IoU or an F1 and treats the per-pixel probability map as the deliverable. For an operator it is not the deliverable. A petrophysicist cannot load a pixel mask into an interpretation package, cannot cross-check it against an LAS file, and cannot run petrophysics on it. The thing they need is a curve: one value per depth, monotone in depth, exportable as a column of numbers. The gap between the mask and that curve is small, unglamorous, and decisive, and it is where a digitisation system either earns the operator's trust or loses it. This whitepaper is the reference pipeline for that gap in VeerNet, the raster well-log digitisation architecture we built at EarthScan. We take the network's three-class mask, background plus two curve classes, and walk it through four deterministic stages: extract the per-depth centreline of each curve region so a band of foreground pixels collapses to a single ordinate; convert that pixel ordinate into a calibrated petrophysical value against the track's printed scale; fit a cubic spline through the resulting depth-value pairs so gaps from faint or broken traces are bridged smoothly rather than left as holes; and resample the spline to a fixed grid of 300 depth points that exports cleanly as CSV. We report the reconstruction errors we actually measured against held-out ground-truth curves under the Dice-trained model: a CSV mean absolute error of 0.11 for the first curve and 0.12 for the second, with mean squared errors of 0.03 and 0.04 respectively, on the model's normalised value range. We cover why the centreline, not the mask boundary, is the correct estimator; why cubic-spline interpolation beats linear for a physically smooth log response; why 300 points is a deliberate resolution choice rather than an arbitrary one; and where the residual error comes from so an interpreting team knows exactly what they are reviewing. The segmentation is upstream of this document. What follows is the deterministic post-processing that turns a mask into a number a petrophysicist can use.

Tarry Singh
Founder & CEO

Tannistha Maiti
Senior AI Researcher

Quamer Nasim
ML Research Engineer

Narendra Patwardhan
Research Collaborator

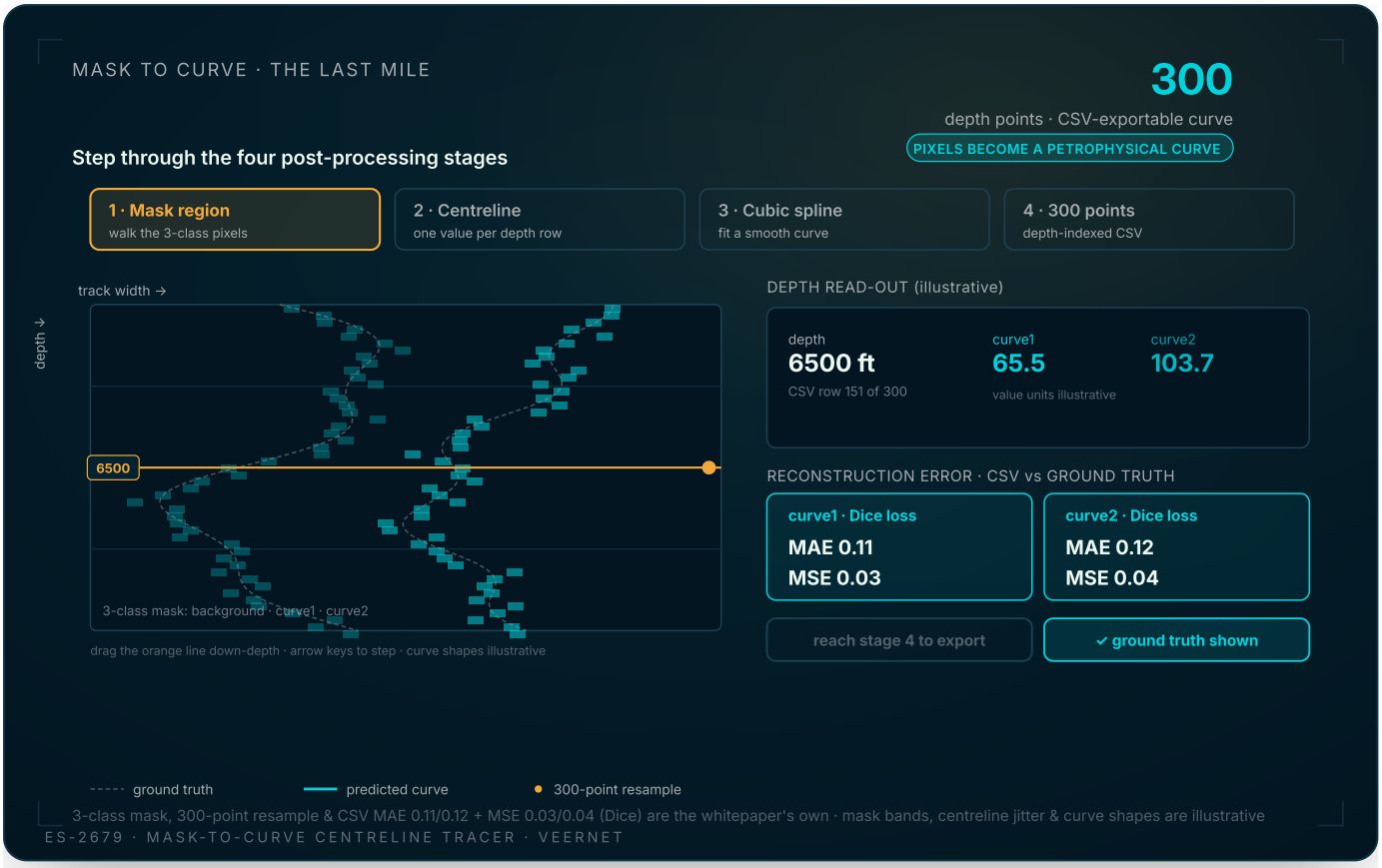
VEERNET / MASK-TO-VECTOR / CURVE-RECONSTRUCTION / CENTRELINE-EXTRACTION / SKELETONIZATION / CUBIC-SPLINE

CONTENTS

A segmentation network does not give a petrophysicist a curve. This is the sentence that most well-log digitisation papers quietly skip, and it is the one that decides whether the work is useful. Run VeerNet, our raster-log digitisation network, on a scanned log and what comes back is a per-pixel probability map: for every pixel in the image, the model's belief that it belongs to the background, to the first curve, or to the second. That map is a genuine achievement, and it is also not a thing an operator can use. You cannot load a pixel mask into NeuraLog. You cannot cross-check a pixel mask against an LAS file pulled from the Texas Railroad Commission archive. You cannot run a porosity calculation on a probability tensor. The deliverable is a curve, one value per depth, and the distance from the mask to that curve is the subject of this paper.

We call it the last mile because it has the last mile's character: short, undramatic, and the place where the whole delivery succeeds or fails. The architecture work and the loss-function ablation are upstream of this document, and the geoscience is upstream of the architecture. None of it reaches a production interpretation seat unless the mask becomes a column of numbers that a petrophysicist trusts. What follows is the reference pipeline for doing exactly that, and the reconstruction errors we measured when we did.

The pipeline is deliberately deterministic. There is no second neural network in this stage, no learned post-processor, nothing to retrain. Once the mask exists, four fixed steps turn it into a curve: extract the per-depth centreline of each curve region, calibrate that pixel ordinate to a petrophysical value, fit a cubic spline through the depth-value pairs, and resample to a fixed grid of 300 depth points for export. Determinism here is a feature, not a limitation. A geoscientist signing off on a digitised curve for a joint-venture audit needs to know that the same mask produces the same numbers every time, and that the only stochastic component in the system was the model that drew the mask.



The last mile of raster-log digitisation, made watchable. A 3-class pixel mask (background, curve1, curve2) becomes a depth-indexed, CSV-exportable petrophysical curve through four deterministic stages: walk each mask region, take the per-depth centreline, fit a cubic spline, and resample to 300 depth points. Drag the orange scrub line down-depth to watch pixels turn into numbers; toggle the schematic ground truth and step through the stages. The 3-class mask, the 300-point resample target, and the reconstruction errors (CSV MAE 0.11/0.12 and MSE 0.03/0.04 for curve1/curve2 under Dice loss) are the whitepaper's own; the mask band geometry, centreline jitter, and curve shapes are illustrative.

Why the mask is not the answer

It helps to be precise about what a three-class mask actually is, because the gap to a curve is easy to underestimate. VeerNet's multiclass head emits three channels: background, curve one, and curve two. After an argmax you have a label image the same size as the input raster, where each pixel carries one of three integer labels. The two foreground classes are not thin lines. They are bands. A printed curve trace on a scanned log is rarely a single pixel wide in practice, because of ink spread, scan blur, anti-aliasing, and the model's own tendency to be generous at the edges of a confident region. So the curve-one class is a ribbon of foreground pixels meandering down the image, several pixels across at any given depth, and the curve-two class is a second ribbon.

A petrophysical curve is the opposite of a ribbon. At every depth it has exactly one value. The whole point of a wireline log is that it is a function: depth in, a single measured quantity out. A band of pixels three or five wide at a given depth row is not a function, it is a set, and you cannot export a set as a curve. So the first real task of the last mile is to collapse each depth row of each curve band down to one representative ordinate. That is the centreline problem, and getting it right is most of what separates a clean reconstruction from a noisy one.

It is worth saying plainly why we do not just take the mask boundary or the mask centroid of the whole region. The boundary of a band traces its left and right edges, which oscillate with ink spread and tell you nothing about where the true curve sits. The centroid of the entire two-dimensional region collapses the depth axis you are trying to preserve. The estimator you want operates per depth row and returns the centre of the foreground run in that row, which is the discrete cousin of the medial axis of the band [2]. That is the centreline, and it is the only one of the obvious choices that respects the function-of-depth structure the curve has to have.

Stage one: per-depth centreline extraction

The centreline is computed row by row down the image. For a given depth row, we look at the run of foreground pixels belonging to a curve class and take its centre. In the clean case, where the band is a single contiguous run, this is just the midpoint of that run, and it is both fast and exact. The result is one horizontal pixel position per depth row, which is precisely the one-value-per-depth structure a curve needs.

The clean case is not the only case, which is why the centreline stage carries more machinery than a midpoint. Three things go wrong on real scanned logs, and each has a deterministic answer.

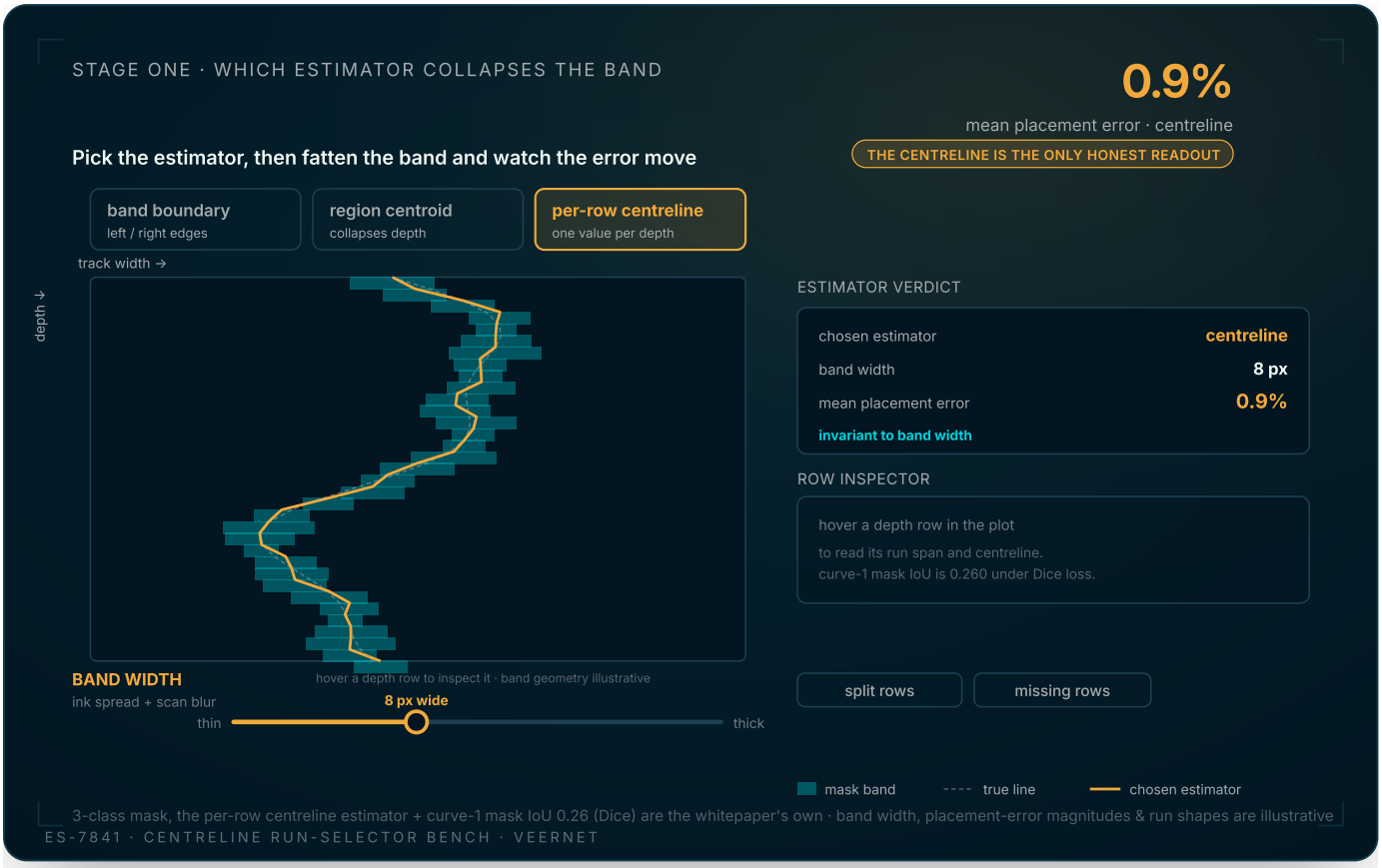
The first is multiple runs in a single row. Where the band is briefly split, by a gridline crossing the trace, by a faded patch, or by a printed annotation overlapping the curve, a depth row can contain two or three separate foreground runs of the same class. Taking the midpoint of the union would place the centreline in the gap between them, which is wrong. We instead select the run most consistent with the centreline immediately above it, which keeps the trace continuous through the interruption rather than letting it jump.

The second is a band thick enough that a naive midpoint is biased. When the foreground region is several pixels wide and slightly asymmetric, the midpoint of the run and the true centre of the underlying printed line can differ by a pixel or two. The principled answer is to thin the band to a unit-width skeleton before reading off the position, which is exactly what the classic two-subiteration thinning algorithm does [1]: it erodes the band symmetrically from both sides until a single-pixel medial line remains, then the centreline is read directly off that skeleton. For a band that is locally a simple ribbon, thinning and run-midpoint agree to within a pixel, and we use the cheaper midpoint; where the band is awkward, the skeleton is the more honest estimator.

The third is missing rows. A faint or broken trace produces depth rows with no foreground pixels of a given class at all. These are not errors to paper over silently, they are genuine gaps in what the scan preserved, and the centreline stage records them as missing rather than guessing. Bridging those gaps is the job of the spline two stages later, and it matters that the gap is carried forward as a known absence rather than filled with a fabricated pixel here.

The output of stage one is therefore not a clean array. It is a set of depth-value pairs, one per row where the curve was found, with explicit gaps where it was not. That sparse, honest representation is the right input to everything downstream.

> **THE CENTRELINE IS THE ESTIMATOR, NOT THE MASK**
>
> A segmentation metric like IoU rewards a model for covering the right pixels. A reconstruction metric rewards it for placing the curve at the right depth-value position. They are not the same thing, and the centreline is where the difference gets resolved. A band that is two pixels too wide on both sides scores worse on IoU but its centreline is unmoved, which is why a model can have a modest IoU and still reconstruct a usable curve. The centreline is the bridge between a pixel-overlap metric and a petrophysical one.



Stage one is an estimator choice, not a script. A foreground mask band is several pixels wide at each depth row, and three obvious readouts exist: the band boundary (its drifting left and right edges), the region centroid (which collapses the depth axis into a single vertical line), and the per-row centreline (the centre of the foreground run). Only the centreline respects the one-value-per-depth function a curve must be. Flip the estimator, drag the band-width lever to fatten the band with ink spread and scan blur, and toggle the split rows (a gridline crossing the trace) and missing rows (a faded stretch) on. Watch the headline placement error: the centreline stays near zero as the band fattens, while the boundary drifts and the centroid is worst of all. The 3-class mask, the per-row centreline estimator, the honest split-run and missing-row handling, and the curve-1 mask IoU of 0.26 under Dice loss are the whitepaper's own; the band geometry, the per-estimator error magnitudes, and the run shapes are illustrative.

Stage two: from pixel ordinate to petrophysical value

The centreline gives a horizontal pixel position per depth. A pixel position is not a petrophysical value. Stage two is the calibration that turns the geometry into a measurement, and it is the stage where the printed log's own conventions do the work.

A wireline track encodes a quantity as horizontal deflection between a left scale value and a right scale value, often on a logarithmic axis for resistivity and a linear one for gamma ray or porosity. The track's scale is printed in the header. So the mapping from a centreline pixel position to a value is a linear, or log-linear, interpolation between the left-edge pixel

mapped to the left scale value and the right-edge pixel mapped to the right scale value. The depth axis is calibrated the same way, against the printed depth scale, so that pixel row maps to depth in feet or metres.

This stage is arithmetic, not inference, and that is the point. Once the track geometry and scale are known, every centreline pixel has a deterministic value. The reason it earns its own section is that calibration error and reconstruction error are different animals, and an interpreting team needs to keep them apart. A perfectly placed centreline on a mis-read scale produces a perfectly smooth, perfectly wrong curve. The errors we report later are reconstruction errors measured against ground-truth curves on the same scale, so they isolate how well the mask-to-centreline-to-spline path recovers the trace, holding calibration fixed. In production the scale detection is its own audited step with its own confidence, upstream of this arithmetic.

Stage three: cubic-spline interpolation

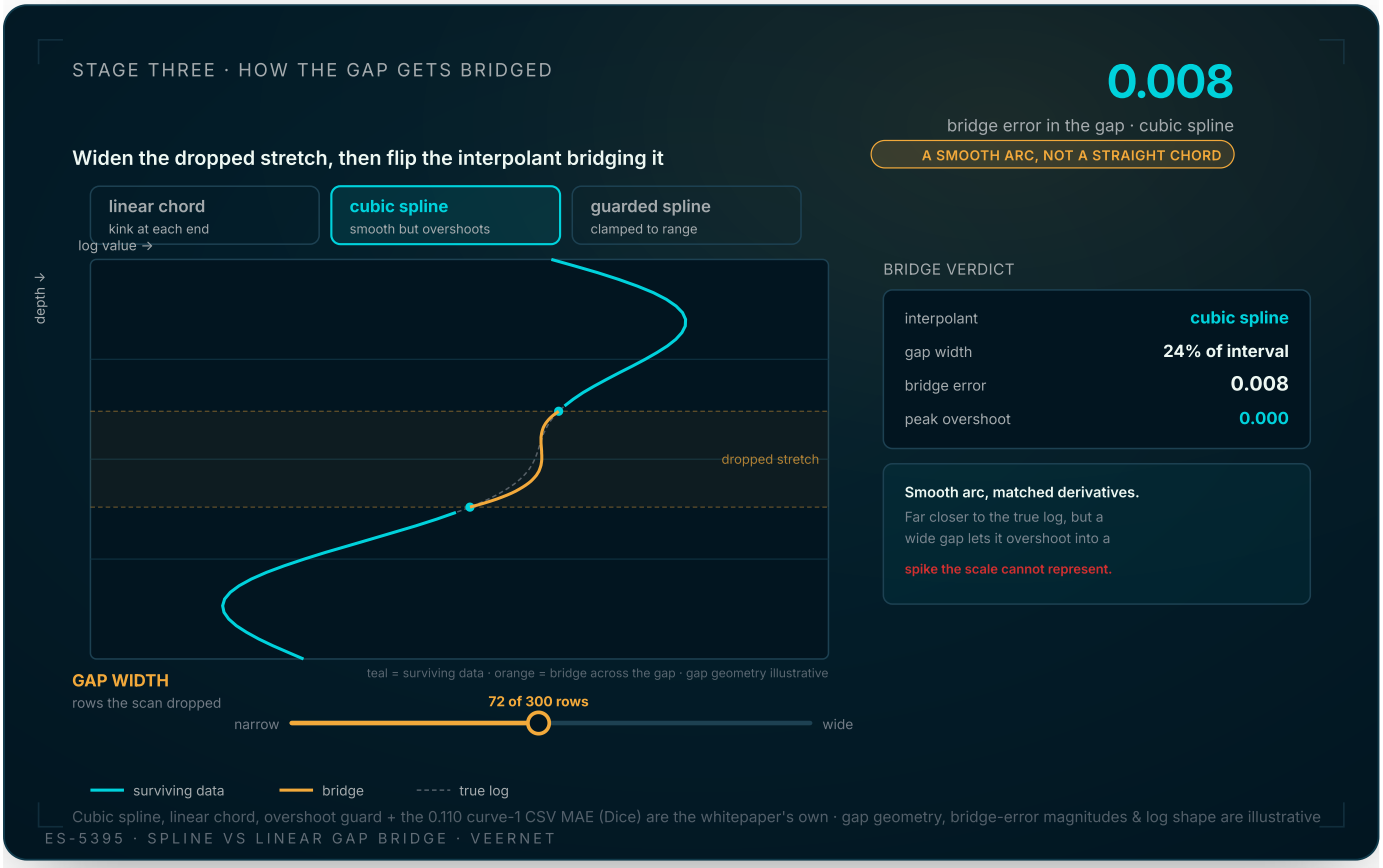
After calibration we have depth-value pairs, irregularly spaced because of the missing rows from stage one, and we need a continuous curve. The choice of interpolant is not cosmetic. It changes the numbers a petrophysicist reads at every depth that fell in a gap.

We fit a cubic spline through the calibrated points [3]. The reason is physical, not aesthetic. A petrophysical log response is smooth: gamma ray, porosity, and resistivity vary continuously with lithology and fluid, and they do not have the sharp corners that a piecewise-linear interpolant would introduce at every knot. Linear interpolation across a gap draws a straight chord between the two surviving points, which understates curvature and plants a visible kink at each end of the gap. A cubic spline matches value, first derivative, and second derivative at the knots, so it bridges a gap with a smooth arc that is consistent with the curve's behaviour on either side of it. For a one-pixel-wide trace that the scan dropped for a few rows, the spline's reconstruction is far closer to the true log than the linear chord would be.

The spline is also where the honest gap-handling from stage one pays off. Because the missing rows were carried forward as genuine absences rather than fabricated pixels, the spline interpolates across them using only real evidence on both sides. Had stage one guessed a pixel in every empty row, the spline would have fit those guesses as if they were

data, and the error would have been baked in below the level where anyone could see it. Carrying the gap forward and letting the interpolant bridge it is the design decision that keeps the residual error interpretable.

One guard rail matters here. A spline through noisy points can overshoot, producing a brief excursion beyond the local data range, which on a log would read as a physically implausible spike. We constrain the fit so the interpolated curve stays within the calibrated value range of its neighbourhood, which trades a hair of smoothness for the guarantee that the exported curve never invents a reading the scale cannot represent.



Stage three is where the interpolant choice changes the number a petrophysicist reads at every depth that fell in a gap. A faint or broken trace leaves the calibrated depth-value pairs sparse, and the dropped stretch has to be bridged. A linear chord draws a straight line between the two surviving knots, understating curvature and planting a kink at each gap edge. A cubic spline matches value and first and second derivative at the knots, so it bridges with a smooth arc consistent with the log on either side, but an unconstrained spline can overshoot into a physically implausible spike. The guarded spline clamps the fit to the local calibrated range. Drag the gap-width lever to widen the dropped stretch and flip the interpolant across the three regimes: the read-out reports the bridge error against the true log and how far the unguarded spline overshoots before the guard pulls it back. The cubic-spline interpolant, the linear-chord alternative, the overshoot guard rail, and the curve-1 CSV MAE of 0.11 under Dice loss that the full path settles toward are the whitepaper's own; the gap geometry, the per-interpolant error magnitudes, and the log shape are illustrative.

Stage four: resampling to 300 depth points

The spline is continuous, so in principle you could sample it at any depth resolution. We resample every reconstructed curve to a fixed grid of **300 depth points** spanning the digitised interval [4]. The fixed grid is what makes the output a clean, comparable, exportable artefact: a CSV with a depth column and one value column per curve, 300 rows, the same shape every time regardless of how tall the source raster was or how many depth rows survived stage one.

Three hundred is a deliberate choice, not a default. It is dense enough to preserve the features a petrophysicist cares about over a typical digitised interval, the inflections and the bed boundaries, without oversampling the spline into a column so long that it implies a precision the original scan never had. A 12,800-pixel-tall raster does not carry 12,800 independent depth measurements; the printed trace was sampled by the logging tool at a far coarser real resolution, and resampling to a sane fixed grid is more honest about the information content than emitting one row per pixel. It also makes every curve in the archive directly comparable and trivially loadable, which is what an operator running petrophysics across thousands of digitised logs actually needs.

The resampled grid is the export. From here the same depth-value columns serialise to CSV for quick inspection and to LAS or the binary wireline log format for the interpretation package. The CSV is the format we measure against, because it is the format in which the curve is most directly comparable to the ground-truth column, value for value, depth for depth.

What the reconstruction actually costs: the errors we measured

A reference pipeline is only worth the name if it comes with measured error. We evaluated the full mask-to-CSV path against held-out ground-truth curves, comparing the 300-point reconstructed CSV column to the true curve value for value, under the Dice-loss-trained multiclass model. These are the numbers.

RECONSTRUCTION ERROR, CSV VS GROUND TRUTH (DICE-TRAINED MODEL)

0.11
Mean absolute error, curve 1

0.12
Mean absolute error, curve 2

0.03
Mean squared error, curve 1

0.04
Mean squared error, curve 2

The mean absolute error of the exported CSV is **0.11 for curve one and 0.12 for curve two**, on the model's normalised value range, with mean squared errors of **0.03 and 0.04** respectively. Two things are worth reading carefully off those figures.

First, MAE and MSE tell complementary stories. The mean absolute error is the average value gap a petrophysicist would see if they overlaid the reconstructed curve on the truth: a typical depth is off by around a tenth of the normalised range. The mean squared error, being dominated by the largest deviations, is the number that catches the worst rows, the places where a faint trace or a gridline crossing produced a centreline that wandered before the spline pulled it back. That the MSE sits at 0.03 and 0.04 says the worst-case excursions are bounded and rare rather than systematic, which is exactly the profile you want from a deterministic post-processor sitting behind a probabilistic mask.

Second, curve two is consistently a touch harder than curve one, by one hundredth on both metrics. That is not noise, it is structure, and it traces straight back to the segmentation. The second curve class is the harder one for the network to separate cleanly where the two

traces run close together or cross, so its mask band is slightly noisier, its centreline slightly more interrupted, and its reconstruction slightly looser. The last mile faithfully transmits the difficulty of the mask it was handed. It does not manufacture error and it does not hide it.

> **THE POST-PROCESSOR INHERITS THE MASK'S QUALITY, HONESTLY**
>
> The reconstruction error is the sum of two things: how well the network drew the mask, and how well the deterministic pipeline recovered the curve from it. Because the pipeline is deterministic, the second term is stable and auditable, which means the variation an operator sees from log to log is almost entirely the first term, the mask. That separation is the practical value of a deterministic last mile: it makes the reconstruction error a clean readout of segmentation quality, not a tangle of two stochastic stages.

Where the residual error comes from

An interpreting team reviewing a digitised curve deserves to know what they are reviewing, so it is worth naming the sources of the residual 0.11 to 0.12 MAE explicitly rather than treating it as an opaque model number.

The largest contributor is mask noise at the band edges, which perturbs the centreline. Where the network is confident and the trace is crisp, the centreline is essentially exact and the local error is far below the average. The average is pulled up by the rows where the band is ragged, split, or faint, because those are the rows where the centre of the foreground run is least certain. This is why the error correlates with scan quality and with how close the two curves run: both make the bands harder to delineate.

The second contributor is spline bridging across the longer gaps. Across a short gap of a few rows the spline reconstruction is excellent. Across a long stretch where the trace faded badly, the spline is interpolating from real evidence at the ends but has no evidence in the middle, so its value there is a smooth guess. It is a good guess, far better than a linear chord, but it is still the part of the curve with the least support, and it contributes disproportionately to the worst-case rows that the MSE picks up.

The third, smaller, contributor is the resampling itself. Mapping a continuous spline onto a 300-point grid introduces a small, bounded discretisation difference against a ground truth sampled on a different grid. It is the least of the three and the most controllable, and we accept it as the price of a clean, comparable, fixed-shape export.

What is explicitly not in the residual is calibration error, because the evaluation holds the scale fixed for both the reconstruction and the ground truth. In production, scale detection adds its own, separately audited, uncertainty on top of these figures, which is why we keep it as its own confidence-scored step rather than folding it into the reconstruction number.

Why this stage deserves its own engineering attention

It would be easy to treat the last mile as a scripting afterthought, a few lines of post-processing nobody needs to think hard about. That instinct is how digitisation systems ship masks that look impressive in a paper and curves that interpreters quietly distrust. The four stages here each carry a real decision: the centreline rather than the boundary or centroid, thinning where the band is awkward, the cubic spline rather than the linear chord, the honest gap rather than the fabricated pixel, the fixed 300-point grid rather than one row per pixel. Get any of them wrong and the error moves below the level where it is visible, which is the worst place for error to live in a curve someone will make a completion decision on.

The deterministic character is the through-line. Everything in this pipeline is reproducible, inspectable, and stable from run to run, which is what lets a geoscientist sign their name to the output and what lets the reconstruction error be read as a clean measure of mask quality. The network is where the intelligence lives. The last mile is where the trust is earned, and it earns it by being boring, exact, and honest about what it could and could not recover.

Get the full whitepaper

This page is the long-form summary. The complete whitepaper includes the per-stage reference implementation, the run-selection rule for split rows, the spline overshoot guard derivation, the full per-curve error tables under each loss function, and the LAS and binary-format export schema.

References

[1] Zhang, T. Y., Suen, C. Y. (1984). A Fast Parallel Algorithm for Thinning Digital Patterns. Communications of the ACM, 27(3), 236-239. The two-subiteration thinning algorithm behind reducing a foreground band to a unit-width skeleton. https://dl.acm.org/doi/10.1145/357994.358023

[2] Lee, T. C., Kashyap, R. L., Chu, C. N. (1994). Building Skeleton Models via 3-D Medial Surface/Axis Thinning Algorithms. CVGIP: Graphical Models and Image Processing, 56(6), 462-478. The medial-axis formulation that frames a curve centreline as the skeleton of its mask region. https://dl.acm.org/doi/10.1006/cgip.1994.1042

[3] de Boor, C. (1978). A Practical Guide to Splines. Applied Mathematical Sciences, Vol. 27. Springer-Verlag. The standard reference for the cubic-spline interpolation we fit through the depth-value pairs. https://link.springer.com/book/9780387953663

[4] Virtanen, P., et al. (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. Nature Methods, 17, 261-272. The interpolation and resampling routines the reference pipeline is built on. https://www.nature.com/articles/s41592-019-0686-2

EARTHSCAN

A Reference Pipeline for Mask-to-Vector Curve Reconstruction

Published March 2023. Generated 2026-07-22.
https://www.earthscan.io/whitepapers/mask-to-vector-curve-reconstruction-pipeline