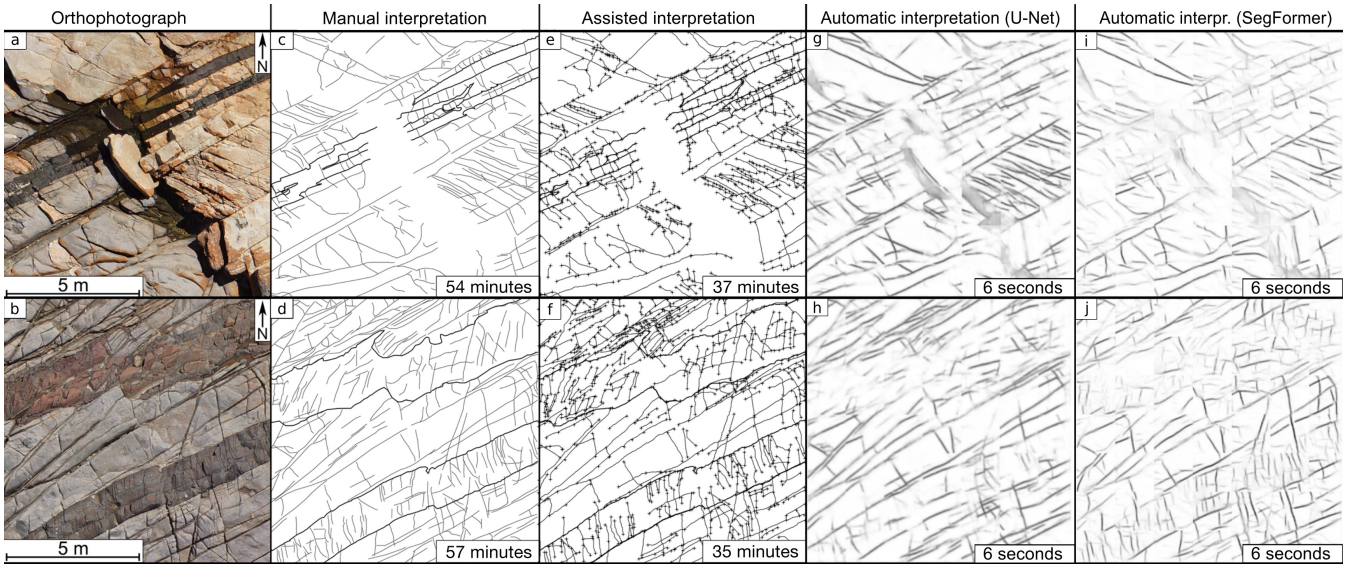

Figure 14. Two 10 × 10 m orthophotographs (Thiele et al., 2017b) (a, b) with corresponding fracture traces digitised manually (Thiele et al., 2017b) (c, d), semi-automatically using the assisted method of (Thiele et al., 2017b) (e, f), automatically with the U-Net model (g, h) and with SegFormer model (i, j)

4 Discussion

We now discuss key observations from the benchmarking experiments and their implications for automated fracture mapping.

4.1 Model Performance

Unsurprisingly, traditional edge and ridge filters proved inadequate for robust fracture mapping across heterogeneous datasets. Their outputs were highly sensitive to parameter choices, and hyperparameter selections often failed to generalise well across combined sites. In practice, these traditional methods remain useful for rapid visualisation or exploratory work, or tailored application to relatively small datasets. However, they are ill-suited for mapping of complex fracture systems, which is required for fracture and structural analyses.

The tested deep learning models were shown to significantly outperform the traditional filters, producing smoother and more continuous traces and substantially better overlap with annotated fractures. Nevertheless, absolute performance remained modest compared with segmentation benchmarks in other domains. This is likely due to the highly unusual nature of this semantic segmentation task - the selection of thin (high-aspect-ratio) pixel groupings (edges) in a highly class-imbalanced dataset - and the variety of geological and environmental factors that change fracture appearance or introduce obfuscating features. Label noise and misalignment are also likely contributors, although practical limitations around data collection and manual labelling make it challenging to acquire better training data.

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

Interestingly, the transformer-based models (SegFormer) behaved quite differently from the CNNs in our experiments (see subsection 3.2. Their results were less robust, producing spurious detections that we attribute to the tendency of global-attention mechanisms to over-interpret background structure when the training signal is weak. This is consistent with known properties of transformers: they can excel when supplied with large, diverse, and clean datasets, but are more data-hungry than convolutional architectures.

Thus, while transformers showed some promise, their advantages are likely to emerge only with substantial improvements in dataset size and annotation quality—although practical limitations around data collection and manual labelling make acquiring such high-quality training data particularly challenging in the context of geological fractures. In the meantime, the tested CNN (U-Net) architecture appears most promising.

Our cross-dataset experiments show that diverse training data improves model performance. A general model trained on all sites consistently outperformed models trained on individual datasets, especially on overlap- and detection-sensitive metrics like F1 score and IoU. This suggests that exposure to varied lithologies, lighting, and fracture patterns helps the model learn more transferable features and reduces overfitting to site-specific conditions. These results highlight the importance of multi-site datasets for robust fracture segmentation.

4.2 The Need for Post-processing

A critical bottleneck in automated fracture mapping is post-processing, which is needed to convert model outputs to polylines or some other vectorised representation of fracture instances that are better suited for topological analyses (e.g., for assessing fracture connectivity). Pixel-wise probability maps are valuable, but they are only the first step toward usable fracture traces. Converting semantic masks into instance-level polylines for length, orientation and spacing measurements requires robust, standardised post-processing pipelines (skeletonisation, connected-component analysis, cleaning, linking, and polyline extraction). Without these steps, downstream fracture statistics remain noisy and difficult to interpret.

Post-processing can and should also incorporate domain knowledge (for example, weak orientation priors) to reduce spurious short segments and improve the geological plausibility of extracted traces. Furthermore, we suggest that it might be possible to develop post-processing algorithms that can leverage the rich information in probabilistic model outputs and local knowledge on e.g., likely fracture orientations to (1) identify and remove false-positive fracture detections, and (2) extrapolate detected fractures across small gaps (based on their orientation), as is commonly done by human interpreters.

4.3 Future Work

Together, our results highlight several concrete next steps that could enable automated and objective fracture mapping: improving annotation precision (multi-scale labels, consensus labelling) in training data, expanding dataset diversity by including more sites (as more public data becomes available), and adopting architectures and loss functions tailored to the unusual (thin) geometry of fractures. Complementary strategies such as ensembling, transfer learning, and self-supervised pretraining offer promising avenues to improve performance and robustness. Ensembling multiple models can reduce variance and improve prediction stability, particularly for thin and discontinuous fracture traces. Transfer learning from larger geospatial or remote

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

sensing datasets can provide stronger low-level feature representations, enabling better generalization when labeled fracture data are limited. Self-supervised pretraining on unlabeled data can further leverage abundant data to learn domain-specific texture and structure features, improving downstream segmentation accuracy while reducing dependence on extensive manual annotations.

5 Conclusions

We present a harmonised benchmarking framework for fracture mapping in high-resolution outcrop imagery and use it to compare traditional image filters to deep learning methods, including a U-Net convolutional network and a transformer based model (SegFormer). Our experiments show that the deep learning models consistently outperform traditional edge and ridge filters by producing smoother, more continuous, and qualitatively reasonable results. This gap highlights that algorithmic improvements alone are insufficient: limited post-processing techniques, variable label quality, dataset heterogeneity, and the complex information (e.g., topological understanding) needed to accurately identify high-aspect ratio and typically continuous fractures are the primary barriers to reliable, fully automated fracture extraction. Beyond these empirical results, the principal contribution of this work are the resources we release: the curated datasets, evaluation protocol, and open-source code. We hope that these resources will foster further developments to address the challenges outlined above.

Code and data availability. The FraXet dataset, including all RGB tiles, DEMs, masks, and metadata, is archived and publicly accessible on Zenodo at https://doi.org/10.5281/zenodo.17069947 (Fatihi et al., 2025a).

The full source code used in this study is openly available at https://github.com/ayoubft/fractex2D.pt (Last access: XX XX) and the version associated with this publication is archived at https://doi.org/10.5281/zenodo.17953223 (Fatihi and Thiele, 2025).

All trained models associated with this work are available on both Hugging Face and Zenodo. The model weights can be accessed on Hugging Face at https://huggingface.co/ayoubft/fraXteX (Last access: XX XX) and are archived on Zenodo at https://doi.org/10.5281/zenodo.17866853 (Fatihi et al., 2025b).

A live deployment of the model is hosted on Hugging Face at https://huggingface.co/spaces/ayoubft/fractex2D_tuto, providing an interactive interface for running inference of the trained deep learning models and also try the computer vision filters.

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

Appendix A: Pre-processing effect

Table A1. Effect of preprocessing on fracture-class pixel proportions across datasets

Dataset	Before pre-processing			After preprocessing		
	train	valid	test	train	valid	test
Ovaskainen22	0.45	0.57	0.68	3.08	3.32	4.14
Matteo21	0.29	0.58	0.55	0.39	0.60	1.51
Samsu19	0.43	0.35	0.49	2.17	1.73	2.41

Appendix B: Data Augmentation Setup

Table B1. Data augmentation techniques and corresponding probabilities and parameter ranges

Augmentation Technique	Probability	Parameter Range
Horizontal Flip	0.5	-
Vertical Flip	0.5	-
Gaussian Blur	0.05	Sigma $\in$ [0.1, 2.0]
Brightness Decrease	0.05	Factor $\in$ [0.7, 0.9]
Brightness Increase	0.15	Factor $\in$ [1.1, 1.7]
Contrast Adjustment	0.05	Factor $\in$ [0.7, 1.5]
Saturation Adjustment	0.05	Factor $\in$ [0.7, 1.5]

Appendix C: Optimal Filters' Parameters

Appendix D: Quantitative Results

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

Table C1. Optimal filters' parameters using the grid search and visually (Ovaskainen22/Matteo21/Samsu19/LapiesdiBou)

Algorithm		Parameter	Opt. Value (grid)	Threshold (grid)	Opt. Value (visual)	Threshold (visual)
Edge detectors	Sobel	-	-	0.8	-	0.19/0.1/0.11/0.11
	Canny	sigma	1.9	-	1.5/0./0.53/1.	-
		low_threshold	0.1		0.3/0.65/0.66/0.1	
		high_threshold	0.7		0.5/1./1./1.	
Ridge detector	Gabor	frequency	0.5	0.0	0.7/1./1./1.	0.12/0.35/0.26/0.5
	Sato	sigmas	range(2, 8, 2)	0.9	range(1, 20, 4)/idem /range(1,4)/(1,)	0.6/.04/0.07/0.6
Phase Congruency		nscale	5	0.5	6/5/6/6	0.002/0.0011/0.15
		norient	5		6/5/6/6	
		minWaveLength	3		5/3/5/5	
		mult	2		2.3/2/2.3/2.3	
		sigmaOnf	0.5		0.6/0.5/0.6/0.6	
		k	2		3.6/3/3.6/3.6	
		cutOff	0.3		0.36/0.3/0.36/0.36	
		g	10		41.6 /10/41.6/41.6	
		noiseMethod	-1		-1	

Table D1. Quantitative results: evaluation metrics on the test set

	AE↓	**MSE↓**	**SSIM↑**	**PSNR↑**	**Acc↑**	**Prec↑**	**F1↑**	**Rec↑**	**Spec↑**	**ROC↑**	**IoU↑**	**CK↑**	**FracSim*↓**	**Loss↓**
phasecong	0.15	0.14	0.0	8.68	0.79	0.28	0.27	0.27	0.88	0.58	0.16	0.15	214.53	0.07
sobel	0.46	0.44	0.0	3.59	0.55	0.18	0.28	0.57	0.55	0.56	0.16	0.06	132.27	0.22
gabor	0.94	0.92	0.0	0.35	0.15	0.14	0.25	**1.00**	0.01	0.50	0.14	0.00	48.12	0.46
sato	0.11	0.10	0.0	10.2	**0.85**	0.30	0.27	0.25	**0.93**	0.59	0.16	0.19	153.76	0.05
canny	0.28	0.27	0.0	5.76	0.69	0.22	0.29	0.42	0.74	0.58	0.17	0.12	71.10	0.13
U-Net	0.07	**0.03**	0.49	**14.7**	0.84	**0.50**	**0.48**	0.45	0.92	**0.69**	**0.31**	**0.39**	**3.40**	**0.02**
SegFormer	0.10	0.05	0.40	13.2	0.79	0.38	0.44	0.52	0.84	0.68	0.28	0.32	3.43	**0.02**
All 1	0.95	0.93	0.01	0.33	0.16	0.16	0.27	**1.00**	0.00	0.50	0.16	0.00	Undef	0.47
All 0	**0.05**	**0.03**	**0.82**	14.69	0.84	0.00	0.00	0.00	0.00	0.50	0.00	0.00	Undef	**0.02**

* was calculated on the modified test set

Author contributions. AF: Conceptualisation, Formal analysis, Data curation, Code development, Investigation, Methodology, Visualisation, Writing – original draft; JC: Writing - review & editing; TB: Methodology, Writing - review & editing; STT: Conceptualisation, Code development, Methodology, Writing - review & editing; AS: Conceptualisation, Methodology, Writing - review & editing;

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

Competing interests. The authors declare that they have no conflict of interest.

Acknowledgements. This study was supported by the University of Lausanne (Université de Lausanne).

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

References

Abid, A., Abdalla, A., Abid, A., Khan, D., Alfozan, A., and Zou, J.: Gradio: Hassle-Free Sharing and Testing of ML Models in the Wild, https://doi.org/10.48550/arXiv.1906.02569, 2019.

Adiri, Z., El Harti, A., Jellouli, A., Lhissou, R., Maacha, L., Azmi, M., Zouhair, M., and Bachaoui, E. M.: Comparison of Landsat-8, ASTER and Sentinel 1 Satellite Remote Sensing Data in Automatic Lineaments Extraction: A Case Study of Sidi Flah-Bouskour Inlier, Moroccan Anti Atlas, Advances in Space Research, 60, 2355–2367, https://doi.org/10.1016/j.asr.2017.09.006, 2017.

An, Y., Du, H., Ma, S., Niu, Y., Liu, D., Wang, J., Du, Y., Childs, C., Walsh, J., and Dong, R.: Current State and Future Directions for Deep Learning Based Automatic Seismic Fault Interpretation: A Systematic Review, Earth-Science Reviews, 243, 104 509, https://doi.org/10.1016/j.earscirev.2023.104509, 2023.

Andrews, B. J., Roberts, J. J., Shipton, Z. K., Bigi, S., Tartarello, M. C., and Johnson, G.: How Do We See Fractures? Quantifying Subjective Bias in Fracture Data Collection, Solid Earth, 10, 487–516, https://doi.org/10.5194/se-10-487-2019, 2019.

Aydin, A.: Fractures, Faults, and Hydrocarbon Entrapment, Migration and Flow, Marine and Petroleum Geology, 17, 797–814, https://doi.org/10.1016/S0264-8172(00)00020-9, 2000.

Badoux, H., Bonnard, E. G., and Burri, M.: Geologischer Atlas der Schweiz. 35 = Bl. 546: St.-Léonard: avec annexe de la feuille Sion / levé géologique par: H. Badoux, E. G. Bonnard, M. Burri, 1959.

Bond, C., Gibbs, A., Shipton, Z., and Jones, S.: What Do You Think This Is? "Conceptual Uncertainty" in Geoscience Interpretation, GSA Today, 17, 4, https://doi.org/10.1130/GSAT01711A.1, 2007.

Candès, E. J. and Donoho, D. L.: Continuous Curvelet Transform: I. Resolution of the Wavefront Set, Applied and Computational Harmonic Analysis, 19, 162–197, https://doi.org/10.1016/j.acha.2005.02.003, 2005.

Candès, E. J. and Guo, F.: New Multiscale Transforms, Minimum Total Variation Synthesis: Applications to Edge-Preserving Image Reconstruction, Signal Processing, 82, 1519–1543, https://doi.org/10.1016/S0165-1684(02)00300-6, 2002.

Cappa, F., Guglielmi, Y., Nussbaum, C., and Birkholzer, J.: On the Relationship Between Fault Permeability Increases, Induced Stress Perturbation, and the Growth of Aseismic Slip During Fluid Injection, Geophysical Research Letters, https://doi.org/10.1029/2018GL080233, 2018.

Chudasama, B., Ovaskainen, N., Tamminen, J., Nordbäck, N., Engström, J., and Aaltonen, I.: Automated Mapping of Bedrock-Fracture Traces from UAV-acquired Images Using U-Net Convolutional Neural Networks, Computers & Geosciences, 182, 105 463, https://doi.org/10.1016/j.cageo.2023.105463, 2024.

Constantine, A.: Sedimentology, Stratigraphy and Palaeoenvironment of the Upper Jurassic-Lower Cretaceous Non-Marine Strzelecki Group, Gippsland Basin, Southeastern Australia, Ph.D. thesis, Monash University, 2001.

Do, M. and Vetterli, M.: The Contourlet Transform: An Efficient Directional Multiresolution Image Representation, IEEE Transactions on Image Processing, 14, 2091–2106, https://doi.org/10.1109/TIP.2005.859376, 2005.

Duda, R. O. and Hart, P. E.: Use of the Hough Transformation to Detect Lines and Curves in Pictures, Communications of the ACM, 15, 11–15, https://doi.org/10.1145/361237.361242, 1972.

Fan, J. and Ni, C.: Comparative Studies on the Extraction of Lineaments and Its Variability Based on an Improved Line Segment Tracking Method—Taking the Gaosong Ore Field in the Gejiu Gejiu Tin Ore Deposit as an Example, Applied Sciences, 13, 1314, https://doi.org/10.3390/app13031314, 2023.

Fatihi, A. and Thiele, S.: Ayoubft/fractex2D.Pt, Zenodo, https://doi.org/10.5281/zenodo.17953223, 2025.

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

Fatihi, A., Caldeira, J., Beucler, T., Thiele, S., and Samsu, A.: FraXet, https://doi.org/10.5281/zenodo.17069947, 2025a.

Fatihi, A., Caldeira, J., Beucler, T., Thiele, S., and Samsu, A.: Fracture Mapping Models Trained on FraXet, https://doi.org/10.5281/zenodo.17866853, 2025b.

Gaikwad, V., Singh, K., Salunke, V., and Kudnar, N.: GIS-based Comparative Analysis of Lineament Extraction by Using Different Azimuth Angles: A Case Study of Mula River Basin, Maharashtra, India, Arabian Journal of Geosciences, 16, 538, https://doi.org/10.1007/s12517-023-11636-2, 2023.

Guo, Y.: Augmentation is AUtO-Net: Augmentation-Driven Contrastive Multiview Learning for Medical Image Segmentation, https://doi.org/10.48550/arXiv.2311.01023, arXiv:2311.01023 [cs], 2023.

Han, L., Liu, Z., Ning, Y., and Zhao, Z.: Extraction and Analysis of Geological Lineaments Combining a DEM and Remote Sensing Images from the Northern Baoji Loess Area, Advances in Space Research, 62, 2480–2493, https://doi.org/10.1016/j.asr.2018.07.030, 2018.

Härmä, P.: Natural Stone Exploration in the Classic Wiborg Rapakivi Granite Batholith of Southeastern Finland – New Insights from Integration of Lithological, Geophysical and Structural Data, Ph.D. thesis, Department of Geosciences and Geography, University of Helsinki, 2020.

Koike, K., Nagano, S., and Ohmi, M.: Lineament Analysis of Satellite Images Using a Segment Tracing Algorithm (STA), Computers & Geosciences, 21, 1091–1104, https://doi.org/10.1016/0098-3004(95)00042-7, 1995.

Kovesi, P.: Image Features from Phase Congruency, Videre: Journal of Computer Vision Research, 1999.

Kovesi, P.: Phase Congruency: A Low-Level Image Invariant, Psychological Research, 64, 136–148, https://doi.org/10.1007/s004260000024, 2000.

Krieger, M. L. H.: Geology of the Prescott and Paulden Quadrangles, Arizona, Tech. Rep. 467, U. S. Govt. Print. Off.,, ISSN 2330-7102, https://doi.org/10.3133/pp467, 1965.

Kuş, Z. and Aydin, M.: MedSegBench: A comprehensive benchmark for medical image segmentation in diverse data modalities, Scientific Data, 11, 1283, https://doi.org/10.1038/s41597-024-04159-2, 2024.

LeCun, Y., Bengio, Y., and Hinton, G.: Deep Learning, Nature, 521, 436–444, https://doi.org/10.1038/nature14539, 2015.

Mallat, S. and Hwang, W.: Singularity Detection and Processing with Wavelets, IEEE Transactions on Information Theory, 38, 617–643, https://doi.org/10.1109/18.119727, 1992.

Masoud, A. and Koike, K.: Applicability of Computer-Aided Comprehensive Tool (LINDA: LINeament Detection and Analysis) and Shaded Digital Elevation Model for Characterizing and Interpreting Morphotectonic Features from Lineaments, Computers & Geosciences, 106, 89–100, https://doi.org/10.1016/j.cageo.2017.06.006, 2017.

Masoud, A. A. and Koike, K.: Auto-Detection and Integration of Tectonically Significant Lineaments from SRTM DEM and Remotely-Sensed Geophysical Data, ISPRS Journal of Photogrammetry and Remote Sensing, 66, 818–832, https://doi.org/10.1016/j.isprsjprs.2011.08.003, 2011.

Mattéo, L., Manighetti, I., Tarabalka, Y., Gaucel, J.-M., Van Den Ende, M., Mercier, A., Tasar, O., Girard, N., Leclerc, F., Giampetro, T., Dominguez, S., and Malavieille, J.: Dataset of Manuscript "Automatic Fault Mapping in Remote Optical Images and Topographic Data with Deep Learning", Journal of Geophysical Research - Solid Earth, 2021, 2020.

Mattéo, L., Manighetti, I., Tarabalka, Y., Gaucel, J.-M., van den Ende, M., Mercier, A., Tasar, O., Girard, N., Leclerc, F., Giampetro, T., Dominguez, S., and Malavieille, J.: Automatic Fault Mapping in Remote Optical Images and Topographic Data With Deep Learning, Journal of Geophysical Research: Solid Earth, 126, e2020JB021 269, https://doi.org/10.1029/2020JB021269, 2021.

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

Nordbäck, N. and Ovaskainen, N.: UAV-acquired Orthomosaics of Loviisa Shoreline Outcrops, https://doi.org/10.5281/zenodo.17878870, 2025.

Oliveira, M. J., Savastano, V., Matos, G., Schmitt, R., Valente, V., Araujo, M., and Inocencio, L.: The Use of Drones and Deep Learning to Identify Igneous Rocks and Fractures, in: Offshore Technology Conference Brasil, https://doi.org/10.4043/29829-MS, 2019.

Ovaskainen, N. and Nordbäck, N.: Manually Mapped Traces from UAV-acquired Images of Loviisa Shoreline Outcrops, https://doi.org/10.5281/zenodo.7077846, 2022.

Peacock, D. C. P., Sanderson, D. J., Bastesen, E., Rotevatn, A., and Storstein, T. H.: Causes of Bias and Uncertainty in Fracture Network Analysis, NORWEGIAN JOURNAL OF GEOLOGY, 2019.

Pola, A., Herrera-Díaz, A., Tinoco-Martínez, S. R., Macias, J. L., Soto-Rodríguez, A. N., Soto-Herrera, A. M., Sereno, H., and Ramón Avellán, D.: Rock Characterization, UAV Photogrammetry and Use of Algorithms of Machine Learning as Tools in Mapping Discontinuities and Characterizing Rock Masses in Acoculco Caldera Complex, Bulletin of Engineering Geology and the Environment, 83, 260, https://doi.org/10.1007/s10064-024-03743-5, 2024.

Prabhakaran, R., Bruna, P.-O., Bertotti, G., and Smeulders, D.: An Automated Fracture Trace Detection Technique Using the Complex Shearlet Transform, Solid Earth, 10, 2137–2166, https://doi.org/10.5194/se-10-2137-2019, 2019.

Raghavan, V., Wadatsumi, K., and Masumoto, S.: Automatic Extraction of Lineament Information from Satellite Images Using Digital Elevation Data, Nonrenewable Resources, 2, 148–155, https://doi.org/10.1007/BF02272811, 1993.

Raghavan, V., Masumoto, S., Koike, K., and Nagano, S.: Automatic Lineament Extraction from Digital Images Using a Segment Tracing and Rotation Transformation Approach, Computers & Geosciences, 21, 555–591, https://doi.org/10.1016/0098-3004(94)00097-E, 1995.

Rahnama, M. and Gloaguen, R.: TecLines: A MATLAB-Based Toolbox for Tectonic Lineament Analysis from Satellite Images and DEMs, Part 1: Line Segment Detection and Extraction, Remote Sensing, 6, 5938–5958, https://doi.org/10.3390/rs6075938, 2014.

Ren and Malik: Learning a Classification Model for Segmentation, in: Proceedings Ninth IEEE International Conference on Computer Vision, pp. 10–17 vol.1, IEEE, Nice, France, ISBN 978-0-7695-1950-0, https://doi.org/10.1109/ICCV.2003.1238308, 2003.

Ronneberger, O., Fischer, P., and Brox, T.: U-Net: Convolutional Networks for Biomedical Image Segmentation, in: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, edited by Navab, N., Hornegger, J., Wells, W. M., and Frangi, A. F., Lecture Notes in Computer Science, pp. 234–241, Springer International Publishing, Cham, ISBN 978-3-319-24574-4, https://doi.org/10.1007/978-3-319-24574-4_28, 2015.

Saint Jean Patrick Coulibaly, H., Talnan Jean Honoré, C., Naga, C., Claude Alain Kouadio, K., Régis Mailly DIDI, S., Diedhiou, A., and Savane, I.: Groundwater Exploration Using Extraction of Lineaments from SRTM DEM and Water Flows in Béré Region, The Egyptian Journal of Remote Sensing and Space Science, 24, 391–400, https://doi.org/10.1016/j.ejrs.2020.07.003, 2021.

Samsu, A., Cruden, A., and Vollgger, S.: Scale Matters: The Influence of Structural Inheritance on Fracture Patterns - Supplementary Material, https://doi.org/10.26180/5CDCAD0A73FE0, 2019.

Scheiber, T., Fredin, O., Viola, G., Jarna, A., Gasser, D., and Łapińska-Viola, R.: Manual Extraction of Bedrock Lineaments from High-Resolution LiDAR Data: Methodological Bias and Human Perception, GFF, 137, 362–372, https://doi.org/10.1080/11035897.2015.1085434, 2015.

Sharma, R., Saqib, M., Lin, C. T., and Blumenstein, M.: A Survey on Object Instance Segmentation, SN Computer Science, 3, 499, https://doi.org/10.1007/s42979-022-01407-3, 2022.

https://doi.org/10.5194/egusphere-2026-1097
Preprint. Discussion started: 13 March 2026

Smeraglia, L., Mercuri, M., Tavani, S., Pignalosa, A., Kettermann, M., Billi, A., and Carminati, E.: 3D Discrete Fracture Network (DFN) Models of Damage Zone Fluid Corridors within a Reservoir-Scale Normal Fault in Carbonates: Multiscale Approach Using Field Data and UAV Imagery, Marine and Petroleum Geology, 126, 104 902, https://doi.org/10.1016/j.marpetgeo.2021.104902, 2021.

Soto-Pinto, C., Arellano-Baeza, A., and Sánchez, G.: A New Code for Automatic Detection and Analysis of the Lineament Patterns for Geophysical and Geological Purposes (ADALGEO), Computers & Geosciences, 57, 93–103, https://doi.org/10.1016/j.cageo.2013.03.019, 2013.

Steck, A., Epard, J.-L., Escher, A., Gouffon, Y., and Masson, H.: Carte tectonique des Alpes de Suisse occidentale 1:100 000, Office féd. Eaux Géologie (Berne), ISBN 978-3-906723-44-0, 2001.

Sumi, M. R., Das, P., Hossain, A., Dey, S., and Schuckers, S.: A Comprehensive Evaluation of Iris Segmentation on Benchmarking Datasets, Sensors, 24, 7079, https://doi.org/10.3390/s24217079, 2024.

swisstopo: Tectonic Map of Switzerland 1: 500 000, Federal Office of Topography Swisstopo, Wabern, 2024.

Thiele, S., Vollgger, S., and Anindita Samsu: GeoTrace and Compass Rapid Trace-Mapping (Example Data), https://doi.org/10.4225/03/5981B31091AF9, 2017a.

Thiele, S. T., Grose, L., Samsu, A., Micklethwaite, S., Vollgger, S. A., and Cruden, A. R.: Rapid, Semi-Automatic Fracture and Contact Mapping for Point Clouds, Images and Geophysical Data, Solid Earth, 8, 1241–1253, https://doi.org/10.5194/se-8-1241-2017, 2017b.

Twiss, R. J. and Moores, E. M.: Structural Geology, W. H. Freeman, New York, NY, second edition edn., ISBN 978-0-7167-4951-6, 2006.

Vasuki, Y., Holden, E.-J., Kovesi, P., and Micklethwaite, S.: Semi-Automatic Mapping of Geological Structures Using UAV-based Photogrammetric Data: An Image Analysis Approach, Computers & Geosciences, 69, 22–32, https://doi.org/10.1016/j.cageo.2014.04.012, 2014.

Vasuki, Y., Holden, E.-J., Kovesi, P., and Micklethwaite, S.: An Interactive Image Segmentation Method for Lithological Boundary Detection: A Rapid Mapping Tool for Geologists, Computers & Geosciences, 100, 27–40, https://doi.org/10.1016/j.cageo.2016.12.001, 2017.

Wang, Y., Khodadadzadeh, M., and Zurita-Milla, R.: Spatial+: A New Cross-Validation Method to Evaluate Geospatial Machine Learning Models, International Journal of Applied Earth Observation and Geoinformation, 121, 103 364, https://doi.org/10.1016/j.jag.2023.103364, 2023.

Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J. M., and Luo, P.: SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers, https://doi.org/10.48550/arXiv.2105.15203, 2021.

Yaqoob, M., Ishaq, M., Ansari, M. Y., Konagandla, V. R. S., Tamimi, T. A., Tavani, S., Corradetti, A., and Seers, T. D.: GeoCrack: A High-Resolution Dataset For Segmentation of Fracture Edges in Geological Outcrops, Scientific Data, 11, 1318, https://doi.org/10.1038/s41597-024-04107-0, 2024.

Zhang, R., Yi, X., Li, H., and Lu, G.: Automatic Extraction of Geological Discontinuities of a Tunnel Surface by Integrating Multiple Features, Tunnelling and Underground Space Technology, 154, 106 072, https://doi.org/10.1016/j.tust.2024.106072, 2024.