Proceedings of the
13th International Conference on Applied Informatics
Eger, Hungary, February 19–21, 2026
DOI: 10.17048/icai.2026.66
pp. 66–80

Topology-Aware Neural Vectorization of Orienteering Maps: Contour Segmentation as a Demanding Test Case for Editable Reconstruction*

Péter Borbás, Ede Troll

Eszterházy Károly Catholic University, Eger, Hungary
borbas.peter@uni-eszterhazy.hu
troll.ede@uni-eszterhazy.hu

Abstract. We present a reproducible end-to-end pipeline for converting printed orienteering maps into editable vector layers. While the broader framework targets multiple ISOM symbol classes, this paper focuses on elevation contours as a particularly demanding test case because they are central to terrain interpretation, frequently occluded by overprinted symbols, and sensitive to topological degradation during raster-to-vector conversion.

To assess practical usefulness rather than raster similarity alone, we combine overlap metrics (Dice, IoU, Boundary-F1) with topology- and structure-aware measures, including Betti error, warping error, and graph-level error proxies. The experiments reveal a consistent raster–vector mismatch: models with nearly identical raster validation scores can produce materially different vector graphs after tracing. In particular, U-Net with a ResNet-18 encoder provided the best efficiency–quality trade-off despite near-tied raster scores with heavier variants, while topology-aware losses improved topology-related raster metrics without a comparably strong advantage in the post-vectorization evaluation. These results show that editable map reconstruction must be evaluated end-to-end in the vector domain, since raster overlap alone is not a reliable proxy for downstream structural fidelity or subsequent editing effort.

Keywords: orienteering maps, map vectorization, contour extraction, U-Net, topology-aware evaluation, raster-to-vector conversion

*This research was supported by the EKÖP-25-I-1 scholarship program.

1. Introduction

This paper examines whether elevation contours can be recovered from printed orienteering maps in a form suitable for editable vector reconstruction. In the broader TANVOM framework, contour extraction serves as a demanding test case for multi-layer recovery.

Printed orienteering maps are a difficult vectorization domain. They are standardized by the International Specification for Orienteering Maps (ISOM), yet visually dense and heavily layered [8]. Contours, paths, vegetation, and other symbols often overlap or partially occlude one another, while some interruptions are cartographically valid rather than erroneous. As a result, classical thresholding–thinning–tracing pipelines are often unstable on noisy or scanned inputs, especially when semantic context is needed to disambiguate overlapping structures.

Contours are particularly suitable for evaluation. They are central to terrain interpretation, but also topologically fragile: small raster defects can easily turn into breaks, spurs, or false junctions after skeletonization and graph tracing. This makes contour extraction a useful benchmark for testing whether raster predictions remain useful after deterministic raster-to-vector conversion.

A further challenge is evaluation. In practice, the final artifact is not a segmentation mask but an editable vector graph. Raster overlap alone is therefore insufficient for model selection. Inspired in part by prior contour-vectorization work that emphasized topology-aware and editing-oriented assessment [20], we ask whether raster-level metrics reliably predict downstream vector quality, or whether model rankings change after tracing.

The main contributions are threefold. First, we describe a reproducible end-to-end pipeline from OCAD-derived symbol masks to editable GeoJSON/OMAP outputs. Second, we define an evaluation protocol that combines raster overlap, topology-aware diagnostics, and post-vectorization benchmarking under a fixed deterministic backend. Third, through controlled architecture and loss comparisons, we show that raster-domain metrics alone are weak predictors of downstream editability, so model selection should be performed end-to-end in the vector domain.

2. Related Work

Map vectorization has a long tradition in document analysis and GIS. Classical systems typically decompose the task into colour or intensity separation, raster cleanup, thinning, and geometric tracing. In historical-map workflows, colour thresholding and statistical pixel classification have been used to isolate thematic layers, after which morphology, skeletonization, and line-following convert raster masks into vector primitives [1, 7]. Practical vectorization tools and contour-specific workflows follow similar assumptions: if the input is clean and the target colour is well separated, thinning-based conversion can produce useful editable curves [14, 21]. However, these methods are local and parameter-sensitive. Paper texture, scan noise, colour drift, compression, and overprinted symbols can change the appearance of the same cartographic object, and small raster errors may become broken polylines, false junctions, or unnecessary manual edits.

Contour extraction from topographic paper maps has also been studied as a dedicated problem. Earlier work on USGS paper maps combined colour separation, line tracking, and geographic-feature extraction to recover contour and auxiliary information [9]. Oka et al. studied contour vectorization from scanned topographic maps using a sequence of segmentation, thinning, and geometric reconstruction steps [13]. These approaches demonstrate that traditional pipelines can be effective under controlled map design and scanning conditions, but they have limited semantic understanding of whether a line interruption is intentional, whether nearby fragments should be connected, or whether a crossing is cartographically meaningful. In printed orienteering maps this limitation is amplified by the ISOM symbol system, where contours, paths, vegetation screens, rock features, erosion gullies, and point symbols may overlap in dense areas, and where some breaks are intentional for legibility [8].

Deep learning changed the front end of this pipeline by replacing hand-tuned colour rules with learned segmentation models. Encoder–decoder architectures such as U-Net are widely used because they combine fine spatial localization with contextual features [17]. More broadly, CNN-based segmentation has become a standard tool in image segmentation and remote sensing, where robustness to texture, local clutter, and appearance variation is crucial [11, 12, 23]. In historical-map research, semantic segmentation has been used to transfer across heterogeneous map styles [16], while road-network extraction from historical maps has been addressed with deep-learning models designed for variable symbols, sheet quality, and class imbalance [5, 19, 22]. These studies show that learned raster prediction can reduce the brittleness of threshold-based systems, but they also highlight a key limitation for cartographic production: a high-quality segmentation mask is not automatically an editable vector map.

Recent map-vectorization research therefore increasingly evaluates the full reconstruction pipeline. Chen et al. proposed a deep edge filtering approach in which a neural model predicts a simplified edge map that is then converted into closed shapes [3]; the same research direction later led to a benchmark for automatic historical-map vectorization [4]. This is close in spirit to our downstream-aware setup, because the neural stage and the deterministic geometric stage are considered together. However, much of this work targets region boundaries and closed polygonal objects. Thin linear objects have different failure modes, including fragmentation, duplicated centerlines, dangling endpoints, false junctions, and unstable graph connectivity. These errors may be visually small but costly in a cartographic editor.

The closest recent work to the present contour setting is the study by Vynikal and Pacina, who investigate automatic elevation-contour vectorization with a deep-learning approach and emphasize evaluation after vectorization [20]. Their work moves contour extraction beyond pixel overlap and toward edit-oriented assessment. Our paper follows the same downstream motivation, but differs in domain and task constraints. Orienteering maps are competition-oriented, highly generalized, multi-layer documents governed by ISOM. This creates additional ambiguity between the printed symbol and the intended editable geometry, especially around valid breaks, overprint, and symbol families sharing similar colours.

A parallel line of research aims to improve the topology of thin-structure segmentation directly through the loss function. Topology-preserving losses can penalize changes in connected components and holes [6]; clDice encourages centerline overlap for tubular structures [18]; and Skeleton Recall Loss was introduced to preserve connectivity more efficiently for thin structures [10]. These losses are relevant to contour recovery because the downstream vector graph depends heavily on connectivity. Nevertheless, recent reassessments suggest that topology-based losses do not always outperform well-tuned baselines and may be sensitive to implementation and evaluation choices [15]. This motivates the central question of the present paper: whether topology-oriented improvements visible in the raster domain survive thresholding, skeletonization, graph tracing, and export into editable vector objects.

3. Problem Setting and Data

3.1. Cartographic domain and task focus

Orienteering maps are specialized topographic maps designed for fast terrain navigation. They encode landforms, runnability, vegetation, paths, and other navigationally relevant features in a compact standardized symbol system. In printed ISOM maps, contours coexist with black path symbols, erosion-related line symbols, vegetation overlays, and other content that may overlap spatially or appear visually similar.

The broader TANVOM framework targets multi-layer ISOM recovery, including linework and region layers. The current system already covers several groups, such as contour geometry, black-road geometry, rare line symbols, multi-class region bases, and multi-label region overlays. In this paper, however, we isolate the contour component and formulate it as a binary segmentation task in which ISOM 101 and 102 are merged into a single target.

This design is deliberate. For editable reconstruction, preserving contour continuity is more important than distinguishing ordinary and index contours during raster prediction. The task is therefore defined as geometry-first contour recovery, making it an appropriate demanding test case for downstream vector reconstruction.

3.2. OCAD-derived ground truth and dataset construction

Ground truth is generated from symbol-specific OCAD exports rather than manual pixel annotation. This is a major practical advantage of the domain: because the original cartographic content already exists in a structured authoring environment, the training targets can be derived from authoritative symbol layers. The contour target is built by merging the ISOM 101 and 102 layers into a single binary mask. Training data are constructed from full map sheets by tiled sampling.

Table 1. Summary of the contour dataset used in the experiments.

Tile size	512×512 px
Stride	128 px
Augmented variants per base tile	4
Training samples	18,880
Validation samples	2,664
Total samples	21,544
Ground truth source	OCAD symbol-layer exports
Primary contour target	ISOM 101+102 merged

The augmentation policy is designed to preserve geometric correctness while simulating print and scan variability. Geometric transforms are restricted to discrete D4-type operations such as flips and right-angle rotations, so masks remain pixel-aligned with downstream tasks that depend on exact correspondence. Appearance augmentations simulate realistic degradations such as contrast changes, blur, motion blur, noise, JPEG-like compression, and mild color jitter. This matters because the practical deployment target is not a perfectly clean vector-native export, but printed or scanned map images whose artifacts can later be amplified by thinning and tracing.

Figure 1. Examples of augmented training tiles used for contour segmentation. The augmentations increase appearance variability while preserving the geometric consistency of the contour labels.

The processed tile-level dataset used in this study is publicly available on Zenodo [2].

3.3. Broader pipeline context

Although contours are the only quantitatively analyzed symbol family in this paper, the broader pipeline context is worth stating explicitly. In the full system, roads are treated in a geometry-first manner by merging ISOM 503–508 into a single roads-black mask, followed by type attribution on the traced polylines. Rare linear symbols are represented as per-code binary masks. Region layers are handled by a two-head formulation: a base landcover head predicts exactly one class per pixel, while overlay heads predict per-code binary masks that may overlap the base layer. A separate road-typing dataset is created by sampling RGB patches along skeletonized road pixels and storing them in a compact chunked format. This broader context is not analyzed in detail here, but it motivates why contour editability is treated as a demanding test case rather than as an isolated end in itself.

4. Method

4.1. End-to-end reconstruction pipeline

The proposed workflow consists of five reproducible stages. First, symbol-specific layers are exported from OCAD to obtain authoritative raster masks. Second, these masks and the corresponding RGB map sheets are converted into a tiled dataset with deterministic splitting and augmentation. Third, a neural segmentation model predicts contour probabilities on tiles or stitched full maps. Fourth, the raster predictions are converted into vectors by a fixed deterministic backend. Finally, the reconstructed linework is exported to editable formats such as GeoJSON and OpenOrienteering Mapper project files.

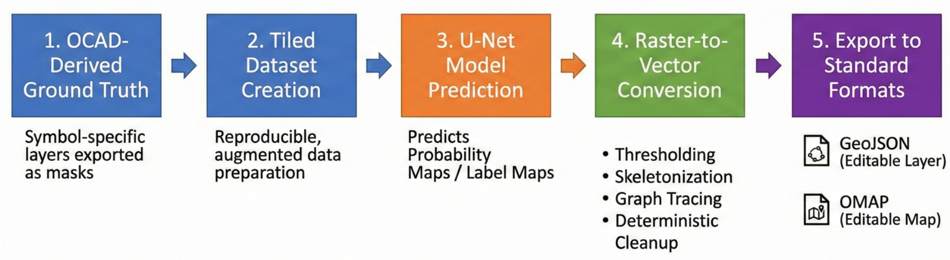


Figure 2. High-level overview of the TANVOM pipeline. Raster-to-vector conversion is treated as part of the evaluated method rather than as an informal post-processing step.

Because the practical output is an editable vector layer, evaluation is performed on the full reconstruction pipeline rather than on raster prediction alone.

4.2. Contour prediction models

For contour prediction, we use U-Net-family models with ResNet encoders. The main architecture comparison includes three variants: U-Net with a ResNet-18 encoder, U-Net with a ResNet-34 encoder, and U-Net++ with a ResNet-34 encoder. The goal is to determine which backbone yields the most useful contour vectors under a shared tracing policy, rather than merely which one maximizes a raster overlap metric.

In addition to standard BCE-style training, we also include a positive-weighted BCE setting that tends to produce thicker strokes. This serves as a controlled calibration test: if a metric is reliable for model selection, its ranking should remain meaningful after vectorization as well.

4.3. Deterministic raster-to-vector conversion

The contour model outputs a probability map, which is thresholded, skeletonized, converted into a graph, and traced into polylines. Simplification, artifact removal, endpoint handling, and graph cleanup are all governed by a fixed policy, after which the results are exported in standard editable formats.

Using a fixed backend is essential for fair comparison. It ensures that differences in final editability can be attributed to the learned raster predictions rather than to per-model tuning of the tracing stage.

4.4. Evaluation protocol

The evaluation is explicitly downstream-oriented. At the raster level, we compute Dice, IoU, and tolerance-aware Boundary-F1. While Dice and IoU summarize overlap, Boundary-F1 is more suitable for thin-line structures because it is less sensitive to small spatial shifts that may be acceptable in practice.

To assess structural correctness, we also compute warping error and Betti error on skeletonized representations. Warping error measures geometric mismatch between predicted and reference centerlines, whereas Betti error captures topological differences in connected components and holes. These metrics are evaluated both on the raster prediction and after vectorization by rasterizing the exported vectors back to the grid.

We further record graph-level proxies such as polyline counts, dangling endpoints, intersections, and generalized error sites obtained by clustering local anomalies. Since some contour interruptions are cartographically valid, topology is interpreted in a context-aware way and allowed breaks are excluded from editing-relevant error summaries.

The final stage is post-vectorization evaluation. After tracing the predicted linework into vectors, we rasterize the exported polylines back to the evaluation grid and recompute the same overlap and topology metrics. This provides a controlled measure of how much quality is preserved, or lost, during thresholding, thinning, tracing, and cleanup.

5. Experiments

5.1. Architecture screening and metric stress test

The first experimental block serves two purposes. The primary goal is to select the most suitable contour backbone among three U-Net-family variants: U-Net with a ResNet-18 encoder, U-Net with a ResNet-34 encoder, and U-Net++ with a ResNet-34 encoder. The secondary goal is to test the stability of the evaluation criteria under a controlled calibration shift induced by positive-class-weighted BCE.

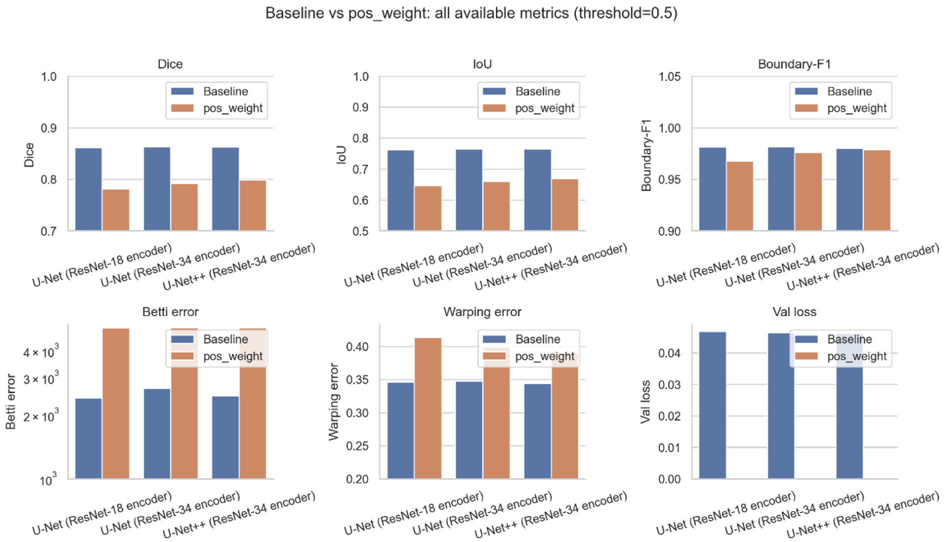
In the calibration-test setting, BCE is reweighted in favor of the sparse contour foreground using a positive-class weight. As expected for a thin-line task, this shifts the probability field toward thicker foreground responses. In raster space, such thickening typically lowers overlap quality and increases structural mismatch, since even small width changes strongly affect sparse masks. Under deterministic tracing, however, the effect is not monotonic: some post-vectorization metrics improve despite worse raster-domain scores, while the exported polyline count may increase because thicker responses can generate short spurs or locally duplicated centerlines after skeletonization. This calibration test is useful precisely because it reveals that metric rankings need not remain stable across the raster-to-vector transition.

For the architecture comparison itself, all three baselines were trained and evaluated under the same data split, augmentation policy, thresholding rule, and tracing backend. Their selected-epoch validation overlap metrics were nearly tied, so the final choice was driven mainly by downstream graph quality and training efficiency. On the proof-tile evaluation, the RN18 backbone produced the lowest generalized error count per tile while also requiring the least training time per epoch. Table 2 summarizes the screening result.

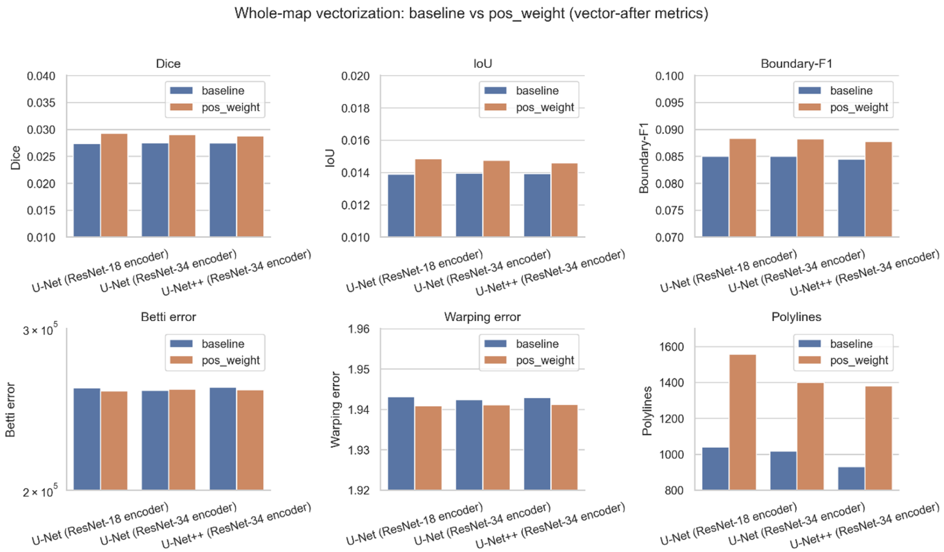
Table 2. Architecture screening for contour extraction. Dice, IoU, and BF1 are selected-epoch validation metrics; generalized errors per tile are measured after vectorization on the proof-tile subset. The RN18 model was selected because its raster scores are competitive while its downstream graph-error count and training cost are both lower.

Model	Dice	IoU	BF1	Gen. err./tile	Min./epoch
U-Net RN18	0.8616	0.7621	0.9814	22.50	12.50
U-Net RN34	0.8632	0.7649	0.9816	32.75	14.04
U-Net++ RN34	0.8629	0.7651	0.9787	24.50	29.75

The conclusion of this stage is not that architecture is irrelevant, but that raster overlap alone is insufficient for choosing among otherwise similar backbones. Under the shared tracing policy, RN18 provides the best efficiency–quality trade-off and is therefore used as the default contour backbone in the subsequent loss study.



(a) Raster-domain metric comparison between the baseline BCE model and the positive-class-weighted BCE variant, evaluated on the raw prediction outputs before vectorization.



(b) Whole-map metric comparison between the baseline BCE model and the positive-class-weighted BCE variant after deterministic vectorization. Unlike the raster-domain comparison, these results reflect the quality of the traced and re-rasterized contour vectors.

Figure 3. Effect of positive-class weighting before and after deterministic vectorization. The top panel shows raster-domain evaluation, while the bottom panel shows whole-map post-vectorization evaluation after tracing and re-rasterization.

5.2. Loss ablation under a fixed vectorization backend

The second experimental block evaluates whether topology-aware objectives improve the final editable contour vectors more reliably than overlap-oriented baselines. All runs use the RN18 backbone, the same train/validation split, the same augmentation protocol, the same inference threshold, and the same deterministic vectorization backend. The compared losses are BCE+Dice with Dice weights $\{0.0, 0.10, 0.25, 0.50\}$, BCE+soft-clDice, and BCE with an additional skeleton-recall term.

This ablation is designed to answer an end-to-end question rather than a purely raster one. If a topology-aware loss is genuinely useful for cartographic editing, then its benefit should remain visible after thresholding, skeletonization, graph tracing, cleanup, and export. A gain that appears only on raster masks but disappears after deterministic vectorization has limited practical value in the present workflow.

To keep the comparison interpretable, raster-domain and post-vectorization results are reported separately. Table 3 lists whole-map raster metrics, while Table 4 gives the corresponding post-vectorization results obtained by tracing the predictions into contour polylines and rasterizing the exported vectors back to the evaluation grid.

Table 3. Whole-map raster-domain results for the RN18 contour loss ablation. Topology-aware losses strongly reduce raster-domain Betti error, while BF1 remains within a narrow range across the competitive variants.

Loss	Dice (raster)	IoU (raster)	BF1 (raster)	Betti (raster)	Warp (raster)
BCE	1.640×10^{-2}	8.265×10^{-3}	0.977	1.802×10^{5}	1.920
BCE+Dice 0.10	1.677×10^{-2}	8.458×10^{-3}	0.978	1.380×10^{5}	1.923
BCE+Dice 0.25	1.733×10^{-2}	8.740×10^{-3}	0.976	1.178×10^{5}	1.922
BCE+Dice 0.50	1.710×10^{-2}	8.623×10^{-3}	0.978	1.011×10^{5}	1.924
BCE+soft-clDice	2.658×10^{-2}	1.347×10^{-2}	0.978	5.102×10^{4}	1.915
BCE+Skeleton-Recall	2.317×10^{-2}	1.172×10^{-2}	0.977	3.169×10^{4}	1.920

Table 4. Post-vectorization results after deterministic tracing and re-rasterization of exported contour polylines. In contrast to the raster-domain Betti reduction, vector-domain Betti and warping errors remain tightly clustered across the competitive variants.

Loss	Polylines	Dice (vector)	IoU (vector)	BF1 (vector)	Betti (vector)	Warp (vector)
BCE	929	6.395×10^{-3}	3.208×10^{-3}	0.916	2.411×10^{5}	1.902
BCE+Dice 0.10	915	3.291×10^{-3}	1.648×10^{-3}	0.953	2.417×10^{5}	1.916
BCE+Dice 0.25	909	2.706×10^{-3}	1.355×10^{-3}	0.958	2.416×10^{5}	1.915
BCE+Dice 0.50	900	2.689×10^{-3}	1.346×10^{-3}	0.959	2.416×10^{5}	1.913
BCE+soft-clDice	918	3.070×10^{-3}	1.537×10^{-3}	0.960	2.414×10^{5}	1.914
BCE+Skeleton-Recall	939	2.584×10^{-3}	1.293×10^{-3}	0.957	2.416×10^{5}	1.915

The raster-domain trend is clear. Relative to plain BCE, the topology-aware losses reduce raster Betti error substantially: soft-clDice lowers it from 1.802×10^{5}

to 5.102×10^4, while Skeleton-Recall reduces it further to 3.169×10^4. At the same time, raster BF1 remains tightly grouped in the 0.976–0.978 range, so the main separation appears in the topology-oriented metric rather than in boundary alignment.

The very low whole-map Dice and IoU values should be interpreted in the context of extremely sparse thin-line targets and strict re-rasterized comparison; in this setting, small centerline shifts or width differences can strongly depress overlap scores even when the reconstructed contour geometry remains practically usable.

After vectorization, this separation is much smaller. Post-vectorization Betti error remains close to 2.41×10^5 across the competitive losses, warping error changes only marginally, and post-vectorization BF1 and IoU form a narrow band. Polyline counts are also similar, ranging from 900 to 939, indicating that the fixed tracing backend largely determines the final contour-graph structure.

Figure 4 supports the same conclusion. The lowest generalized error count is obtained by BCE+Dice with Dice weight 0.50 (1259). Thus, topology-aware losses improve raster topology, but do not provide a clear advantage in the final exported contour graph under the current vectorization policy.

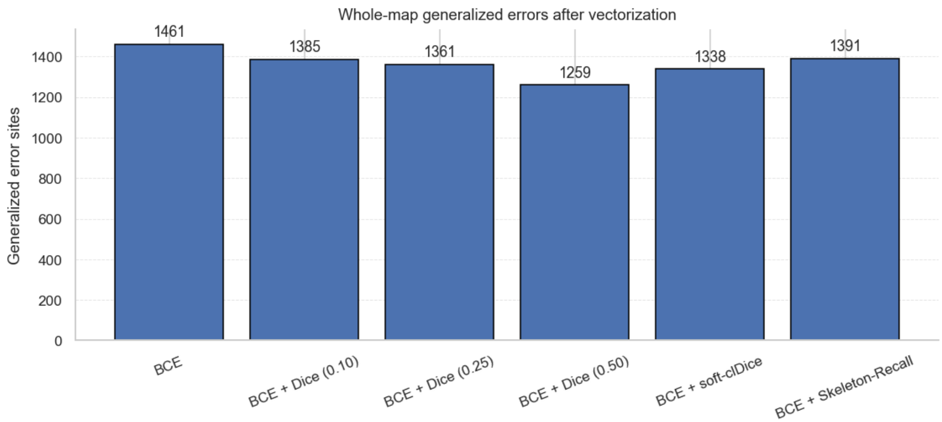


Figure 4. Whole-map generalized error sites after deterministic vectorization for the evaluated RN18 loss variants. Lower values indicate fewer clustered topological repair locations in the exported contour graph.

From a practical perspective, BCE+Dice with Dice weight 0.50 is the strongest default among the tested losses. It combines the best raster BF1, competitive post-vectorization BF1, the lowest polyline count among the strongest variants, and the lowest generalized error count after vectorization. Under the current fixed backend, the main limitation is therefore not the absence of topology-aware supervision, but the limited extent to which raster-domain improvements survive deterministic raster-to-vector conversion.

6. Results and Discussion

6.1. Raster–vector mismatch

The main finding of the paper is the mismatch between raster-domain quality and downstream vector quality. In both the architecture screening and the RN18 loss ablation, rankings changed after deterministic tracing: settings that improved raster-domain metrics did not necessarily yield better exported contour graphs. This shows that raster overlap and raster topology metrics are not sufficient as stand-alone criteria for model selection in editable contour reconstruction.

This is practically important because the target artifact is an editable vector layer rather than a raster mask. Subsequent editing effort is driven by fragmentation, duplicated centerlines, spurs, unstable junctions, and unexplained endpoints, all of which may emerge or worsen during thresholding, skeletonization, and tracing. For this reason, post-vectorization metrics and graph-level repair proxies are more informative than raster scores alone.

Whole-map correlations across the six RN18 loss variants support the same conclusion. Although the sample is small, raster-domain and post-vectorization rankings show weak agreement under the fixed tracing backend.

Table 5. Whole-map raster-to-vector correlation across the six RN18 loss variants. Low absolute correlations indicate only weak agreement between raster-domain and post-vectorization rankings.

Metric	Pearson r
Dice	-0.356
IoU	-0.356
Boundary-F1	-0.048
Betti error	-0.411
Warping error	0.127

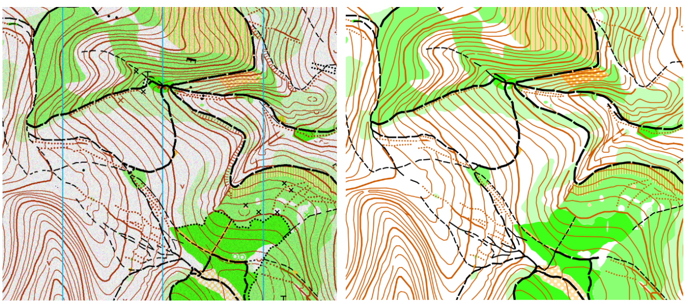


Figure 5. A qualitative end-to-end reconstruction example showing how contour extraction fits into the broader editable-layer pipeline.

The same pattern appears in the post-vectorization repair proxies. Under the current backend, BCE+Dice with Dice weight 0.50 produces the lowest generalized error count and the lowest polyline count among the strongest variants, while topology-aware losses do not provide a clear downstream advantage despite their stronger raster-domain Betti reductions.

6.2. Implications for broader recovery

Although the experiments are contour-centered, the implications extend to the broader TANVOM framework. As additional symbol families are added, evaluation must remain end-to-end, whole-map based, and sensitive to downstream vector quality. The surrounding experiments also suggest that different symbol groups require different calibration and that aggregate raster scores may hide class-specific failure modes. This makes head-specific postprocessing and class-aware checkpoint selection more realistic than a single global criterion.

6.3. Limitations and future work

The study intentionally focuses its quantitative analysis on contours. This makes the experiments controlled and highlights a difficult thin-line case, but it also limits how far the numerical conclusions can be generalized to other ISOM symbol families. Roads, rare linear symbols, landcover regions, and overlay patterns are already represented in the broader TANVOM pipeline, yet their full quantitative evaluation is left for future work. A second limitation is that editing effort is estimated through structural proxies such as unexplained endpoints, intersections, polyline counts, and generalized error clusters. These quantities are closer to the practical goal than pixel overlap, but they are still only approximations. A user-based editor study, measuring actual correction time and mapper judgement, would provide a stronger validation of the proposed error proxies.

Future work should therefore extend the benchmark to additional symbol classes, perform postprocessing and calibration sweeps under the same vector-after protocol, and compare structural proxy scores with real manual editing sessions. This would make it possible to separate three sources of error more clearly: neural recognition, probability calibration, and deterministic graph reconstruction.

7. Conclusion

This paper presented a contour-centered study of topology-aware neural vectorization for printed orienteering maps. Contours were used as a demanding test case because they are both cartographically central and highly sensitive to topological degradation during raster-to-vector conversion. The proposed workflow combines OCAD-derived ground truth, tiled training data, neural contour prediction, deterministic vectorization, and editable export in a single reproducible pipeline.

The experiments lead to a consistent conclusion: raster-domain metrics remain useful, but they are not sufficient for selecting models when the final output is an editable vector layer. Under the present fixed backend, U-Net with a ResNet-18 encoder provided the best efficiency–quality trade-off, while BCE+Dice with Dice weight 0.50 was the strongest practical default among the tested losses.

More broadly, the study shows that map-reconstruction benchmarks should be aligned with the final artifact. If the goal is editable vector recovery, then evaluation must be performed end-to-end in the vector domain.

References

[1] A. G. Auffret, A. Kimberley, J. Plue, et al.: *HistMapR: Rapid Digitization of Historical Land-Use Maps in R*, Methods in Ecology and Evolution 8.11 (2017), pp. 1453–1457.

[2] P. Borbás, E. Troll: *TANVOM v1: Dataset for Orienteering Map Vectorization*, 2026, doi: 10.5281/zenodo.19437536, url: https://doi.org/10.5281/zenodo.19437536.

[3] Y. Chen, E. Carlinet, J. Chazalon, C. Mallet, B. Duménieu, J. Perret: *Vectorization of Historical Maps Using Deep Edge Filtering and Closed Shape Extraction*, in: Lecture Notes in Computer Science, Springer, 2021, pp. 510–525.

[4] Y. Chen, J. Chazalon, E. Carlinet, M. Ô. V. Ngoc, C. Mallet, J. Perret: *Automatic Vectorization of Historical Maps: A Benchmark*, PLOS ONE 19 (2024), e0298217.

[5] B. Ekim, E. Sertel, M. E. Kabadayi: *Automatic Road Extraction from Historical Maps Using Deep Learning Techniques: A Regional Case Study of Turkey in a German World War II Map*, ISPRS International Journal of Geo-Information 10.8 (2021), p. 492.

[6] X. Hu, F. Li, D. Samaras, C. Chen: *Topology-Preserving Deep Image Segmentation*, in: Advances in Neural Information Processing Systems, vol. 32, 2019.

[7] I. Iosifescu, A. Tsorlini, L. Hurni: *Towards a Comprehensive Methodology for Automatic Vectorization of Raster Historical Maps*, e-Perimetron 11.2 (2016), pp. 57–76.

[8] *ISOM 2017-2: International Specification for Orienteering Maps*, Revision 6, January 2024, International Orienteering Federation, 2024.

[9] A. Khotanzad, E. Zink: *Contour Line and Geographic Feature Extraction from USGS Color Topographical Paper Maps*, IEEE Transactions on Pattern Analysis and Machine Intelligence 25.1 (2003), pp. 18–31.

[10] Y. Kirchhoff, M. R. Rokuss, S. Roy, et al.: *Skeleton Recall Loss for Connectivity Conserving and Resource Efficient Segmentation of Thin Tubular Structures*, in: European Conference on Computer Vision (ECCV 2024), Springer, 2024, pp. 218–234.

[11] Y. LeCun, Y. Bengio, G. Hinton: *Deep Learning*, Nature 521.7553 (2015), pp. 436–444.

[12] S. Minaee, Y. Y. Boykov, F. Porikli, A. J. Plaza, N. Kehtarnavaz, D. Terzopoulos: *Image Segmentation Using Deep Learning: A Survey*, IEEE Transactions on Pattern Analysis and Machine Intelligence 44.7 (2021), pp. 3523–3542.

[13] S. Oka, A. Garg, K. Varghese: *Vectorization of Contour Lines from Scanned Topographic Maps*, Automation in Construction 22 (2012), pp. 192–202.

[14] OpenOrienteering: *CoVe, the Contour Vectorizer (Mapper Manual)*, Accessed: 2026-04-17, 2020, url: https://www.openorienteering.org/mapper-manual/pages/cove.html.

[15] OpenReview: *Reassessing the Utility of Topology Based Losses for Image Segmentation*, ICLR 2025 Workshop submission, Accessed: 2026-04-17, 2025, url: https://openreview.net/forum?id=bpBDx2OprB.

[16] R. Petitpierre, F. Kaplan, I. di Lenardo: *Generic Semantic Segmentation of Historical Maps*, in: Proceedings of the Computational Humanities Research Conference, 2021, pp. 228–248.

[17] O. Ronneberger, P. Fischer, T. Brox: *U-Net: Convolutional Networks for Biomedical Image Segmentation*, in: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, vol. 9351, Springer, 2015, pp. 234–241.

[18] S. Shit, J. C. Paetzold, A. Sekuboyina, I. Ezhov, A. Unger, K. Pöppel, G. Tetteh, B. H. Menze: *clDice – A Topology-Preserving Loss Function for Tubular Structure Segmentation*, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 16560–16569.

[19] J. H. Uhl, S. Leyk, Y.-Y. Chiang, C. A. Knoblock: *Towards the Automated Large-Scale Reconstruction of Past Road Networks from Historical Maps*, Computers, Environment and Urban Systems 94 (2022), p. 101794.

[20] J. Vynikal, J. Pacina: *Automatic Elevation Contour Vectorization: A Case Study in a Deep Learning Approach*, ISPRS International Journal of Geo-Information 14.5 (2025), p. 201, doi: 10.3390/ijgi14050201.

[21] WinTopo: *Vectorisation – Thinning (Help Documentation)*, Accessed: 2026-04-17, n.d. url: https://www.wintopo.com/help/html/vectorise.htm.

[22] Y. Zhao, G. Wang, J. Yang, T. Li, Z. Li: *AU3-GAN: A Method for Extracting Roads from Historical Maps Based on an Attention Generative Adversarial Network*, Journal of Geovisualization and Spatial Analysis 8 (2024), p. 26.

[23] X. X. Zhu, D. Tuia, L. Mou, G.-S. Xia, L. Zhang, F. Xu, F. Fraundorfer: *Deep Learning in Remote Sensing: A Comprehensive Review*, IEEE Geoscience and Remote Sensing Magazine 5.4 (2017), pp. 8–36.