ACPV-Net: All-Class Polygonal Vectorization for Seamless Vector Map Generation from Aerial Imagery

Supplementary Material

Overview

In this Appendix, we provide the following information: Sec. 1 provides more details of the Deventer-512 dataset; Sec. 2 specifies the evaluation metrics, including per-class polygon quality metrics, global topology-consistency metrics, and vertex–boundary alignment and peak-shape metrics; Sec. 3 provides additional details of the methodology, including the detailed architecture of SSC, the construction and simplification of the PSLG, and the proof of the sufficient condition for the topological reconstruction; Sec. 4 details the implementation and training, including data preprocessing, ACPV-Net and baseline training for fair comparison; Sec. 5 includes more ablation studies and extended experimental results, as well as additional evaluations on the Shanghai dataset and TopDiG metrics on Deventer-512; Lastly, Sec. 6 closes this supplementary with additional qualitative visualizations and discussion with limitations.

1. Dataset Details for Deventer-512

1.1. Vector Annotation and Topology Validation

The raw polygon annotations of Deventer-512 originate from the official Dutch topographic map, the *Basisregistratie Grootschalige Topografie* (BGT) [9], which provides polygon-based representations of multiple land-cover and land-use categories, and supports mapping at scales between 1:500 and 1:5000. Its standard positional accuracy is reported as 20 cm for objects with a high absolute positional accuracy, and 40 cm for objects with a low absolute positional accuracy. The BGT is created and maintained by various governmental bodies, with revision intervals typically ranging from 6 to 18 months [7].

Because the outlines of different land-cover classes are delineated independently, small geometric inconsistencies may occur along shared boundaries, occasionally producing microscopic gaps or overlaps. To improve topological validity at the patch level, we apply a lightweight topology-validation procedure. Microscopic gaps and overlaps are detected using standard vector overlay operations (*e.g.*, union, intersection), after which adjacent vertices within a tiny geometric tolerance are merged into a single endpoint. This improves shared-edge consistency while preserving the semantic regions at meaningful geometric scales. The resulting cleaned polygon network forms a coherent planar straight-line graph representing a topology-consistent planar partition with shared boundaries. These polygons serve as the ground-truth annotations in all experiments.

1.2. Geometric Statistics and Complexity

Per-class Geometric Statistics. Table S1 reports per-class polygon statistics on the training set, including instance counts, vertex-count distribution (median, P_{90}, max), hole counts, and vertex density. The maximum vertex count ranges from 176 (buildings) to 708 (roads), indicating a wide span of geometric complexity with a strongly long-tailed distribution, particularly for roads and unvegetated regions. Hole counts also differ by orders of magnitude, reflecting highly non-uniform structural complexity across classes.

We additionally compute vertex density–the average number of vertices per unit boundary length, where the boundary length includes both the outer ring and all interior holes. Higher density corresponds to more jagged or fragmented boundaries, whereas lower density indicates smoother shapes. This measure clearly separates smooth water bodies from highly irregular unvegetated areas and road networks.

Table S1. Per-class polygon complexity statistics on the training set. “#Inst.”: number of polygon instances; “Med. Vert.”/“P_{90} Vert.”/“Max Vert.”: vertex count statistics; “Vert. Density”: mean vertex count per unit perimeter.

Class	#Inst.	Med. Vert.	P_{90} Vert.	Max Vert.	#Holes	Vert. Density
Buildings	24,329	5	17	176	133	0.0779
Roads	3,557	14	222	708	8,142	0.0506
Unvegetated	13,254	8	54	454	8,690	0.0828
Vegetation	19,324	7	25	314	1,100	0.0676
Water	2,880	9	33	318	18	0.0461

Geometric Diversity and Structural Complexity. These statistics highlight several dataset-level properties that substantially increase the difficulty of polygonal vectorization. First, the extreme multi-scale variation (from very small buildings to large agricultural parcels and water bodies) requires models to operate across wide spatial ranges. Second, thin and elongated structures such as roads are highly sensitive to small localization errors. Third, many polygons contain holes or nested components, introducing additional boundary loops and complex interior topology. Finally, intra-class shape variability is large, especially in unvegetated and vegetation regions. Together, these characteristics reflect the heterogeneous nature of real-world cadastral geometry and make Deventer-512 a challenging and diverse benchmark for all-class polygonal vectorization.

Representative visual examples are provided in Sec. 6.

1.3. Challenging Visual Conditions and Subsets

Aerial imagery often contains regions where polygon boundaries cannot be reliably inferred from visual evidence. We identify two major types of challenging cases and annotate corresponding subsets on the *test set*: (i) *shadow/occlusion*, and (ii) *semantically ambiguous boundaries*. Table S2 summarizes the number of test patches belonging to each subset; a patch may contain multiple tags.

Shadow / Occlusion. Targets are partially covered by trees or built structures, obscuring true boundaries and creating visually missing segments.

Semantically Ambiguous Boundaries. A distinctive challenge of Deventer-512 is that polygons follow *cadastral or administrative units* rather than purely visual contours. This produces three typical forms of semantic–visual discrepancy: (1) parcel- or property-based divisions whose boundaries are not clear (*e.g.*, private courtyards with invisible boundaries), (2) functional zoning rules that assign different semantic classes to visually identical surfaces (*e.g.*, internal access roads labeled as unvegetated area), and (3) seasonal hydrological and vegetation changes that alter the visible extent of water bodies (*e.g.* vegetation overgrowth or water-level fluctuations exposing sandbars). These cases lack reliable visual cues and make boundary localization inherently ambiguous. In contrast, existing land-cover datasets such as LoveDA [17] or ISPRS Vaihingen/Potsdam [1, 2] primarily follow visual boundaries.

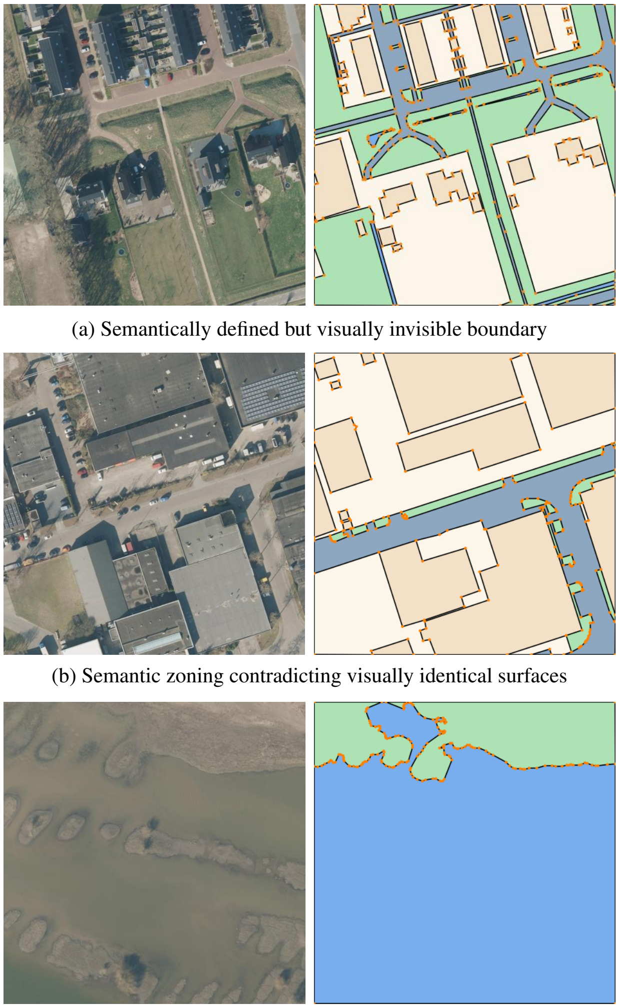
Representative examples are shown in Fig. S1.

Table S2. Challenging subsets in the test set. Each patch may belong to multiple subsets; percentages are computed relative to the entire test set.

Subset	#Patches	Ratio (%)
Shadow / Occlusion	194	88.2
Ambiguous Boundaries	94	42.7

2. Evaluation Metrics

We evaluate all methods using two complementary families of metrics: (i) *per-class accuracy metrics*, assessing semantic fidelity, geometric accuracy, and topological fidelity of individual polygons; and (ii) *global topology-consistency metrics*, evaluating whether the predicted polygons collectively form a valid planar partition without gaps or overlaps. In addition, for the ablation study (Sec. 6.3 of the main paper), we report *vertex–boundary alignment and peak-shape metrics* to quantify the quality of vertex response peaks.


(a) Semantically defined but visually invisible boundary

(b) Semantic zoning contradicting visually identical surfaces

(c) Seasonal hydrological changes misaligned with semantic water labels

Figure S1. Representative examples of semantically ambiguous boundaries caused by cadastral or administrative labeling rules. Each row corresponds to one discrepancy type described in Sec. 1.3.

2.1. Per-class Accuracy Metrics

These metrics measure how well the predicted polygons match ground-truth semantic regions, geometric boundaries, vertex usage, and connectivity structure. All metrics are computed per class and averaged over the test set.

Semantic fidelity. *IoU* (Intersection over Union) measures the region-overlap between predicted and ground-truth semantic masks (range $[0, 1]$, higher is better). *B-IoU* (Boundary IoU) [4] computes IoU within a thin band around ground-truth boundaries, emphasizing boundary-level localization accuracy (range $[0, 1]$, higher is better).

Geometric accuracy. *PoLiS* [3] is a symmetric vertex-to-boundary distance between two polygons. Given matched polygons A and B with vertex sequences $\{a_i\}_{i=1}^{n}$ and $\{b_j\}_{j=1}^{m}$, respectively, PoLiS computes the average minimal distance from each vertex of one polygon to the boundary of the other:

$$\begin{aligned} d_{\text{PoLiS}}(A,B) = \tfrac{1}{2}\Bigg(\tfrac{1}{n}\sum_{i=1}^{n} \operatorname{dist}(a_i, \partial B) \\ + \tfrac{1}{m}\sum_{j=1}^{m} \operatorname{dist}(b_j, \partial A) \Bigg), \end{aligned} \tag{S1}$$

where ∂A and ∂B denote the polygon boundaries, and $\operatorname{dist}(a_i, \partial B)$ (resp. $\operatorname{dist}(b_j, \partial A)$) is the minimal Euclidean distance from a_i (resp. b_j) to the edges of B (resp. A). Lower values indicate smaller boundary displacement between the two polygons.

MTA (Max Tangent Angle Error) [6] measures tangent-direction deviation between predicted and ground-truth contours. For each predicted contour C_p, tangent angles $\theta_p(x_k)$ are estimated at uniformly sampled boundary points x_k and compared to the tangent $\theta_g(y_k)$ at the closest ground-truth point y_k. The contour-level error is the maximum angular deviation:

$$\operatorname{MTA}(C_p, C_g) = \max_k \left|\operatorname{wrap}(\theta_p(x_k) - \theta_g(y_k))\right|, \tag{S2}$$

where $\operatorname{wrap}(\cdot)$ maps angle differences to $[-\pi, \pi]$. Lower values indicate better alignment of boundary tangent directions.

Both metrics are computed on matched polygon pairs, where a prediction is matched to the ground-truth polygon with IoU exceeding 0.5 (following HiSup [20]). For each metric, we report the mean value over all matched pairs in the test set.

Vertex efficiency. *N-ratio* measures the relative vertex usage between prediction and ground truth. For a matched pair (A, B) with N_A and N_B vertices,

$$\text{N-ratio}(A,B) = \frac{N_A}{N_B}, \tag{S3}$$

with values $\to 1$ being optimal: values > 1 indicate redundant vertices, while values < 1 indicate under-sampling.

C-IoU (Efficiency-aware IoU) [23] jointly evaluates region overlap and vertex-count efficiency. Let $\operatorname{IoU}(A, B)$ denote the region IoU. The relative vertex-count deviation is defined as

$$\operatorname{RD}(A,B) = \frac{|N_A - N_B|}{N_A + N_B}. \tag{S4}$$

The efficiency-aware IoU is then

$$\text{C-IoU}(A,B) = \operatorname{IoU}(A,B)\,\big(1 - \operatorname{RD}(A,B)\big). \tag{S5}$$

Higher values indicate both accurate region overlap and efficient vertex usage, penalizing polygons that are oversimplified or contain redundant vertices.

Topological fidelity. *APLS* (Average Path Length Similarity) [16] is used to assess graph-level connectivity for classes with elongated polygonal structures, namely roads and water. Since our outputs are polygon masks rather than explicit graphs, we first convert each class mask into a pseudo centerline graph following the skeletonization strategy of [8]. Concretely, we compute a 1-pixel-wide medial-axis skeleton from the binary mask; each skeleton pixel is treated as a graph node, and edges are inserted between 8-neighboring pixels, yielding a densely connected pseudo graph. To reduce redundancy and focus on semantically meaningful paths, we select all nodes with degree $\neq 2$ (endpoints and junctions) as control nodes and decompose the graph into simple polylines by tracing shortest paths between each pair of control nodes.

Given the ground-truth and predicted graphs G and G' constructed in this way, let $\mathcal{P}$ be the set of control-node pairs (a, b) in G with finite shortest-path length $L_G(a, b)$. For each $(a, b) \in \mathcal{P}$, we locate the nearest nodes a', b' in G' and define the normalized path-length discrepancy [16]

$$\delta(a,b) = \min\left(1, \; \frac{\left|L_G(a,b) - L_{G'}(a',b')\right|}{L_G(a,b)}\right). \tag{S6}$$

The APLS score is then

$$\operatorname{APLS}(G, G') = 1 - \frac{1}{|\mathcal{P}|} \sum_{(a,b)\in\mathcal{P}} \delta(a,b), \tag{S7}$$

ranging from 0 (poor) to 1 (perfect connectivity and path lengths) for the road and water graphs.

χ-err. Betti numbers are fundamental topological invariants that characterize k-dimensional “holes” in a topological space [18]. For a binary mask M, the 0-th and 1-st Betti numbers, $\beta_0(M)$ and $\beta_1(M)$, correspond to the number of connected components and interior holes. The Euler characteristic is a classical topological invariant that summarizes the global structure of a space by combining its Betti numbers through an alternating sum [14]. Following [10], we define the 2D specialization of the Euler characteristic as

$$\chi(M) = \beta_0(M) - \beta_1(M), \tag{S8}$$

and compute the Euler-characteristic error

$$\chi\text{-err}(A,B) = \left|\chi(A) - \chi(B)\right|. \tag{S9}$$

Lower values indicate that the predicted mask preserves the correct overall topological structure in terms of the aggregate balance between connected components and interior holes.

β-err. Betti-number discrepancies are a standard way to quantify topological errors in image segmentation [5]. Using the Betti numbers defined above, the Betti-number error between prediction A and ground truth B is

$$\beta\text{-err}(A, B) = \big|\beta_0(A) - \beta_0(B)\big| + \big|\beta_1(A) - \beta_1(B)\big|. \tag{S10}$$

Lower values indicate more faithful preservation of both connectivity (β_0) and interior-hole structure (β_1).

2.2. Global Topology-Consistency Metrics

These metrics assess whether the predicted polygons collectively form a topologically valid planar partition of the image domain.

Gap rate. The fraction of the image domain not covered by any predicted polygon. This captures missing regions in the partition (lower is better).

Inter-class overlap. The fraction of area simultaneously assigned to two or more different classes. This identifies semantic conflicts between classes (lower is better).

Intra-class overlap. The fraction of area overlapped by multiple polygons of the same class. This indicates boundary intersections or duplicated geometry within a single semantic region. (lower is better).

Shared-edge consistency. In a valid planar partition (a planar straight-line graph), every *interior* edge must be incident to exactly two polygons. We therefore measure the fraction of predicted interior edges that satisfy this two-sided incidence condition:

$$\text{SEC} = \frac{\#\{\text{interior edges shared by exactly two polygons}\}}{\#\{\text{all interior edges}\}}. \tag{S11}$$

Edges lying on the patch boundary are excluded. Higher values indicate stronger adherence to planar-partition topology.

2.3. Vertex–Boundary Alignment and Peak-Shape Metrics

In the ablation study (Sec. 6.3 of the main paper) we further analyze the quality of vertex response peaks. Let $\mathcal{V}$ be the set of predicted vertices, obtained by non-maximum suppression (NMS) on the vertex heatmap, and let $\mathcal{B}_c$ denote the predicted semantic boundaries of all classes (extracted as class-transition boundaries from the multi-class segmentation mask).

Vertex-to-boundary alignment (V2B@δ). We measure the fraction of predicted vertices that lie within a distance δ of their nearest predicted semantic boundaries. For each vertex $v \in \mathcal{V}$, we compute its minimal Euclidean distance to the predicted boundary,

$$d(v) = \min_{x \in \hat{\mathcal{B}}_{c(v)}} \operatorname{dist}(v, x), \tag{S12}$$

and define

$$\text{V2B}_\delta = \frac{\#\{\, v \in \mathcal{V} \mid d(v) \leq \delta \,\}}{|\mathcal{V}|}. \tag{S13}$$

$\text{V2B}_\delta \in [0, 1]$ (higher is better) quantifies how well the predicted vertex peaks align with the predicted class boundaries, a property crucial for reliable topological reconstruction.

Peak-shape metrics. For each detected peak location x_0, we extract a local $K \times K$ patch H from the corresponding vertex heatmap and normalize it so that $H(x_0) = 1$. We then characterize the local peak shape using three metrics commonly used in imaging and spectroscopy to quantify peak width and sharpness.

Area@0.5. We first threshold the patch at half maximum and consider the 8-connected component that contains the peak center x_0. Let $\mathcal{C}_{0.5} = \{x \in H \mid H(x) \geq 0.5\}$ be this component. We define

$$\text{Area@0.5} = \#\,\mathcal{C}_{0.5}, \tag{S14}$$

i.e., the number of pixels in the central half-maximum support (smaller values indicate more spatially concentrated peaks). Since the patch size $K \times K$ is fixed, areas are directly comparable across methods.

FWHM. To capture peak width and anisotropy, we measure axis-aligned full widths at half maximum along the horizontal and vertical profiles through x_0. Let p_x and p_y be the 1D profiles of H along the central row and column, respectively. We define FWHM_x (resp. FWHM_y) as the distance between the two points where p_x (resp. p_y) crosses 0.5, using linear interpolation between neighboring pixels. Lower values correspond to narrower peaks.

Sharpness. To quantify local curvature at the peak, we apply Gaussian smoothing with standard deviation σ to H and compute the Hessian at x_0. Let $\Delta H(x_0)$ denote the Laplacian of the smoothed patch at the center. We define a scale-normalized sharpness measure

$$\text{Sharpness} = \max\big(0,\, -\sigma^2 \Delta H(x_0)\big), \tag{S15}$$

so that sharper, more peaked responses yield higher values.

All peak-shape metrics are averaged over all detected peaks and are used only in our ablation analysis.

3. Method: Additional Details

3.1. SSC: Architecture Details

Semantic encoder. We instantiate S_ψ using a VMamba-based state-space backbone [11] operating at four spatial scales. Its multi-level features are unified by a UPerNet-style pyramid aggregation module [19], producing a single semantic feature map $S_\psi(I) \in \mathbb{R}^{C_s \times \frac{H}{4} \times \frac{W}{4}}$ aligned to the quarter-resolution grid of the image.

Segmentation head for explicit supervision. To provide direct supervision for S_ψ, we attach a lightweight segmentation head composed of two submodules. The first is a stack of three 3×3 Conv–BN–ReLU blocks that refine the semantic features. The second is a predictor consisting of a 3×3 convolution followed by a 1×1 projection. The output is then bilinearly upsampled to full resolution to produce per-pixel logits.

Conditioning fusion. The semantic feature map $S_\psi(I)$ is injected into the denoising network through channel-wise concatenation at the quarter-resolution latent scale. Empirically, cross-attention offered negligible improvement while incurring substantial computational overhead, so concatenation is adopted for all experiments.

Denoising UNet. Our denoiser follows the widely used UNet architecture employed in latent diffusion frameworks [15], but is substantially lightened to suit the ACPV setting. We use 128 base channels with channel multipliers $[1, 2, 2]$, a single residual block per resolution stage, and self-attention layers at resolutions 32×32 and 16×16 (attention resolutions 4 and 8 in the latent grid), with 32 head channels. The network receives a 640-channel input formed by concatenating the noisy latent vector with $S_\psi(I)$ and predicts the denoising target in the latent diffusion formulation. Despite being substantially reduced in depth and width, this lightweight denoiser provides sufficient capacity for the ACPV task, and deeper variants did not yield meaningful improvements in our experiments.

Latent Distributional Vertex Modeling We use the publicly released, pretrained KL-regularized variational autoencoder (VAE) employed in latent diffusion frameworks [15]. It maps an $H \times W$ heatmap to a $\frac{H}{4} \times \frac{W}{4} \times 4$ latent vector. Both the encoder and decoder remain frozen and simply provide a fixed mapping between pixel space and the latent space used for diffusion. All results reported in the paper use this default KL-VAE without modification.

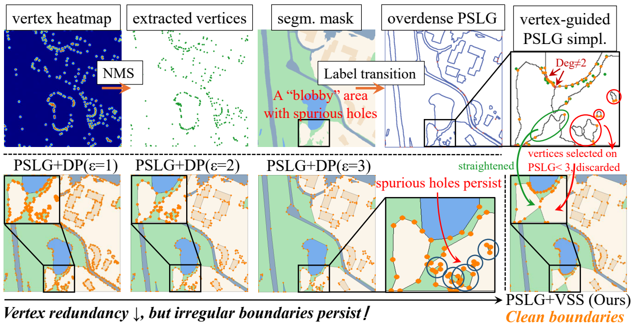


Figure S2. Illustration of the PSLG-based polygonization pipeline under segmentation noise.

3.2. PSLG Construction and Vertex-Guided Simplification

To further clarify the PSLG-based polygonization pipeline described in Sec. 4.2 of the main paper, we provide a more detailed explanation together with an illustrative example.

The planar straight-line graph (PSLG) used in ACPV is constructed from semantic label transitions in the segmentation mask. Pixels where semantic labels change are treated as vertices, and adjacent transitions on the image grid are connected to form edges, forming a planar graph aligned with semantic boundaries.

Redundant vertices are then removed through PSLG simplification guided by the predicted vertex heatmaps. The heatmaps provide sparse cues for selecting representative vertices along the PSLG using distance-based criteria, reducing vertex redundancy while preserving geometrically meaningful boundary points.

Polygon connectivity is subsequently obtained by graph traversal on the simplified PSLG rather than by directly linking predicted vertices, producing consistent polygon boundaries derived from the planar graph.

This PSLG-based polygonization is also more robust to imperfect segmentation masks. Segmentation predictions may contain artifacts such as blobby regions, fragmented areas, or spurious holes. Conventional polygonization methods (e.g., raster contour tracing followed by Douglas–Peucker simplification) tend to propagate such raster noise into polygon boundaries. In contrast, vertex-guided PSLG simplification suppresses these artifacts by selecting geometrically meaningful vertices on the planar graph.

Figure S2 illustrates the complete pipeline under segmentation noise and compares the proposed approach with DP-based contour tracing. The PSLG-based polygonization yields cleaner polygon boundaries and more stable vertex selection under noisy segmentation conditions.

3.3. Topological Reconstruction: Sufficient Condition Proof

For convenience, we restate Proposition 1 from the main paper and provide a more detailed justification.

Proposition 1 (Sufficient condition for ACPV compliance)**.** Let $G = (V, E)$ be a planar straight-line graph embedded in $\mathbb{R}^2$. Suppose that:

(i) Each edge $e \in E$ lies along a label-transition boundary in the multi-class mask $\hat{M}$, and the two faces adjacent to e have distinct semantic labels; and

(ii) The vertex set V consists exclusively of *geometric keypoints*, i.e., vertices whose removal would alter the graph's topology or geometric structure (junctions, salient corners, and endpoints of semantic transitions).

Then the polygonal partition obtained by tracing all face boundaries on G and assigning face labels according to $\hat{M}$ satisfies all ACPV constraints (a)–(f).

Proof sketch. Because G is a planar straight-line graph whose embedding includes the image boundary, it induces a planar subdivision $\mathcal{F}$ of the domain into finitely many bounded faces with pairwise–disjoint interiors. Tracing the boundary cycle of each face in $\mathcal{F}$ yields a simple polygon.

Condition (i) guarantees that every edge of G coincides with a semantic transition in $\hat{M}$ and is incident to two faces with distinct labels. Hence each face in $\mathcal{F}$ admits a unique semantic label compatible with all its boundary edges, producing a well-defined face labeling. Because the interiors of the faces are disjoint and their union equals the whole domain, the induced face labeling defines a gap-free and overlap-free partition.

Condition (ii) requires that the vertex set V contains all geometric keypoints, thereby preventing spurious degree-two vertices or self-intersecting boundary cycles. Consequently, each edge is shared by at most two faces with opposite orientations, implying consistent shared boundaries across classes.

Together, these properties imply that the labeled face decomposition of G forms a valid all-class planar partition satisfying all ACPV constraints (a)–(f).

4. Implementation and Training Details

This section summarizes all implementation details for ACPV-Net and the baseline methods, including dataset preprocessing, training schedules, hyperparameters, and fairness considerations. Our main experiments are conducted on two benchmarks, Deventer-512 and WHU-Building. All ablation studies are run on Deventer-512.

4.1. Datasets and Preprocessing

All models operate on fixed-size 512×512 patches. For Deventer-512, we directly use the official 512×512 patches and training/validation/test splits (Sec. 5 of the main paper), which contain 1716/219/220 patches with 5 semantic classes. For WHU-Building, we adopt the official split on the original images and then extract non-overlapping 512×512 patches, discarding patches without buildings, resulting in 9344/4015/5388 patches for training/validation/test.

For ACPV-Net, we use low-intensity geometric augmentation only: on Deventer-512 we apply random horizontal/vertical flips, and on WHU-Building we apply random horizontal/vertical flips plus random 90° rotations. For all baselines, we keep the authors' official implementations and their default augmentation pipelines, which typically include random flips and rotations and, for some methods, additional photometric jitter or stronger geometric transforms.

4.2. ACPV-Net Training Details

Training configurations differ slightly between Deventer-512 and WHU-Building due to dataset size. Unless otherwise stated, DDIM with T=200 steps is used for inference, and an exponential moving average (EMA) with decay 0.9999 is applied to all model weights.

Deventer-512. We train ACPV-Net for $160{,}000$ iterations with batch size 8. Optimization uses AdamW with learning rate 6×10^{-5}, (β_1, β_2)=$(0.9, 0.999)$, and weight decay 0.01. A linear warmup of $1{,}500$ iterations is followed by a PolyLR schedule with power 1.0 and minimum learning rate 0.

WHU-Building. Due to the larger training set, ACPV-Net is trained for $869{,}000$ iterations with batch size 8. We use the same AdamW configuration as above. Learning rate warmup lasts $8{,}150$ iterations, followed by the same PolyLR decay.

4.3. Baselines: Training Details

We evaluate five state-of-the-art polygonization methods: DeepSnake [13], FFL [6], TopDiG [21], HiSup [20], and GCP [22]. All models are trained from scratch on our datasets using the authors' official public implementations. We strictly follow their recommended hyperparameters, optimization settings, and training schedules. No architectural modifications are made.

DeepSnake. We use the publicly released implementation [12] and train the model for 100 epochs on both Deventer-512 and WHU-Building. Unlike the other baselines, DeepSnake supports multi-class training and is therefore trained once on the full Deventer-512 dataset rather than per semantic class.

FFL. Following the official implementation, we train each model for 1,000 epochs and select the checkpoint with the best validation performance. We apply this setting separately to every semantic class in Deventer-512 and to the WHU-Building dataset. For polygonization, we use the Active Skeleton Model (ASM) as recommended in the original paper.

TopDiG. We follow the official two-stage training protocol: (1) training the node-detection network, followed by (2) training the full TopDiG model, including the adjacency-prediction modules. Both stages use the authors' early-stopping strategy exactly as implemented in the public code.

A key task-specific hyperparameter is the maximum number of nodes allowed per image. For Deventer-512, we set this value per semantic class to 600 (buildings), 500 (roads), 700 (unvegetated), 400 (vegetation), and 200 (water), each chosen to cover the vast majority of image patches within its semantic class. For water bodies, a limit of 100 covers most patches but yields poor performance; we therefore use 200, which results in normal behavior for this class. For WHU-Building, we use a node limit of 300.

HiSup. Following the official implementation, HiSup is trained for 30 epochs without early stopping on every semantic class of Deventer-512 and WHU-Building.

GCP. Following the official implementation, GCP is trained in two stages. (1) The Mask2Former backbone is trained for 50 epochs as the base instance segmentation model. (2) The polygonization modules are then trained for 12 epochs using the recommended number of queries (Q=300). The same two-stage schedule is applied to every semantic class of Deventer-512 and WHU-Building.

Fairness. For every baseline, we keep the original optimization settings (learning rate, scheduler, batch size, optimizer, augmentation) as released by the authors. This ensures that each method is evaluated under its intended operating regime without re-tuning or weakening.

5. Extended Experiments

5.1. Per-Class Topology Statistics

To complement the global topology evaluation in Table 7 of the main paper we further report per-class intra-class overlap rates in Table S3. While inter-class overlaps and gaps can arise when each semantic class is polygonized independently, intra-class overlaps are not expected, since these methods are specifically designed for single-class outline extraction. Such intra-class violations indicate topological errors within the category, such as self-intersections or other invalid polygon representations.

Table S3. **Per-class intra-class overlap rates (%) on Deventer-512.** Best values are shown in **bold**. All percentage values are reported to two decimal places.

Method	Building $\downarrow$	Road $\downarrow$	Unvegetated $\downarrow$	Vegetation $\downarrow$	Water $\downarrow$
DeepSnake [13]	5.92	12.33	21.14	28.76	19.55
FFL [6]	**0.00**	0.01	0.06	0.01	**0.00**
TopDiG [21]	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
HiSup [20]	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
GCP [22]	6.54	0.31	6.91	29.05	0.08
ACPV-Net (Ours)	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**

ACPV-Net achieves zero measured intra-class overlap across all categories in our evaluation, indicating that its outputs are topologically consistent both within each semantic class and at the global planar-partition level.

5.2. Robustness to NMS Thresholds

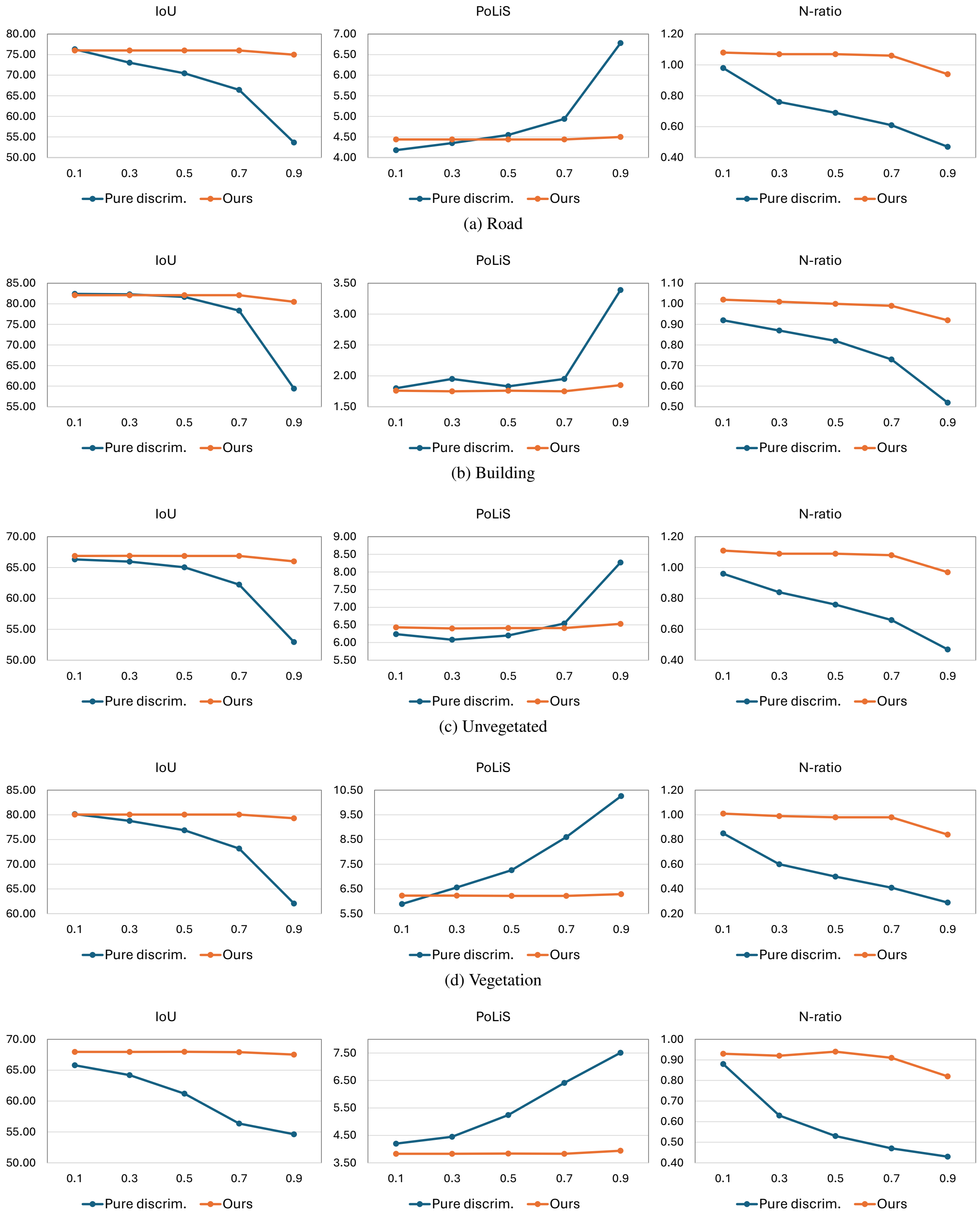
To further evaluate the stability of distributional vertex modeling, we analyze the sensitivity of polygonal metrics with respect to the NMS threshold τ used for extracting vertices from the predicted heatmaps. The NMS kernel size is fixed to 3×3, matching the spatial scale of the Gaussian kernels used to construct ground-truth vertex heatmaps, and therefore does not constitute a tunable hyperparameter.

Figure S3 reports per-class sensitivity curves for the distributional model and the pure discriminative decoder. The distributional model exhibits minimal variation across a wide range of NMS thresholds, whereas the discriminative baseline shows pronounced fluctuations in IoU (semantic fidelity, $\uparrow$ better), PoLiS (geometric accuracy, $\downarrow$ better), and N-ratio (vertex efficiency, $\rightarrow 1$ better). This observation is consistent with the peak-shape analysis in Sec. 6.3 of the main paper: sharper and more compact peaks produced by latent diffusion lead to more stable vertex extraction and consequently more robust polygon reconstruction.

5.3. VSS vs. DP: Pareto Fronts

To complement the discussion in Sec. 6.3 of the main paper, we compare our vertex-guided subset selection (VSS) with the classical Douglas–Peucker (DP) simplification. For each method, we sweep its respective control parameter (NMS threshold for VSS; tolerance ε for DP) and plot the resulting class-averaged trade-off between geometric error (PoLiS, lower is better) and symmetric redundancy $R = \max(\text{N-ratio}, 1/\text{N-ratio})$ (closer to 1 is better).

Figure S4 shows that VSS consistently achieves lower redundancy at comparable or lower PoLiS, indicating that learned vertex cues provide a more effective basis for keypoint preservation than purely geometric simplification. The diamond marker denotes the extreme case $\tau = 0.9$ for VSS.



(a) Road

(b) Building

(c) Unvegetated

(d) Vegetation

(e) Water

Figure S3. Sensitivity of polygonal quality to the NMS threshold used for vertex extraction. Performance of pure discriminative decoder vs. our distributional vertex modeling measured in IoU (semantic fidelity, ↑ better), PoLiS (geometric accuracy, ↓ better), and N-ratio (vertex efficiency, → 1 better).

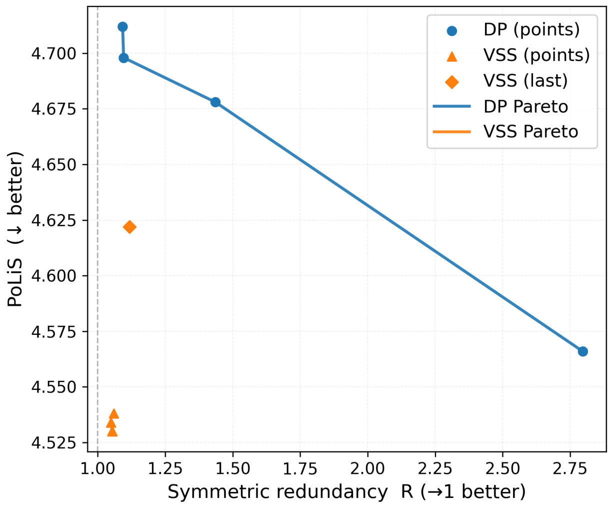


Figure S4. Class-averaged Pareto fronts between geometry error (PoLiS, ↓ better) and symmetric redundancy $R = \max(\text{N-ratio}, 1/\text{N-ratio})$ ($\rightarrow 1$ better) for *PSLG+DP (w/o VSS)* and *PSLG+VSS (ours)*. The diamond marker denotes the extreme case ($\tau = 0.9$) for VSS.

5.4. Sanity Check for the Frozen KL-VAE Encoder–Decoder

To ensure that using a pretrained and frozen KL-VAE does not distort the vertex heatmaps used by ACPV-Net, we perform a direct encoder–decoder reconstruction test. Each ground-truth vertex heatmap is passed through the frozen VAE encoder to obtain a latent representation, which is then decoded back to the heatmap space. Representative examples are shown in Fig. S5.

The reconstructed heatmaps faithfully preserve peak location, anisotropy, and contrast, with negligible visual degradation. This confirms that the frozen KL-VAE has sufficient capacity for representing vertex heatmaps and does not introduce task-relevant artifacts.

5.5. Cross-region Generalization to External Aerial Datasets

To assess cross-region generalization, we apply the model trained on Deventer-512 directly (no fine-tuning) to aerial imagery from multiple countries and independent national mapping agencies. Specifically, we evaluate on orthorectified RGB aerial photographs at $0.3\,\mathrm{m}$ ground resolution from (i) Eastern Tyrol and Innsbruck in western Austria, acquired through the Austrian federal aerial survey program, and (ii) Bellingham in Washington State, USA, obtained from local governmental aerial surveys. We further test on urban aerial imagery from Christchurch, New Zealand, captured by the national LINZ aerial mapping program and released at $0.3\,\mathrm{m}$ resolution.

Figs S6 and S7 show representative qualitative results. Although polygon regularity naturally degrades under domain shift (different countries, urban layouts, seasons, and illumination), ACPV-Net still produces globally coherent vector basemaps with consistent shared boundaries and no visually apparent gaps or overlaps in these examples. These observations indicate that, through semantically supervised conditioning, the model learns geometric primitives that are not tied to dataset-specific visual cues, but capture a transferable distribution over aerial-image vertices and cadastral-style partitions.

At the dataset level, these experiments also indicate that, despite its compact spatial extent, Deventer-512 provides sufficiently diverse geometry and visual conditions for training models that generalize reasonably to other high-resolution aerial regions. We view Deventer-512 as a compact yet challenging testbed on which future methods can benchmark both in-domain performance and cross-region generalization.

5.6. Evaluation on the Shanghai dataset

To further evaluate the generalization ability of ACPV-Net on other public datasets, we report additional experiments on the Shanghai dataset used in HiSup [20]. We apply the official HiSup predictions and metrics with our comprehensive evaluation.

Table S4 summarizes the quantitative comparison. ACPV-Net achieves competitive or superior performance across most metrics, particularly on polygon-level metrics such as CIoU and PoLiS.

5.7. TopDiG metrics on Deventer-512

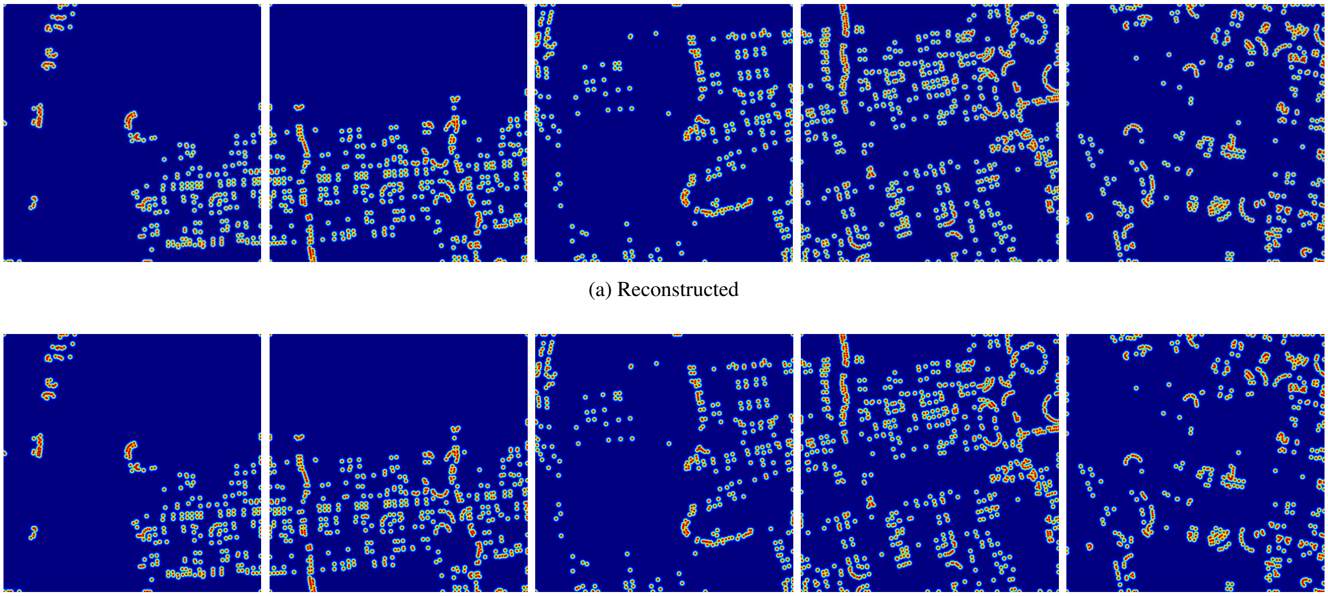
To enable a more direct comparison with state-of-the-art methods such as TopDiG [21], we additionally report the TopDiG evaluation metrics on Deventer-512. These metrics mainly evaluate pixel-level and boundary-based performance, which differs from the polygon-level metrics adopted in ACPV.

Direct comparison on the datasets used in TopDiG is not applicable because TopDiG targets directional graph extraction and is primarily evaluated on semantic segmentation datasets without polygonal ground truth, whereas ACPV requires vectorized polygons for polygon-level evaluation.

Table S5 presents the macro-averaged results. ACPV-Net achieves consistently better performance than TopDiG across the reported metrics.

6. Additional Qualitative Visualizations

To complement the quantitative evaluations in the main paper, we present representative qualitative examples across the principal sources of geometric and visual difficulty in Deventer-512. For each scenario category, figures in this



(a) Reconstructed

(b) Ground truth

Figure S5. Sanity check for the frozen KL-VAE Encoder-Decoder. Vertex heatmaps are reconstructed with highly preserved peak location, anisotropy, and contrast.

Table S4. Quantitative results on the Shanghai dataset. For baseline methods, AP, AP$_{boundary}$, and CIoU are taken from the HiSup paper, while the remaining metrics are computed using the official HiSup predictions with our evaluation protocol.

Method	AP↑	AP$_{boundary}$↑	CIoU↑	IoU↑	BIoU↑	PoLiS↓	MTA↓
PolyMapper	4.3	0.8	28.6	–	–	–	–
FFL	16.3	3.9	16.4	–	–	–	–
HiSup	45.0	19.4	52.8	74.9	56.8	2.4	41.3
Ours	42.1	19.4	64.9	76.7	58.6	2.2	39.2

section provide side–by–side comparisons between ACPV-Net and the strongest polygonization baselines (HiSup [20], TopDiG [21], and GCP [22]), illustrating differences in shared-boundary consistency, vertex localization, and global topology.

6.1. Multi-Class Geometric Complexity

We first show examples containing dense mixtures of buildings, roads, vegetation, water, and unvegetated regions (Fig. S8). These include large-scale unvegetated parcels with multiple interior holes, nested polygons, and complex multi-class adjacencies. ACPV-Net produces a coherent planar partition with clean shared boundaries, whereas stitched single-class baselines frequently introduce gaps, inter-class overlaps, or broken interior rings.

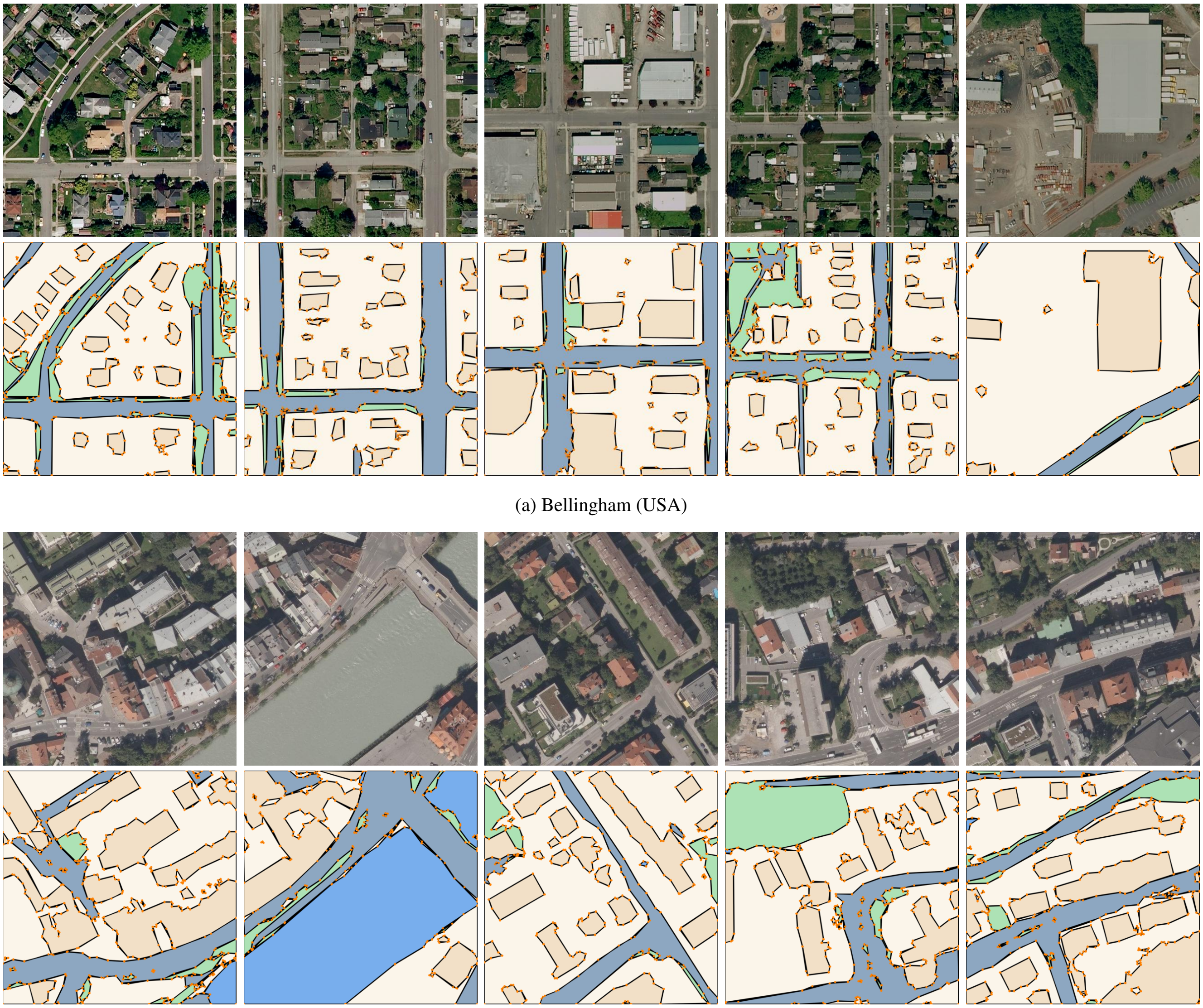
6.2. Thin and Topologically Fragile Structures

Fig. S9 presents narrow and fragile structures such as thin roads, elongated water bodies, vegetation strips, and small garden beds. These geometries are extremely sensitive to the boundary localization errors. ACPV-Net preserves connectivity and topological integrity, while baselines commonly exhibit breakage, merging, or self-intersecting segments in these cases.

6.3. Challenging Visual Conditions (Successes and Limitations)

In Fig. S10, we collect examples with strong shadows, weak or missing texture, reflective surfaces, dense vegetation boundaries, and cadastral–visual mismatches where the true boundary is not visually implied. ACPV-Net remains robust under most of these adverse conditions, although large cadastral–visual discrepancies can still degrade polygon regularity. These visualizations highlight both the strengths of semantically supervised conditioning and the remaining failure modes inherent to ambiguous imagery.

Limitation. As shown in Fig. S10, strong occlusion and



(a) Bellingham (USA)

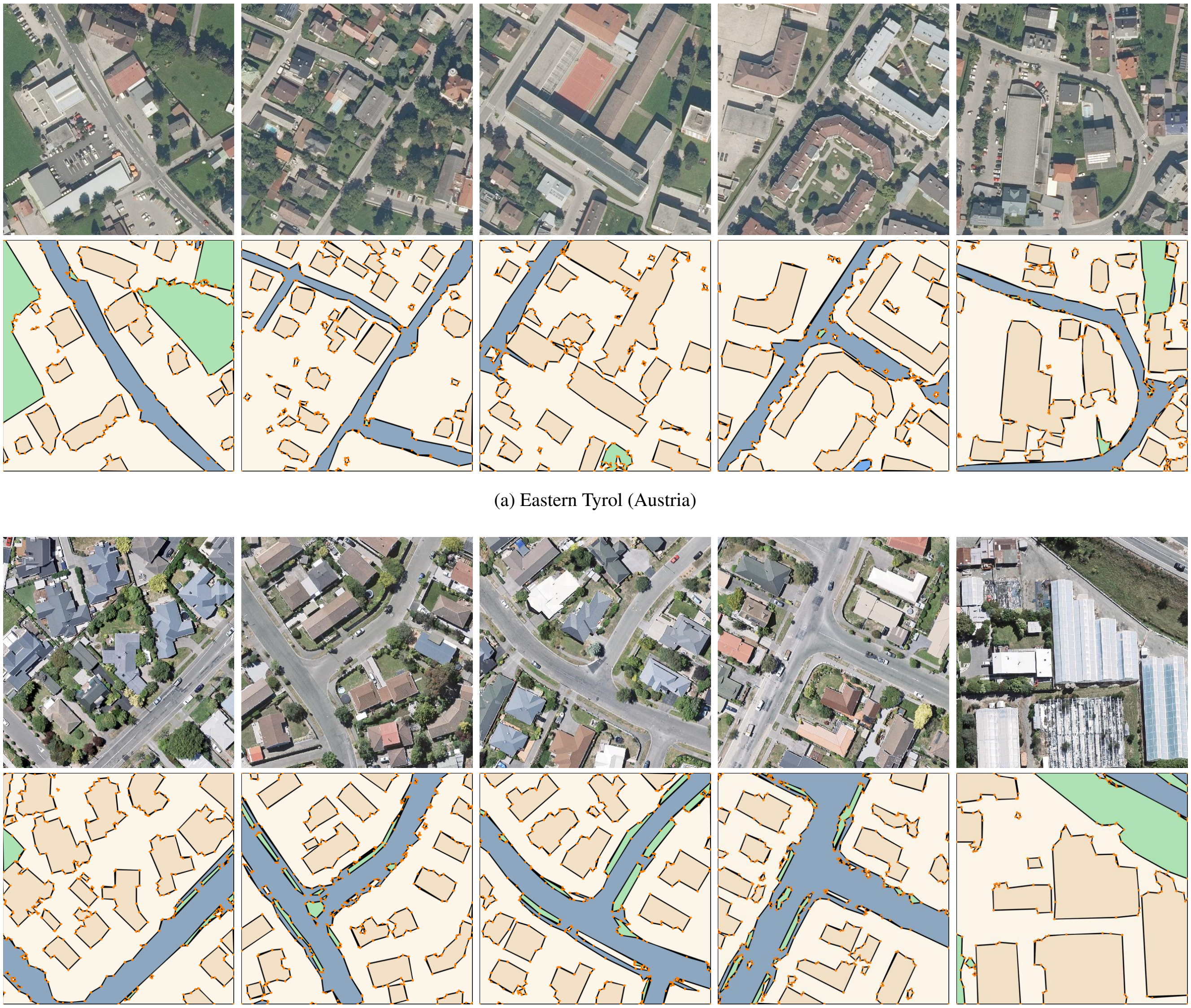
(b) Innsbruck (Austria)

Figure S6. **Cross-region qualitative results (Part 1).** Polygons of different semantic classes are shown in distinct colors. Predicted vertices are marked as orange points. Topological inconsistencies, if present, would appear in black (gaps) or red (overlaps). These examples illustrate ACPV-Net's behavior on imagery from independent national mapping agencies **without fine-tuning**.

Table S5. TopDiG metrics on Deventer-512 ($\delta = 5$).

Method	PA_{mask}	F1_{mask}	mIoU_{mask}	PA_{topo}	F1_{topo}	mIoU_{topo}	APLS
TopDiG	91.4	72.1	58.0	90.4	49.6	34.6	50.7
Ours	96.4	89.8	81.6	94.2	74.9	60.4	76.2

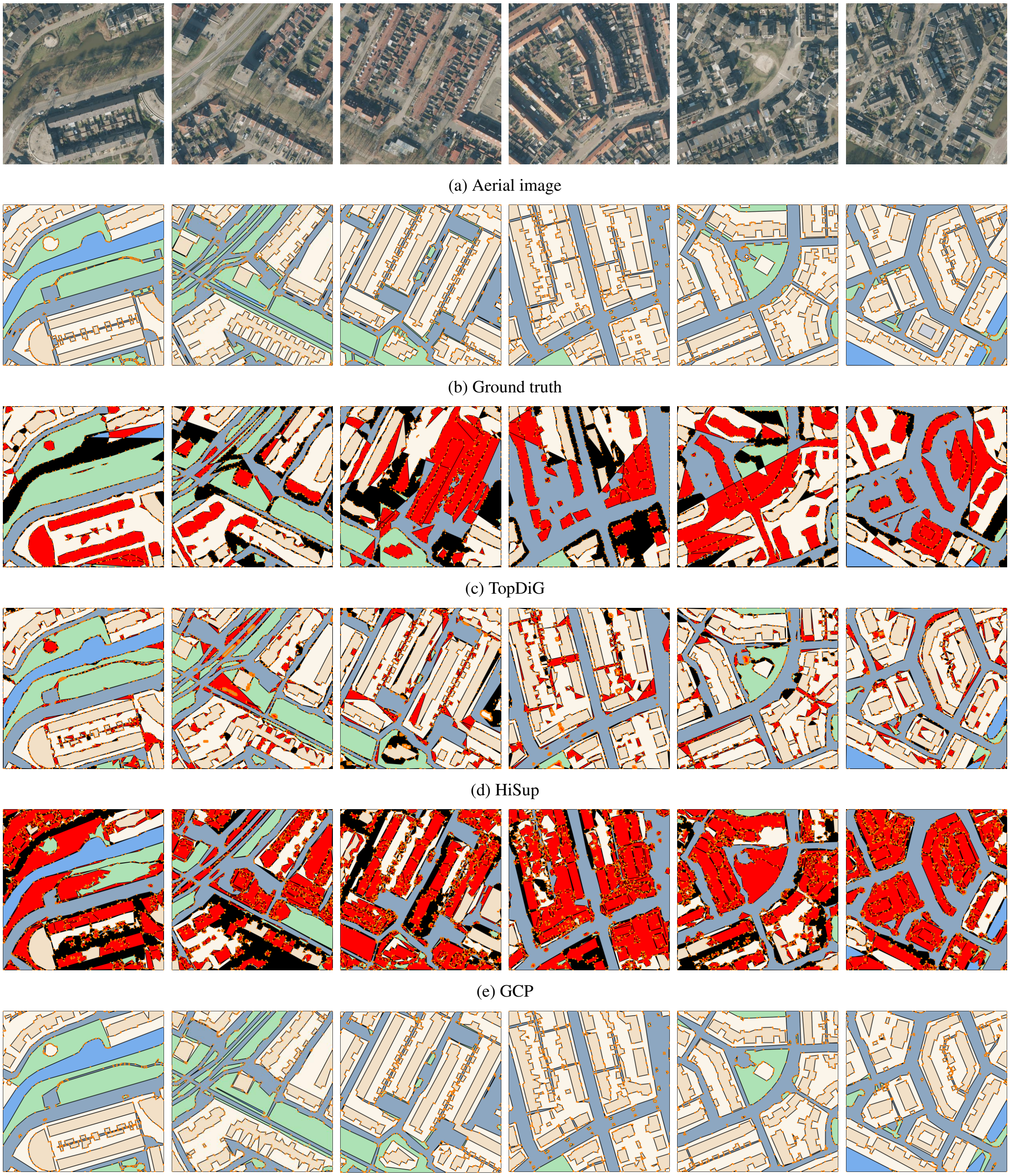
cadastral-visual mismatch can lead to suboptimal performance. Also, the semantically ambiguous boundaries as shown in Sec. 1.3 make the polygon vectorization challenging. In future work, we will seek uncertainty quantification to outline such cases, in order to guide the correction with minimum human effort.



(a) Eastern Tyrol (Austria)

(b) Christchurch (New Zealand)

Figure S7. **Cross-region qualitative results (Part 2).** Polygons of different semantic classes are shown in distinct colors. Predicted vertices are marked as orange points. Topological inconsistencies, if present, would appear in black (gaps) or red (overlaps). These examples illustrate ACPV-Net's behavior on imagery from independent national mapping agencies **without fine-tuning**.



(a) Aerial image

(b) Ground truth

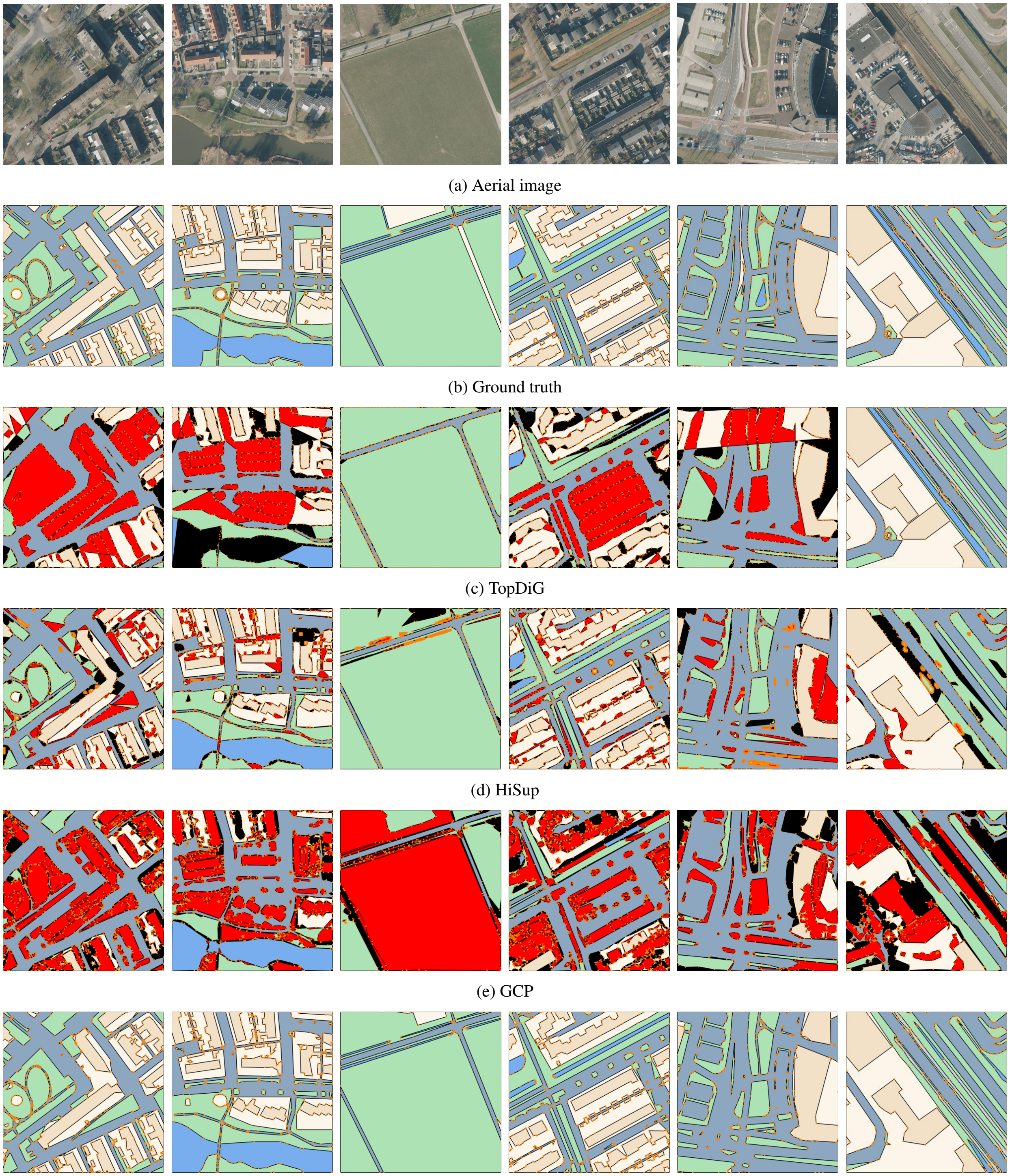
(c) TopDiG

(d) HiSup

(e) GCP

(f) Ours

Figure S8. **Examples of multi-class geometric complexity.** Polygons of different semantic classes are shown in distinct colors. Predicted vertices are marked as orange points. Topological inconsistencies, if present, would appear in black (gaps) or red (overlaps).



(a) Aerial image

(b) Ground truth

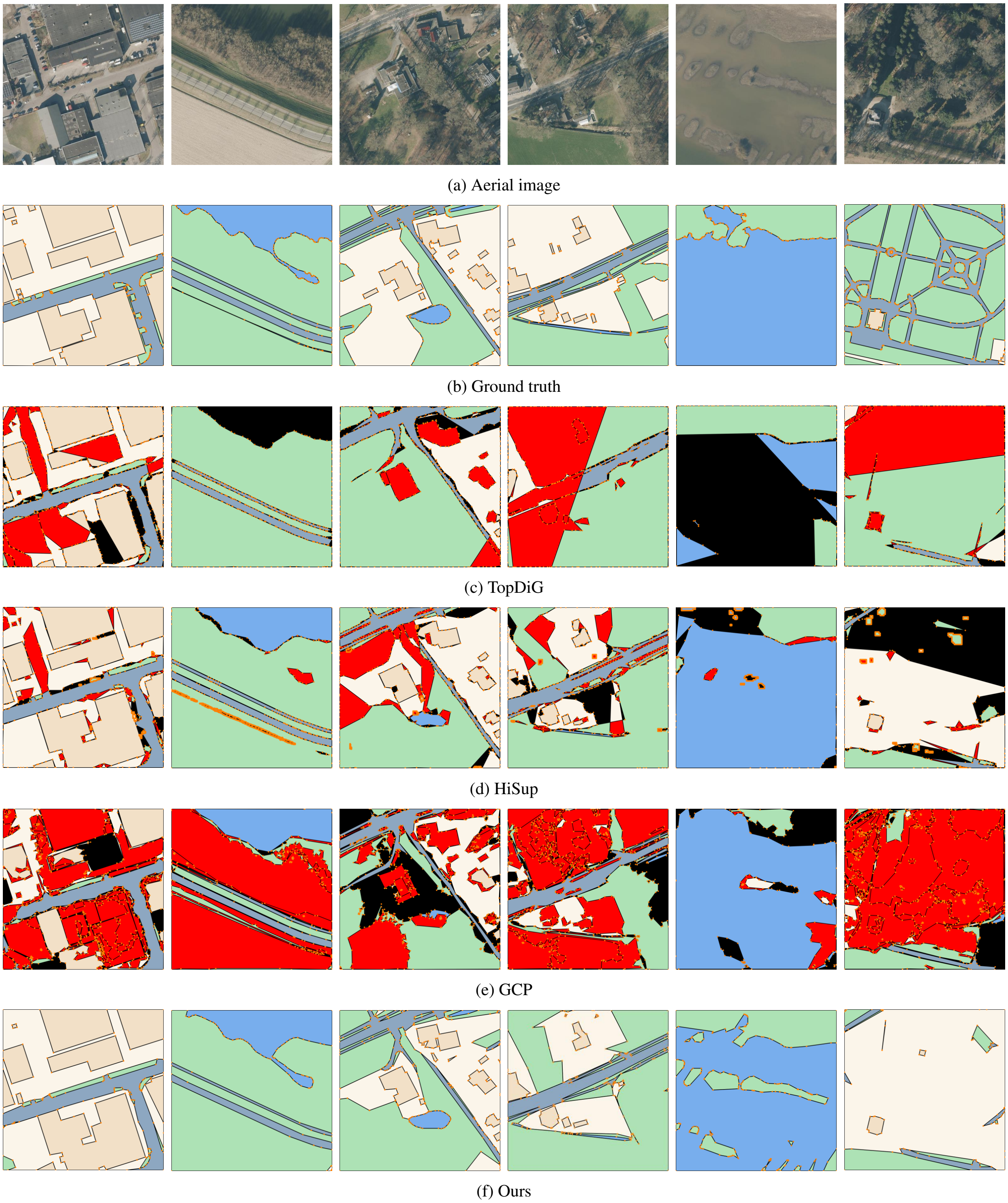
(c) TopDiG

(d) HiSup

(e) GCP

(f) Ours

Figure S9. **Examples of thin and topologically fragile Structures.** Polygons of different semantic classes are shown in distinct colors. Predicted vertices are marked as orange points. Topological inconsistencies, if present, would appear in black (gaps) or red (overlaps).



(a) Aerial image

(b) Ground truth

(c) TopDiG

(d) HiSup

(e) GCP

(f) Ours

Figure S10. **Examples of challenging visual conditions.** Polygons of different semantic classes are shown in distinct colors. Predicted vertices are marked as orange points. Topological inconsistencies, if present, would appear in black (gaps) or red (overlaps).

References

[1] Isprs 2d semantic labeling - potsdam. https://www.isprs.org/resources/datasets/benchmarks/UrbanSemLab/2d-sem-label-potsdam.aspx, . Accessed: 2025-11-10. 2

[2] Isprs 2d semantic labeling - vaihingen. https://www.isprs.org/resources/datasets/benchmarks/UrbanSemLab/2d-sem-label-vaihingen.aspx, . Accessed: 2025-11-10. 2

[3] Janja Avbelj, Rupert Müller, and Richard Bamler. A metric for polygon comparison and building extraction evaluation. *IEEE Geoscience and Remote Sensing Letters*, 12(1):170–174, 2014. 3

[4] Bowen Cheng, Ross Girshick, Piotr Dollár, Alexander C Berg, and Alexander Kirillov. Boundary iou: Improving object-centric image segmentation evaluation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15334–15342, 2021. 2

[5] James R Clough, Nicholas Byrne, Ilkay Oksuz, Veronika A Zimmer, Julia A Schnabel, and Andrew P King. A topological loss function for deep-learning based image segmentation using persistent homology. *IEEE transactions on pattern analysis and machine intelligence*, 44(12):8766–8778, 2020. 4

[6] Nicolas Girard, Dmitriy Smirnov, Justin Solomon, and Yuliya Tarabalka. Polygonal building extraction by frame field learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5891–5900, 2021. 3, 6, 7

[7] Jeroen Grift, Claudio Persello, and Mila Koeva. Cadastrevision: A benchmark dataset for cadastral boundary delineation from multi-resolution earth observation images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 217: 91–100, 2024. 1

[8] Weiqin Jiao, Hao Cheng, George Vosselman, and Claudio Persello. Ldpoly: Latent diffusion for polygonal road outline extraction in large-scale topographic mapping. *ISPRS Journal of Photogrammetry and Remote Sensing*, 230:820–842, 2025. 3

[9] Kadaster. Dataset: Basisregistratie Grootschalige Topografie (BGT). https://www.pdok.nl/introductie/-/article/basisregistratie-grootschalige-topografie-bgt-, 2022. Accessed: 2025-11-21. 1

[10] Liu Li, Qiang Ma, Cheng Oyang, Johannes C Paetzold, Daniel Rueckert, and Bernhard Kainz. Topology optimization in medical image segmentation with fast χ euler characteristic. *IEEE Transactions on Medical Imaging*, 2025. 3

[11] Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, Jianbin Jiao, and Yunfan Liu. Vmamba: Visual state space model. *Advances in neural information processing systems*, 37:103031–103063, 2024. 5

[12] Zichen Liu, Jun Hao Liew, Xiangyu Chen, and Jiashi Feng. Dance: A deep attentive contour model for efficient instance segmentation. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 345–354, 2021. 6

[13] Sida Peng, Wen Jiang, Huaijin Pi, Xiuli Li, Hujun Bao, and Xiaowei Zhou. Deep snake for real-time instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8533–8542, 2020. 6, 7

[14] Eitan Richardson and Michael Werman. Efficient classification using the euler characteristic. *Pattern Recognition Letters*, 49:99–106, 2014. 3

[15] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 5

[16] Adam Van Etten, Dave Lindenbaum, and Todd M Bacastow. Spacenet: A remote sensing dataset and challenge series. *arXiv preprint arXiv:1807.01232*, 2018. 3

[17] Junjue Wang, Zhuo Zheng, Ailong Ma, Xiaoyan Lu, and Yanfei Zhong. Loveda: A remote sensing land-cover dataset for domain adaptive semantic segmentation. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*. Curran Associates, Inc., 2021. 2

[18] Larry Wasserman. Topological data analysis. *Annual review of statistics and its application*, 5(2018):501–532, 2018. 3

[19] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *Proceedings of the European conference on computer vision (ECCV)*, pages 418–434, 2018. 5

[20] Bowen Xu, Jiakun Xu, Nan Xue, and Gui-Song Xia. Hisup: Accurate polygonal mapping of buildings in satellite imagery with hierarchical supervision. *ISPRS Journal of Photogrammetry and Remote Sensing*, 198:284–296, 2023. 3, 6, 7, 9, 10

[21] Bingnan Yang, Mi Zhang, Zhan Zhang, Zhili Zhang, and Xiangyun Hu. Topdig: Class-agnostic topological directional graph extraction from remote sensing images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1265–1274, 2023. 6, 7, 9, 10

[22] Fahong Zhang, Yilei Shi, and Xiao Xiang Zhu. Global collinearity-aware polygonizer for polygonal building mapping in remote sensing. *IEEE Transactions on Geoscience and Remote Sensing*, 2025. 6, 7, 10

[23] Stefano Zorzi, Shabab Bazrafkan, Stefan Habenschuss, and Friedrich Fraundorfer. Polyworld: Polygonal building extraction with graph neural networks in satellite images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1848–1857, 2022. 3