HoliTracer: Holistic Vectorization of Geographic Objects from Large-Size Remote Sensing Imagery

Supplementary Material

1. MCR Algorithm Details

The MCR algorithm is detailed in Algorithm 1, which is used to reconstruct polygon contours and align them with ground truth polygons.

Algorithm 1 Mask Contour Reformer

```
Input: $S, G = [g_1, g_2, \dots, g_M], \epsilon, l$
Output: $R, G', C$
$S' = [s'_1, s'_2, \dots, s'_m]$          ▷ DP simplification with $\epsilon$
For each $(s'_i, s'_{i+1})$ in $S'$ (with $s'_{m+1} = s'_1$ if closed):
    $\vec{v}_i = s'_{i+1} - s'_i$, $q_i = \|\vec{v}_i\|$, $\hat{v}_i = \vec{v}_i / q_i$
    $K_i = \lfloor q_i / l \rfloor$
    $[s'_i + kd\hat{v}_i$ for $k = 0$ to $K_i] + [s'_{i+1}]$
$R = [r_1, r_2, \dots, r_N]$          ▷ Concatenate all points
For each $g_j \in G$:
    $r_{k_j} = \arg\min_{r \in R} \|r - g_j\|$
Vertices: $\{r_{k_j} | j = 1, 2, \dots, M\}$
Indices: $i_1 < i_2 < \cdots < i_M$ in $R$
For each $k = 1$ to $M$ (with $i_{M+1} = i_1$ if closed):
    $n_k = i_{k+1} - i_k - 1$
    $p_{k,m} = g_k + \frac{m}{n_k+1}(g_{k+1} - g_k)$ for $m = 1$ to $n_k$
$G' = [g_1, p_{1,1}, \dots, p_{1,n_1}, g_2, \dots, g_M, p_{M,1}, \dots, p_{M,n_M}]$
$C = [c_1, \dots, c_N]$, $c_i = 1$ if $r_i$ is a vertex, else 0
Return $R, G', C$
```

2. Metrics for Evaluation

For a comprehensive assessment of semantic segmentation, instance segmentation, and vector generation quality, we report three widely used categories of metrics: semantic metrics, instance metrics, and vector metrics.

Vector Metrics. Vector metrics include PoLiS [1] and Complexity-aware IoU (C-IoU) [9]. For two given polygons A and B, PoLiS is defined as the average distance between each vertex $a_j \in A$, $j = 1, \dots, q$, of A and its closest point (b) on the boundary ∂B, and vice versa. Assuming polygon B has vertices $b_k \in B$, $k = 1, \dots, r$, the PoLiS metric [1] is expressed as:

$$\begin{aligned} \text{PoLiS}(A, B) = & \frac{1}{2q} \sum_{a_j \in A} \min_{b \in \partial B} \|a_j - b\| \\ & + \frac{1}{2r} \sum_{b_k \in B} \min_{a \in \partial A} \|b_k - a\| \,, \end{aligned} \tag{1}$$

where $\frac{1}{2q}$ and $\frac{1}{2r}$ are normalization factors. The IoU threshold for filtering predicted building polygons is set to 0.5, following [8]. A lower PoLiS value indicates greater similarity between predicted and ground truth polygons. The Complexity-aware IoU (C-IoU) [9] is also computed for polygon evaluation, defined as:

$$\text{C-IoU}(A, B) = \text{IoU}(A_m, B_m) \cdot (1 - \text{RD}(N_A, N_B)) \,, \tag{2}$$

where $\text{IoU}(A_m, B_m)$ denotes the standard IoU between the polygon masks A_m and B_m, and $\text{RD}(N_A, N_B) = |N_A - N_B| / (N_A + N_B)$ represents the relative difference between the number of vertices N_A in polygon A and N_B in polygon B. C-IoU balances segmentation and polygonization accuracy while penalizing both oversimplified and overly complex polygons relative to the ground truth complexity.

For the road dataset, we also report the Average Path Length Similarity (APLS) metric [6], which measures road network similarity by comparing the path lengths between node pairs in the predicted and ground truth graphs. The APLS metric is defined as follows. Given a ground-truth graph G_{gt} and a predicted graph G_{pred}, the APLS is computed based on the symmetric difference of shortest path lengths between all reachable pairs of nodes in both graphs:

$$\text{APLS} = 1 - \frac{1}{|P|} \sum_{(i,j) \in P} \frac{|d_{pred}(i,j) - d_{gt}(i,j)|}{d_{gt}(i,j)}$$

where P is the set of all node pairs (i, j) with valid paths in G_{gt}, $d_{gt}(i, j)$ is the shortest path length between nodes i and j in the ground truth graph, and $d_{pred}(i, j)$ is the corresponding path in the predicted graph. The APLS value ranges from 0 to 1, with higher values indicating better topological alignment.

Instance Metrics. Instance metrics adopt the standard COCO measure, mean Average Precision (AP), calculated over multiple Intersection over Union (IoU) thresholds. AP is averaged across ten IoU values ranging from 0.50 to 0.95 with a step size of 0.05, rewarding detectors with better localization. Additionally, $AP_{(S,M,L)}$ is used to evaluate performance on objects of different sizes. Given that geographical instances occupy more pixels in large-size very-high-resolution (VHR) remote sensing images, we redefine size categories relative to the COCO standard: small, medium, and large correspond to areas $< 128^2$, between 128^2 and 512^2, and $> 512^2$ pixels, respectively, where the area is measured as the number of pixels in the segmentation mask.

Semantic Metrics. Semantic metrics include the F1-score and Mean Intersection over Union (MIoU). The F1-score, the harmonic mean of Precision and Recall, provides

Table 1. Comparison with segmentation method on WHU-Building.

Method	PoLiS ↓	CIoU	AP	APs	APm	APl	IoU	F1
HRNet + DP	6.10	50.01	48.13	26.58	70.59	38.60	86.51	92.68
HRNet + PST	5.91	61.51	49.01	26.59	71.30	50.74	86.48	92.67
CAN + DP	4.02	60.32	58.42	37.50	79.76	49.13	91.50	**95.52**
CAN + PST	**3.63**	**82.30**	**61.07**	**40.37**	**80.30**	**60.00**	**91.60**	95.41

Table 2. APLS metric comparison with various vectorization methods.

Metric	TS-MTA	LCF-ALE	DeepSnake	E2EC	FFL	UniVec	HiSup	Ours
APLS	14.49	15.67	11.26	11.59	12.68	10.22	20.46	**28.35**

a comprehensive evaluation of model performance. MIoU, the average ratio of intersection to union between predicted and ground truth segmentation results, reflects the model's overall segmentation effectiveness across the entire image.

3. Implementation Details

Details of VHR-road Dataset. The VHR-road dataset is comprised of high-resolution remote sensing imagery of major urban areas in France, acquired from BD ORTHO [4]. The corresponding raw road labels are sourced from European Union's Copernicus Land Monitoring Service information [3]. We subsequently filter and rectify inaccuracies within these labels, culminating in a final dataset of 208 image tiles, each with a dimension of 12500×12500 pixels.

Hyperparameter Settings. When constructing the multi-scale pyramid, we set the scale factors d to $\{1, 3, 6\}$ for buildings and $\{1, 5, 10\}$ for water bodies and roads. During boundary point reconstruction, the Douglas-Peucker simplification parameter ϵ is set to 5. For small targets such as buildings, the interpolation distance l is set to 25, while for larger targets like water bodies and roads, it is set to 50.

Training Process. HoliTracer involves two training processes. For CAN training, we employ the Adam optimizer with a learning rate of 0.0001. For PST training, we use the Adam optimizer with a learning rate of 0.01. All loss function hyperparameters are set to 1, and the angle penalty term $\theta_{\text{threshold}}$ is fixed at 135 degrees. All experiments are conducted using the PyTorch framework on four NVIDIA A100 GPUs.

4. Supplementary Experiments

Comparison with Segmentation Methods and Flexibility of PST. Table 1 presents direct comparisons with the segmentation-based method HRNet [7], showing the superior performance of our Context Attention Net (CAN). Although the main focus is on vectorization methods, segmentation-based approaches are also considered, such as Hisup, which uses HRNet as its backbone. To evaluate the flexibility of the proposed PST, the table also reports results for HRNet+PST and HRNet+DP simplification. PST consistently achieves better performance in vectorizing general segmentation masks. While CAN enhances PST's performance on large-size RSI within the complete HoliTracer pipeline, these results demonstrate PST's general utility across different segmentation outputs.

Evaluation with APLS Metric. To further evaluate the quality of road network vectorization, the Average Path Length Similarity (APLS) metric is adopted. Table 2 reports the APLS scores for different methods on the road dataset.

Computational Efficiency and Scalability. Table 3 summarizes the computational complexity and inference performance of HoliTracer. Although HoliTracer has more parameters and a higher computational load compared to lightweight methods, the overhead remains acceptable for vectorization tasks, which typically do not require real-time processing. HoliTracer processes large images directly, eliminating the need for patch-wise inference and subsequent stitching. This design reduces total inference time. Scalability experiments confirm that HoliTracer handles images up to $40{,}000 \times 50{,}000$ pixels using 64GB of CPU RAM. Our implementation also includes GPU-based parallelism for inference on large images, providing robust scalability across different computational environments.

5. Supplementary Ablation Studies

To further investigate the effectiveness of the image pyramid within CAN and the PST, we conduct additional ablation studies on the other two datasets. This also serves to justify our choice of different hyperparameter settings across datasets.

Table 5 reports ablation studies on the image pyramid within CAN. We compare different scale settings on three datasets. The results show that scales $\{1, 3, 6\}$ work best for buildings, while $\{1, 5, 10\}$ perform best for water bodies and roads. This indicates that smaller buildings need less context, and too much context may harm performance. In

Table 3. Computational cost analysis on WHU-Building

Method	Training Time (h)	Params (M)	GPU Memory (G)	Infer Time (s/sample)
HiSup [8]	10.86	74.29	0.48	2568.89
Ours CAN	16.27	268.52	1.88	125.31
Ours MCR	-	-	-	132.78
Ours PST	15.66	43.24	1.25	200.01
Ours (ALL)	31.93	311.76	1.88	458.10

Table 4. The ablation studies on the PST.

Dataset	Vectorize method	Vector metrics		Instance metrics				Semantic metrics	
		$PoLiS \downarrow$	$CIoU$	AP	AP_s	AP_m	AP_l	IoU	$F1$
WHU-Building	Baseline[1]	3.83	18.47	58.75	37.87	79.85	49.13	91.55	**95.54**
	Baseline[1] + DP [2]	4.02	60.32	58.42	37.50	79.76	49.13	91.50	95.52
	Baseline[1] + PST	**3.63**	**82.30**	**61.07**	**40.37**	**80.30**	**60.00**	**91.60**	95.41
GLHWater	Baseline[1]	82.85	26.74	20.30	10.48	35.44	58.25	**85.76**	**91.55**
	Baseline[1] + DP [2]	83.72	55.08	20.31	**10.53**	35.34	58.25	85.74	91.54
	Baseline[1] + PST	**82.42**	**57.88**	**20.84**	10.08	**38.08**	**70.35**	85.50	91.40
VHRRoad	Baseline[1]	135.05	1.43	**1.71**	0.08	**0.43**	3.74	**46.80**	**61.03**
	Baseline[1] + DP [2]	138.01	5.41	1.70	0.08	0.42	3.74	46.65	60.88
	Baseline[1] + PST	**134.13**	**6.10**	1.58	**0.08**	0.40	**3.99**	46.48	60.63

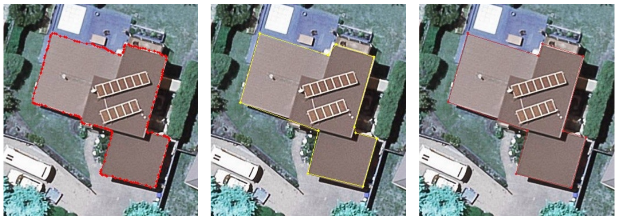
[1] Using TC89-KCOS [5] to extract polygon contours.

Table 5. The ablation studies on the image pyramid within CAN.

Dataset	Context	IoU	F1
WHU-building	1	91.23	95.29
	1, 3, 6	**92.21**	**95.94**
	1, 4, 8	92.12	95.68
	1, 5, 10	92.14	95.72
GLH-water	1	85.73	91.14
	1, 3, 6	86.45	92.59
	1, 4, 8	86.90	93.00
	1, 5, 10	**87.95**	**93.59**
VHR-road	1	48.35	65.09
	1, 3, 6	49.74	66.32
	1, 4, 8	49.69	66.40
	1, 5, 10	**50.47**	**67.09**

contrast, larger and more connected water bodies and roads benefit from more contextual information. Note that segmentation is evaluated at the pixel level, so semantic metrics differ from vectorized semantic metrics.

Table 4 presents ablation studies on the PST. Comparing the baseline and PST methods across three datasets, PST notably improves vector and instance metrics. Although semantic metrics slightly decrease, PST achieves better instance extraction and vector representation.



Baseline

Baseline + DP

Baseline + PST

Figure 1. The visualization of output polygons of different methods.

6. Supplementary Visualization Results

We provide additional visualization results on the WHU-building, GLH-water, and VHR-road datasets in Fig. 1, Fig. 2 and Fig. 3. Fig. 1 illustrates the output polygons of different methods, including the baseline method using TC89-KCOS for polygon contour extraction, baseline with DP simplification, and baseline with our proposed PST. Fig. 2 displays the vectorization results of all baseline methods and our HoliTracer on the WHU-building, GLH-water, and VHR-road datasets. Fig. 3 presents the vectorization results of HoliTracer on large-size RSI. The results demonstrate that HoliTracer produces more accurate and complete vector representations compared to existing patch-based methods, and it effectively handles diverse geographic objects across large-size RSI.

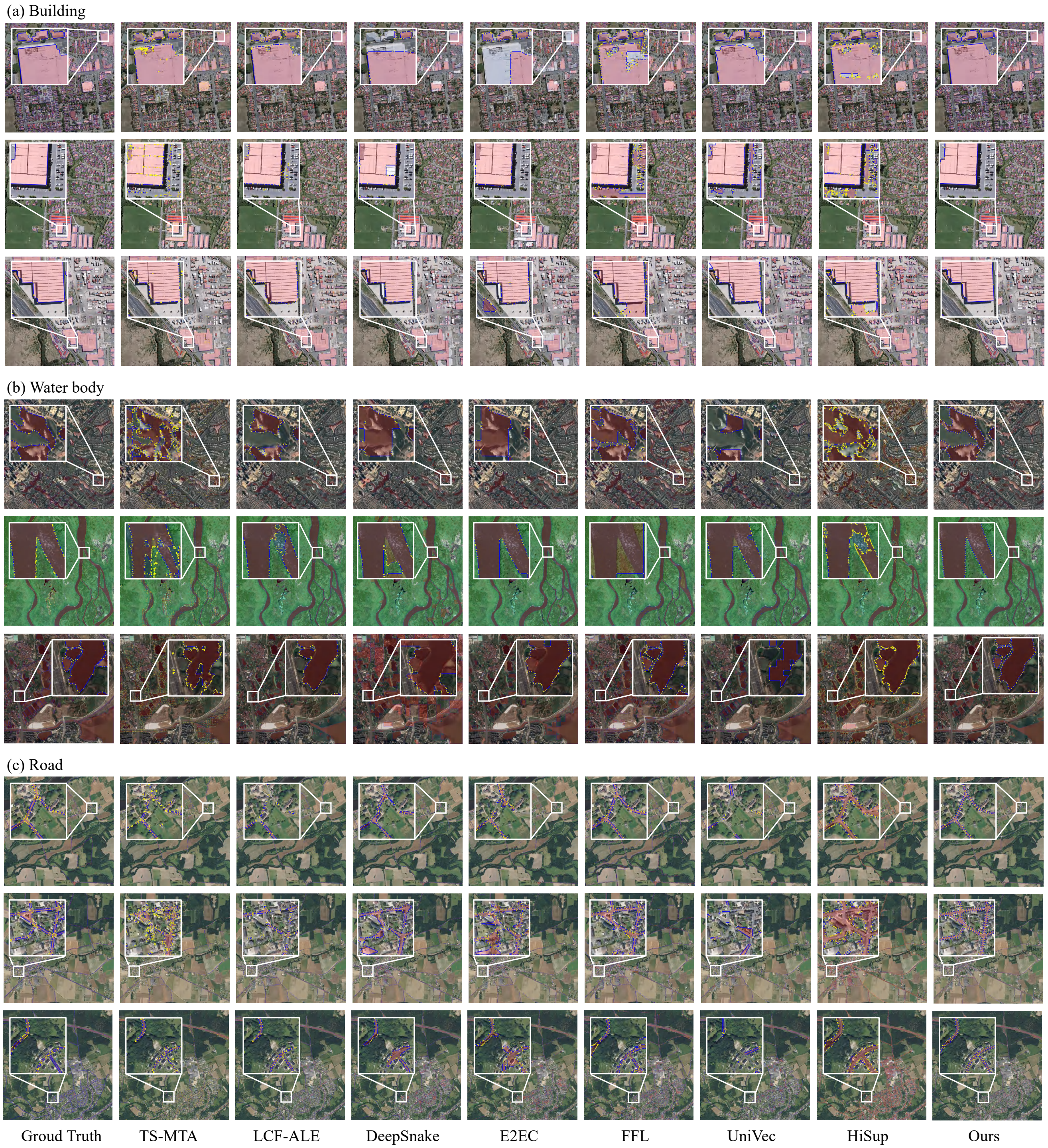


Figure 2. Visualization of vectorization results of the all methods on WHU-building, GLH-water, and VHR-road test datasets. Our method produces more accurate and complete vector representations compared to existing patch-based methods.

References

[1] 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. 1

[2] David H Douglas and Thomas K Peucker. Algorithms for the reduction of the number of points required to represent a dig-

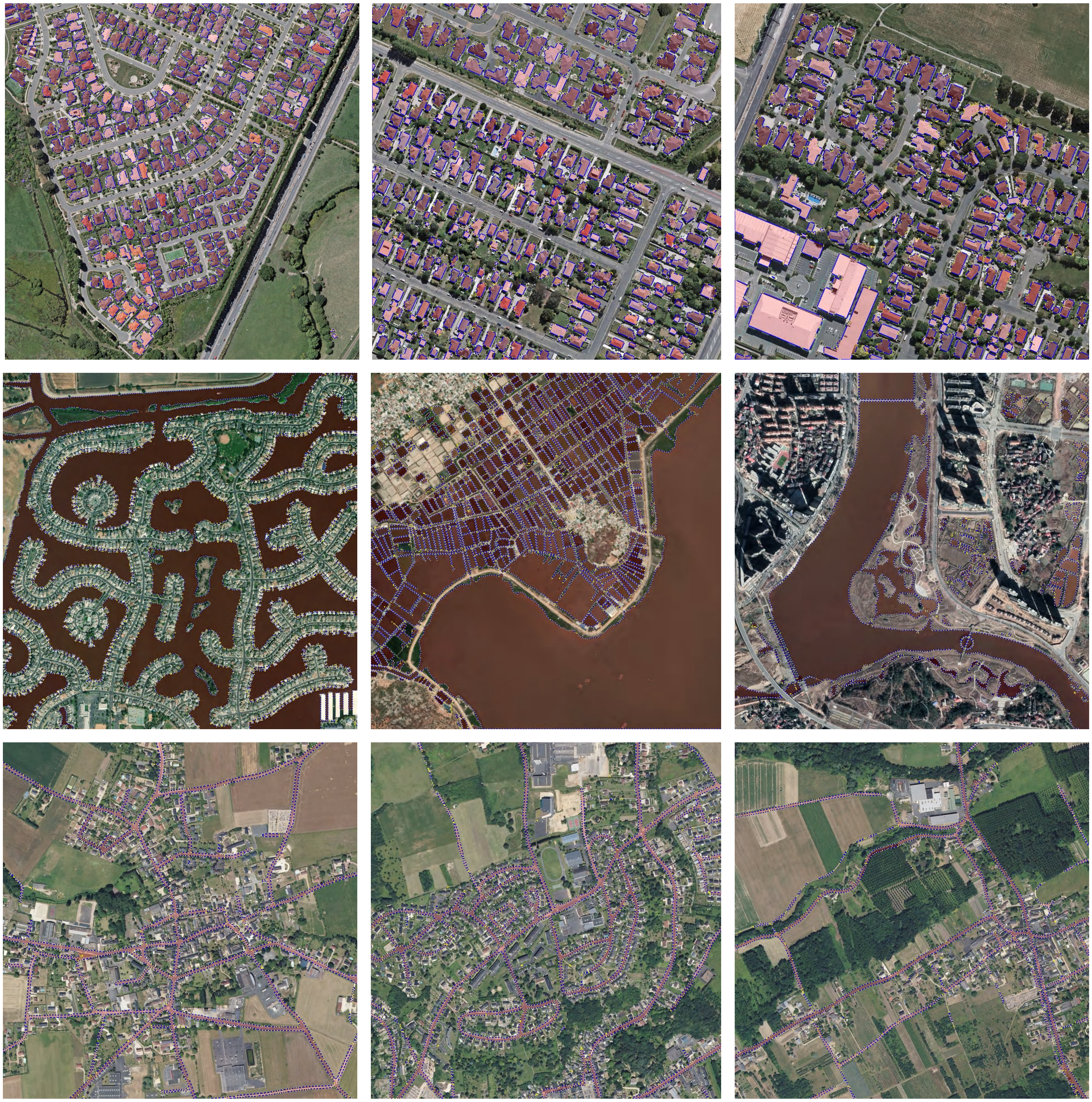


Figure 3. Visualization of HoliTracer's vectorization results on large-size RSI.

itized line or its caricature. *Cartographica: the international journal for geographic information and geovisualization*, 10 (2):112–122, 1973. 3

[3] European Environment Agency (EEA). Urban Atlas Land Cover/Land Use 2018 (vector), Europe, 6-yearly. European Union's Copernicus Land Monitoring Service information, 2021. 2

[4] Institut national de l'information géographique et forestière (IGN). BD ORTHO®, 2025. Dernière consultation le 21 juillet 2025. 2

[5] C-H Teh and Roland T. Chin. On the detection of dominant points on digital curves. *IEEE Transactions on pattern analysis and machine intelligence*, 11(8):859–872, 1989. 3

[6] Adam Van Etten, Dave Lindenbaum, and Todd M Bacastow. Spacenet: A remote sensing dataset and challenge series. *arXiv preprint arXiv:1807.01232*, 2018. 1

[7] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui

Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 43(10):3349–3364, 2020. 2

[8] 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. 1, 3

[9] 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. 1