Road Network Vectorization With Geometric Enforcement

Zhenyu Zhu, Elena Di Bernardino, Florent Lafarge

Road Network Vectorization With Geometric Enforcement

Zhenyu Zhu[1,2], Elena Di Bernardino[2], Florent Lafarge[1]

[1]Centre Inria d'Université Côte d'Azur – Sophia-Antipolis, France – firstname.lastname@inria.fr
[2] Université Côte d'Azur, Laboratoire J.A. Dieudonné – Nice, France – firstname.lastname@univ-cotedazur.fr

Keywords: Road network vectorization, graph-based representation, line-segment detection, polygonal partitioning, geometric guarantees

Abstract

We present an automatic algorithm for graph-based road network extraction from remote sensing images. While existing works mostly focus on improving accuracy, we address the problem of the geometric quality of the output graphs. The state-of-the-art methods largely overlook this aspect by generating graphs without strong geometric guarantees, regularity preservation and low-complexity, which, ultimately, reduces their impact in many application scenarios. Our algorithm relies upon foundation models that analyze road networks with pixel-based representations, as well as geometric algorithms and data structures in charge of connecting geometric primitives into planar graphs. This hybrid strategy allows us to strongly enforce the geometric quality of the output graphs while bringing a high level of generalization. We show the potential of our algorithm and its advantages over existing methods on two datasets commonly-used in the field using both the conventional accuracy metrics and new metrics introduced to measure the geometric quality of the output graphs.

1. Introduction

Automatic extraction of road networks from remote sensing images is a long-standing research problem with numerous potential applications, e.g., in traffic control, navigation and autonomous driving, road network planning and smart cities. For the large majority of these applications, road networks must be represented with a vector format, typically under the form of graphs where nodes corresponds to junctions, turns and dead-ends, and edges, to portions of road between two nodes.

Recent deep learning models proposed in the field directly produce such graph-based representations. They usually rely upon end-to-end architectures with modules in charge of predicting the position of the nodes and connecting them to form a graph. These methods have made significant progress to improve the accuracy of the detected segments of roads (Lu and Weng, 2025, Chen et al., 2022), but have largely overlooked the geometric quality of the produced graphs. Among the main geometric problems commonly found in their results, one can list

(a) the geometric defects contained in the graphs, e.g., the non-guarantee of the graph planarity with typically two crossing edges that do not intersect at a node position, or isolated nodes or edges disconnected from the road network;

(b) the complexity of the graphs where nodes are typically sampled at a too high frequency along straight lines and where road networks are often disconnected;

(c) the non-preservation of geometric regularities with, e.g., the parallelism, orthogonality or collinearity of road segments that are not respected in the graphs.

Unfortunately, the lack of geometric guarantees on the outputs produced by these methods strongly reduced their applicability on concrete tasks. Some methods proposed to inject geometric priors (typically via loss functions) to attenuate this issue, but

Figure 1. Our algorithm extracts road networks as low-complexity graphs while preserving geometric regularities (e.g., parallelism and orthogonality) and guaranteeing key applicability properties such as the graph planarity (right). In contrast, a state-of-the-art method such as SAM-Road (Hetang et al., 2024) produces an overly complex and irregular graph, prone to geometric defects (left). Simplifying SAM-road results with the popular Douglas-Peucker method reduces the complexity of the graph, but increases defects and irregularity (center).

such priors are not sufficient to guarantee the geometric validity of the graphs due to the limited nature of (i) architectures unable to handle topological operations on graphs in an efficient and differentiable manner, and (ii) existing training sets composed of geometrically-inconsistent Ground Truth graphs. Correcting these defects by post-processing (typically with heuristic-based geometry processing) is also a delicate task which is likely to degrade the accuracy of the graph without resolving all the defects and returning guarantees.

To address this issue, we propose a method based on the polygonal decomposition of the input images. In contrast to end-to-end-strategies, we approach the problem with a hybrid model that relies upon both foundation models that analyze road networks with pixel-based representations and geometric algorithms and data structures in charge of connecting geometric primitives into planar graphs. Besides ensuring geometric guarantees, this hybrid strategy also offers flexibility and high level

of generalization as the learning part relies upon generic foundation models likely to progress in the future only, without trying to learn how to assemble geometric primitives from defect-laden Ground Truth graphs.

Our method operates in a dual way of the traditional end-to-end strategies such as (He et al., 2020, Xu et al., 2022, Hetang et al., 2024): instead of predicting nodes and connecting them, we predict line-segments and extend them to form a polygonal partition of the image domain. This prevents us from having to analyse the topologically-complex adjacency between predicted nodes. Instead, nodes are created by extending line-segments and the output graphs are extracted as a subset of edges of the polygonal partition (which is guaranteed to be a valid embedding, i.e. with no gap or overlap between polygonal regions). This solution does not only guarantee the planarity property of the graph (problem (a)), it allows us to address problem (b) when the extraction of edges from the partition is formulated as an optimization problem under constraints, and problem (c) when predicted line-segments are detected under geometric regularities, i.e. parallelism, orthogonality and co-linearity.

We show the potential of our algorithm and its competitiveness against existing methods on two datasets commonly-used in the field on both the conventional accuracy metrics and new metrics introduced to measure the geometric quality of the output graphs. In particular, we demonstrate our algorithm produces output graphs with a higher geometric quality while approaching the accuracy scores of the best existing methods.

2. Related work

Our review of related work covers the recent end-to-end architectures for graph-based road network extraction as well as learning models exploiting geometric priors and vectorization techniques that simplify chains of pixels to polylines.

End-to-end architectures. End-to-end models that directly output graph-based results are usually composed of several modules for predicting nodes and for connecting them. The key differences between these models often originate from the connection module. The first efficient models in this direction (Chu et al., 2019, Li et al., 2019) have exploited Recurrent Neural Networks with an iterative construction of the connectivity. Graph Neural Networks (GNNs) have also been considered to select possible edges from an exhaustive connectivity. Sat2Graph (He et al., 2020) uses a Graph-Tensor Encoding whereas SERGE (Bahl et al., 2022) relies upon a k-Nearest Neighbor search to reduce the number of possible connections. More recently, transformers have been deeply used for the node prediction, e.g. with RNGDet++ (Xu et al., 2022), and for the analysis of the node connectivity, e.g. with SAM-road (Hetang et al., 2024) that exploits a lightweight transformer-based GNN in combination with SAM-based embeddings (Kirillov et al., 2023). The latter has been improved in (Yin et al., 2025) with a larger training and several technical ingredients as a node-guided resampling and a connectivity classifier. While they offer a good accuracy on the position of nodes and edges, these methods produce graphs with a low geometric quality.

Models with geometric priors. Several models have been proposed to increase the geometric quality of output graphs. Sotiris et al. (Sotiris et al., 2023) address the problem of overly disconnected road edges with a graph completion model via reinforcement learning. GraphMorph (Zhang et al., 2024) relies upon a so-called Morph module to generate a centerline mask aligned with the predicted graph, the idea being to better analyse the node connectivity with centerlines and produce graphs with a lower complexity and a better topology. GraphMapper (Wang et al., 2023b) presents an end-to-end model where predicted nodes are regularized according to parallel and orthogonal alignments of nodes. While interesting, these methods only partially address the geometric quality problem of graphs. In contrast, we propose a more complete solution to the problems (a), (b) and (c) listed in Section 1.

Vectorization techniques. Methods have also been proposed to simplify graphs or to vectorize chains of pixels into graphs. The popular Douglas-Peucker algorithm (Douglas and Peucker, 1973) and its variants (Agarwal et al., 2002, Song and Miao, 2016) constitute the de facto solution. While simple and natural, this iterative node reduction mechanism does not fix topological issues and is likely to create crossing edges. Geometry processing techniques operating on Delaunay triangulations bring more geometric guarantees, but they are unable to preserve regularities and require costly optimization procedures based on optimal transportation (de Goes et al., 2011) or Monte Carlo sampling (Favreau et al., 2020). Note that Vector Graphics techniques (Tian and Günther, 2024) are closely related to this simplification problem as pixel-based object contours often need to be approximated by polylines. Recently, efficient differentiable rasterizers such as DiffVG (Li et al., 2020) have been proposed, without offering specific geometric guarantees.

3. Algorithm

Overview. The algorithm takes an aerial or satellite image as input, and outputs a planar graph whose edges align with road portions in the input image. The graph is guaranteed to be planar by construction, i.e. without crossing edges.

Figure 2 illustrates the different steps of our algorithm. It starts by computing a road probability map of the road network at the pixel level, typically by using a recent foundation model proposed in the field. We then extract the centerlines of high probabilities by skeletonization, and fit line-segments to it. The detected line-segments are then extended to form a polygonal partition of the image domain. The final step then consists in extracting a planar graph from the edges of the partition using a global optimization under geometric constraints. We present next these different steps in detail.

Road probability map. The road probability map is computed through a state-of-the-art image encoder in the field. We use in our experiments the same SAM-based embedding as the one proposed in SAM-Road (Hetang et al., 2024). The latter directly provides a probability map corresponding to the existence of roads. Note that it also provides the probability map of the existence of intersection points, but we do not use this prediction which is not required for our line-segment extension strategy.

Centerline extraction. Directly fitting line-segments to the road probability map leads to imprecise results due to the heterogeneous thickness of road parts and the noise located near complex road junctions. Inspired by GraphMorph (Zhang et al., 2024), we address this imprecision issue by extracting a

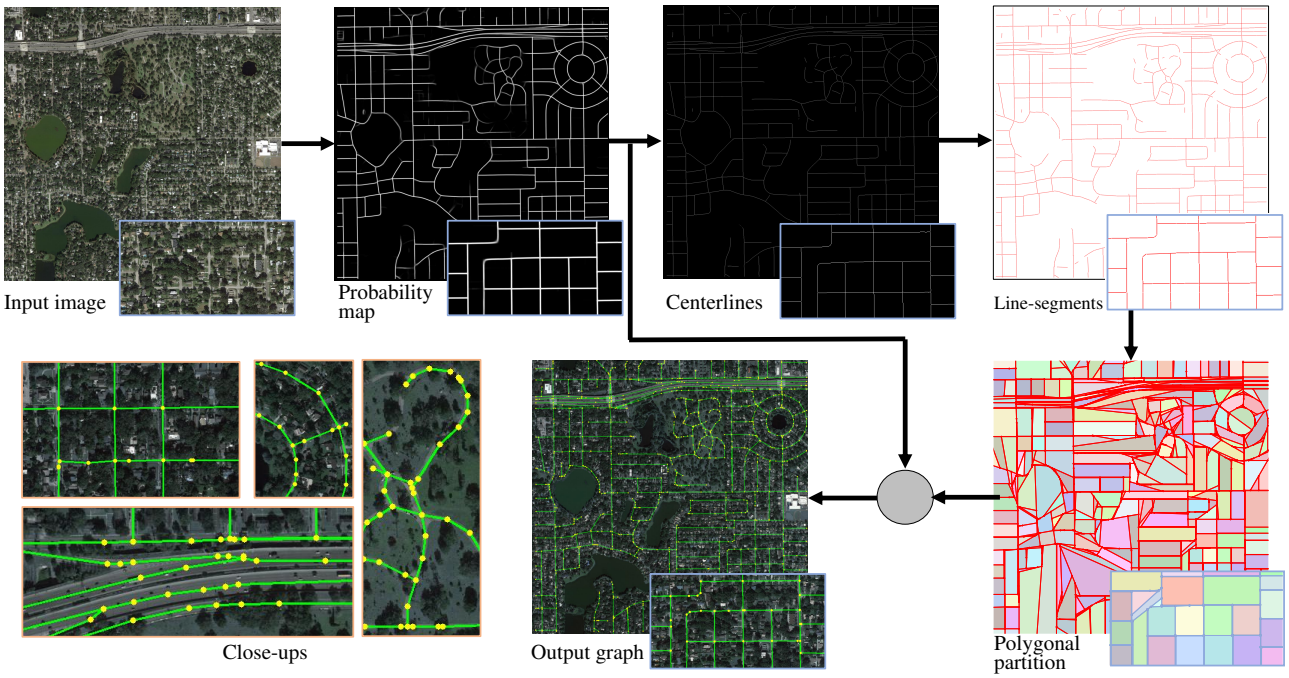


Figure 2. Overview of our vectorization algorithm. Starting from an input image, we compute a road probability map using a foundation model, extract its centerlines and fit line-segments to it. We then construct a polygonal partition by extending the detected line-segments and select the edges to be part of the output graph by global optimization (grey disk). Close-ups show results in various local configurations, i.e., curved roads, a multi-road highway and grid-based road layouts.

one-pixel-thickness skeleton from the probability map, as illustrated in Figure 3. This representation by centerlines allows us to solve topological ambiguities at complex junctions and to more accurately fit line-segments. In practice, we use the skeletonization method proposed in (Menten et al., 2023) from a binarized version of the probability map by Otsu method (Otsu, 1979).

Figure 3. Centerline extraction. The centerlines (right, thin red lines on white pixels) are extracted as the one-pixel-thickness skeleton from the probability map. Note that centerlines are often wavy at some locations: this pixel-level noise is typically absorbed by the line-segment detector.

Line-segment detection. Popular line-segment detectors such as DeepLSD (Pautrat et al., 2023) or ScaleLSD (Ke et al., 2025) are mostly designed to fit line-segments on line-like structures and are not adapted to our strategy where line-segments must also capture curved roads. We then use the more generic LineFit detector (Boyer et al., 2024) which offers two interesting properties. First, this algorithm has the particularity to approach the detection problem with a point-to-line fitting formulation capable to also approximate curves with sequences of line-segments. Second, the algorithm detects line-segments while preserving three key geometric regularities in our context, i.e., parallelism, orthogonality and co-linearity. This allow the alignment of the road portions to better recover grid-based road networks. In our experiments, we considered the centerline map as the input image gradient of this method and set the fitting tolerance parameter to two pixels and the minimal line-segment size to ten pixels.

Polygonal decomposition. The line-segments are then extended at constant speed within a kinetic simulation framework until they collide and form a polygonal decomposition of the image domain. We use the method proposed in (Bauchet and Lafarge, 2018) to produce a geometrically-valid polygonal decomposition, i.e., with guarantee that there is no overlap or gap in between the polygonal regions. In our experiments, only one collision per line-segment are allowed, leading to the construction of the polygonal decomposition as a motorcycle graph. Nodes corresponding to road junctions and turns are thus not predicted as in existing methods, but computed by intersecting line-segments. This strategy by extension of line-segments is particularly interesting in presence of highly occluded roads and atypical junctions unseen in training sets.

This polygonal partition however suffers from two drawbacks: (i) three or more line-segments that should meet at a junction are unlikely to intersect exactly at the same point, and (ii) dead-ends are unlikely to be captured by nodes with this line extension strategy.

To solve problem (i), we operate a recursive merging of the pairs of adjacent vertices if their distance is lower than 5 pixels once the polygonal partition is built. After each merge, we update the edge connectivity in the star-domain of the collapsed edge so that the crossing of edges does not occur.

Problem (ii) is addressed by adding extra-nodes on the edges of the partition where potential end points exist. This operation thus does not modify the topology of the polygonal partition, but simply splits some existing edges. We operate by first detecting the end points in the centerline map with a basic connectivity analysis, and then checking whether these end points are located along an edge of the partition, excluding their

extremities. If this test is valid (i.e. under a two pixel tolerance distance between the end point and the edge in our experiments), we project the end point to the edge and split it according to this projected point. Note this test is often valid in practice as line-segments are usually correctly detected in dead-end situations.

Edge selection. The last step of our vectorization algorithm consists in selecting the subset of edges forming the output graph from the polygonal partition. By construction of the polygonal partition, this subset is guaranteed not to contain crossing edges. Another interesting property of the polygonal partition is its conciseness: the number of edges is low compared to the commonly-used exhaustive strategy where each pair of nodes gives a possible edge to select. In our case, the edges are created in a simple and natural manner by extending line-segments without arbitrary heuristics and distance-based parameters.

We formulate the edge selection as a binary linear programming (BLP) problem. We define by $x_i = \{0, 1\}$, the dummy variable that specifies whether edge i of the polygonal partition is selected ($x_i = 1$) or non-selected ($x_i = 0$) to be part of the output road graph. We denote by $x = (x_i)_{i \in E}$, a possible output graph where E corresponds to the set of edges in the polygonal partition. We measure the quality of a configuration by a two-term objective function subject to two types of geometric constraints. Our BLP problem is expressed by

$$\begin{aligned} &\max_{x} \left[f(x) + \beta g(x) \right] \\ &\text{s.t. } x_i + x_j \leq 1 \qquad \forall (i,j) \in \mathcal{A}, \quad (c1) \\ &\sum_{k \in \mathcal{N}(i)} x_k \geq x_i \qquad \forall i \in E, \quad (c2) \end{aligned} \tag{1}$$

where $f(x)$, $g(x)$ are the two terms of the objective function and β the parameter balancing them respectively, $\mathcal{A}$ corresponds to the set of pairs of adjacent edges in the polygonal partition whose relative angle is lower than five degrees and $\mathcal{N}(i)$ is the set of adjacent edges to edge i. It corresponds to all the edges connected to the two extremities of edge i. Then, constraint (c1) in Equation (1) avoids very small angle configurations between adjacent edges and constraint (c2) avoids isolated edges.

The term $f(x)$ of the objective function aims to encourage the selection of edges that align with high probability pixels in the road probability map. We formulate this term with a linear expression of the form

$$f(x) = \sum_{i \in E} \frac{l_i}{l} \left[w_i x_i + (1 - w_i)(1 - x_i) \right], \tag{2}$$

where l_i and l correspond to the length of edge i and the average length of all the edges of the polygonal partition respectively. The weight w_i is a real value between 0 and 1 that scores how well edge i aligns with the road probability map. In our experiments, w_i is computed as the ratio of high probability pixels that overlap with the pixelized edge i.

The term $g(x)$ acts as a (soft) geometric prior by encouraging the connectivity between selected edges. In practice, we penalize the non-selection of edges contained in the minimum spanning tree of the polygonal partition by

$$g(x) = \sum_{i \in E} (1 - x_i) 1_{\{i \in MST\}}, \tag{3}$$

where $1_{\{.\}}$ is the Heaviside function and MST is the set of edges of the minimum spanning tree. While conceptually simple, the idea of using a minimum spanning tree has been proven to enforce connectivity between edges quite efficiently in previous works (Wang et al., 2023a, Ameri and Valadan Zoej, 2015).

To solve this binary linear optimization problem with linear constraints, we use the CP-SAT solver of the OR-Tools library (Perron and Didier, n.d.). This solver offers good performance and good scalability in practice.

4. Experiments

We now expose our experimental setup, present comparative results, ablations and discuss limitations of our algorithm and its applicability to other vectorization problems. The Experiments were run on Intel Core Ultra 7 165H CPU with 64 GB RAM and a RTX 2000 Ada GPU.

Datasets and evaluation metrics. We evaluated our algorithm on two datasets commonly-used in the field, SpaceNet (Etten et al., 2018) and City-Scale (He et al., 2020), which exhibit a good diversity of urban landscapes. The former is composed of 2549 images of 400 × 400 pixels, including 382 for testing, whereas the latter has 180 images of 2048 × 2048 pixels with 29 images for testing. For the two datasets, the ground sampling distance is one meter, and Ground Truth is provided under the form of dense graphs.

To measure accuracy, we used the four standard metrics: the TOPO-Precision (P), TOPO-Recall (R), TOPO-F1 (F1) introduced in (Biagioni and Eriksson, 2012) and the Average Path Length Similarity (APLS) formalized in (Etten et al., 2018).

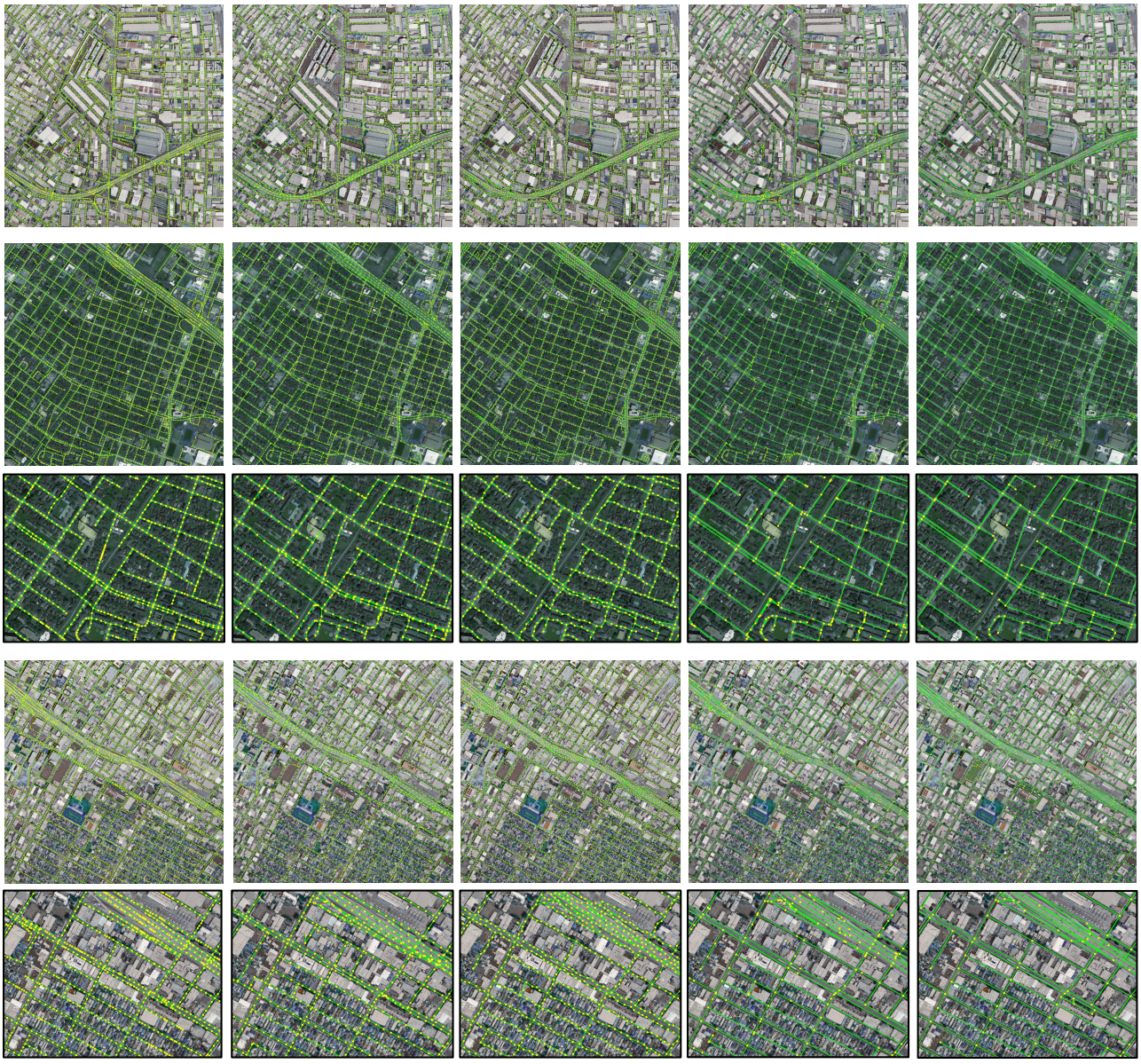
We also proposed five metrics to measure the geometric quality of output graphs. The Crossing Edge Ratio (CER) returns the percentage of crossing edges in the output graph that do not intersect at a node position. This score is zero in case of planar graphs. Similarly, the Isolated Edges Ratio (IER) gives the percentage of disconnected edges in the output graph. We also introduced the Sharp Turn Ratio (STR) that measures the percentage of very small angle (i.e., lower than five degrees) between two adjacent edges of the output graph. To measure the level of regularity of the output graph in terms of parallelism, orthogonality and co-linearity, we use the Degree of Freedom (DoF) score introduced in (Boyer et al., 2024). Finally, we measure the complexity of the output graph with its Average Number of Nodes (ANN).

Baselines. We compared our algorithm to three graph-based methods: Sat2Graph (He et al., 2020), RNGDet++ (Xu et al., 2022) and SAM-road (Hetang et al., 2024). For each of them, we used the models pre-trained on the SpaceNet and City-Scale datasets, provided by the authors. The same training splits were used for these methods and our model. Because these three methods produce road graphs significantly more complex than ours, we also proposed a variant for each method by simplifying their output graphs so that the average number of nodes becomes similar to ours. This post-processing simplification consists in collapsing edges by the popular Douglas-Peucker algorithm (Douglas and Peucker, 1973).

Results. Table 1 presents quantitative comparisons in terms of accuracy, geometric quality and performance while Figures 4 and 5 show visual results on images of the two datasets.

		Accuracy				*Geometric quality*					
		F1 (↑)	P(↑)	R (↑)	APLS (↑)	CER (↓)	DoF (↓)	STR (↓)	IER (↓)	ANN	Time
SpaceNet	Sat2Graph	80.97	85.93	**76.55**	64.43	0.23	60.44	0.00	0.00	105	72min
	RNGDet++	**82.51**	91.34	75.24	67.73	0.81	67.17	0.13	1.66	78	142min
	SAM-Road	80.52	**93.03**	70.97	**71.64**	0.03	73.71	0.00	1.21	107	39min
	Sat2Graph + DP	80.10	85.91	75.03	62.03	0.47	82.14	0.21	6.45	36	72min
	RNGDet++ + DP	79.22	90.02	70.74	67.50	1.22	57.99	0.23	3.23	58	142min
	SAM-Road + DP	78.36	92.87	67.78	69.58	0.04	79.73	0.00	2.03	84	39min
	Our method	75.98	90.01	65.73	62.12	**0.00**	**49.62**	**0.00**	**0.00**	72	44min
CityScale	Sat2Graph	**80.70**	72.28	**76.26**	63.14	0.58	41.35	0.00	0.00	4477	148min
	RNGDet++	78.91	87.10	72.29	67.37	3.26	44.00	0.29	0.05	3109	264min
	SAM-Road	77.23	**90.47**	67.69	**68.37**	0.48	48.75	0.02	0.25	2847	42min
	Sat2Graph + DP	76.97	81.10	73.58	63.64	1.71	51.79	0.03	0.01	1337	148min
	RNGDet++ + DP	78.81	87.13	72.11	67.15	6.58	52.62	0.62	0.23	1288	264min
	SAM-Road + DP	77.03	90.37	67.45	66.86	0.57	54.26	0.02	0.51	1333	42min
	Our method	72.26	85.31	63.14	62.35	**0.00**	**38.81**	**0.00**	**0.00**	1183	36min

Table 1. Quantitative comparisons on *SpaceNet* and *CityScale*. Bold and underlined values indicate the best and second best scores respectively. "+ DP" means the output of the baseline has been simplified by a Douglas-Peucker algorithm.



Sat2Graph RNGDet++ SAM-road Our method Ground Truth

Figure 4. Qualitative results on CityScale. In contrast to the baselines, our algorithm delivers compact graphs where a straight road section between two junctions is likely to be represented by a unique edge, and where geometric regularities are preserved. In the closeups, note how pairs of parallel, close roads are made exactly parallel, whereas baselines are prone to geometric defects with crossing edges and road fusions.

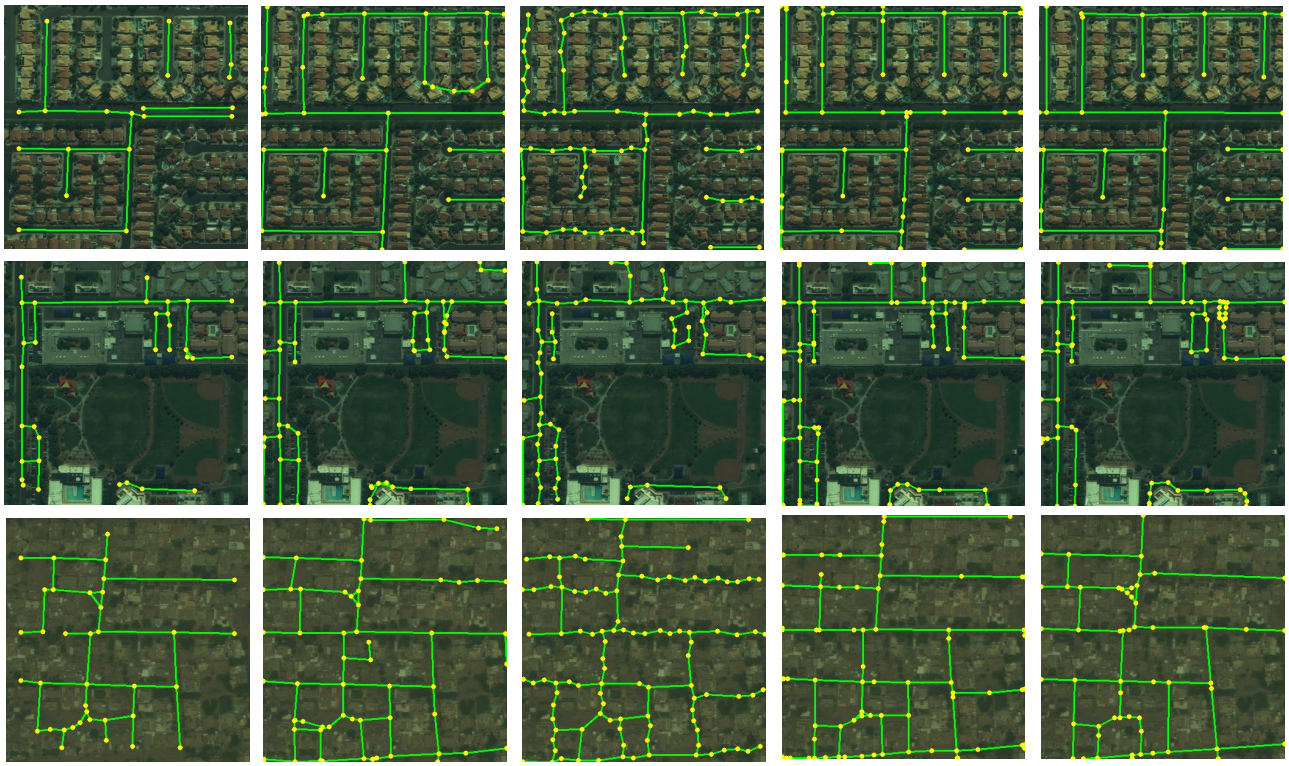


Figure 5. Qualitative results on SpaceNet. Simplifying the results of Sat2Graph, RNGDet++, and, to a lesser extent, SAM-road by a Douglas-Peucker (DP) algorithm produces low-complexity graphs but exhibits geometric and topological inaccuracies. Our algorithm directly returns concise graphs that are similar to the low-complexity ground truth.

Unsurprisingly, the best accuracy scores are obtained by end-to-end models in their original version. SAM-road exhibits a slightly better TOPO-precision and APLS scores while Sat2Graph has a higher TOPO-recall and TOPO-F1. The simplification of their results by Douglas-Peucker typically decreases the scores by a few points. The accuracy scores of our algorithm are slightly lower, typically between two to five points below those of SAM-Road. While our accuracy scores remains good, this drop can be explained by two factors. First, our results have a lower level of expressiveness as graphs that do not fulfil the geometric constraints cannot be retained as output solution. Second, TOPO and APLS scores, which are the de facto metrics for the end-to-end models in the field, give a clear advantage to methods that position nodes very accurately with respect to the Ground Truth. Our "dual" strategy in which nodes are extracted by extending line-segments is thus naturally penalized by these scores. This is particularly true in presence of road networks with complex junctions connecting more than three roads.

Our algorithm performs best on the geometric quality metrics as a result of the geometric guarantees that it offers. In particular, the number of crossing edges, very sharp turns and isolated edges are guaranteed to be zero. In contrast, other methods do not offer such guarantees, even if Sat2Graph exhibits good scores on the SpaceNet dataset (where road networks are smaller and simpler than for the CityScale dataset). Their scores are even degraded when their results are simplified to reach a similar graph complexity (ANN) than our results. Our algorithm also scores significantly better than the baselines on the degree of freedom (DoF) metric, showing the high level of regularity in the layout of edges. This is particularly true for urban scenes with grid-based road networks.

Regarding performance, our processing times are similar to the inference timing of SAM-road, the faster end-to-end baselines. This is very promising given the fact (i) our current implementation is purely sequential and does not exploit GPU computing, and (ii) our training effort is limited to a road probability map prediction. Our method also scales very well and does not need tiling strategies to process large scenes.

As shown in Figure 6, our algorithm offers good robustness to imperfect probability maps that contain holes along the roads. Edges are typically created on such holes by our algorithm in a natural manner, by simply extending the close line-segments during the polygonal partition construction.

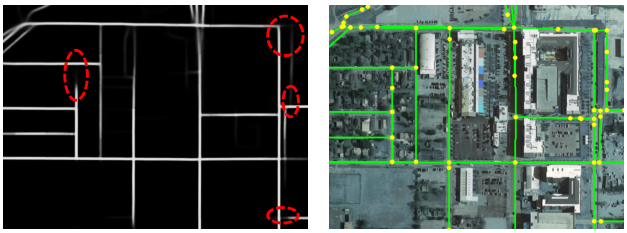


Figure 6. Robustness to imperfect probability maps. Our algorithm offers good robustness to holes contained in probability maps, partly due to occlusions (see red ellipses in the probability map on the left).

Ablation. We conduct ablation experiments to evaluate the effects of each design choice on the Cityscale dataset. The results are shown in Table 2.

(1) Fitting line-segments directly to the road probability map reduces the processing time by half, the skeletonization for extracting the centerlines being the most costly operation of our

	Accuracy				*Geometric quality*					
	F1 (↑)	P(↑)	R (↑)	APLS (↑)	CER (↓)	DoF (↓)	STR (↓)	IER (↓)	ANN	Time
Vanilla	72.26	85.31	63.14	62.35	0.00	38.81	0.00	0.00	1183	36min
(1) No centerline extraction	67.30	89.23	54.82	61.07	0.00	40.36	0.00	0.00	772	17min
(2) No line-segment regularization	70.01	86.73	59.27	58.53	0.00	41.81	0.00	0.00	1098	36min
(3) No node merging	71.30	83.50	62.89	57.63	0.00	25.24	0.00	0.00	1744	35min
(4) No end-point adding	69.61	81.70	61.27	58.77	0.00	26.58	0.00	0.00	1502	36min
(5) Soft constraint optimization	74.30	84.70	66.64	64.27	0.00	40.6	0.23	0.18	1206	36min
(6) $\beta = 0$	71.66	86.25	61.78	60.25	0.00	41.46	0.00	0.00	1186	36min
(7) No length weight	71.07	86.62	60.77	60.77	0.00	72.00	0.00	0.00	1146	36min
(8) No optimization	74.51	81.85	68.85	64.15	0.00	38.63	0.26	0.79	1274	35min

Table 2. Ablation study on the CityScale dataset.

algorithm. This solution however degrades the accuracy scores as line-segments are less accurately fitted, especially when the variation of road widths is high in the urban scenes.

(2) The deactivation of the line-segment regularization during the fitting decreases the DoF score while having a marginal impact on accuracy scores.

(3) Not merging the nearby nodes of the polygonal partition has also a minor impact on the accuracy metrics, but gives more complex output graphs.

(4) The deactivation of the extra-node adding makes the TOPO metrics and APLS decrease by a few points, mostly due to the incapacity of our algorithm to capture dead-ends. The topology of the output graphs is also simplified, leading to an improvement of the DoF score.

(5) Relaxing the hard constraints (c1) and (c2) from Equation (1) as soft constraints in the objective function allows us to gain a few points on the accuracy metrics, but does not guarantee the STR and IER score to be at zero anymore.

(6) Removing the minimum spanning tree term from the objective function increases the TOPO-precision but degrades all the other accuracy metrics without having a clear impact on the geometric quality metrics.

(7) Not weighting the unary components of objective term $f(\cdot)$ by the edge length has not a big impact on the accuracy scores, but tends to strongly degrade the DoF score (the output graphs are more likely to contain small, irregular edges).

(8) By selecting edges as just the $\arg\max$ configuration of $f(\cdot)$, we reach high accuracy scores, in particular with the best TOPO-F1, TOPO-recall and APLS scores of our ablations. This scenario however degrades the geometric quality scores. Note that, by construction of our polygonal partition, the output graphs in all the scenarios are guaranteed to be planar with a CER score at zero.

Applicability to other vectorization tasks. Our algorithm offers a good genericity potential as it departs from a simple probability map. In particular, one can vectorize other types of linear structures as long as a probability map can be accurately predicted. Figure 7 shows two use cases for the vectorization of blood vessels in retina images (Staal et al., 2004) and of river networks in aerial images (Li et al., 2024). Note that we deactivated the regularization of line-segments during detection as parallelism and orthogonality are improbable relations for these organic objects.

Limitations. Our algorithm has a few shortcomings. While generic and extensible to various vectorization tasks, our framework can only handle a certain range of geometric guarantees, typically interactions that can be modeled with linear or quadratic constraints if using the current solver. Our algorithm also targets low-complexity graphs, and is not adapted to produce denser graphs, as those returned by the existing methods in the field. The tolerance fitting parameter of the line-segment detector allows us to impact the complexity of the graph, but without a direct control of the number of nodes or edges in the output graphs.

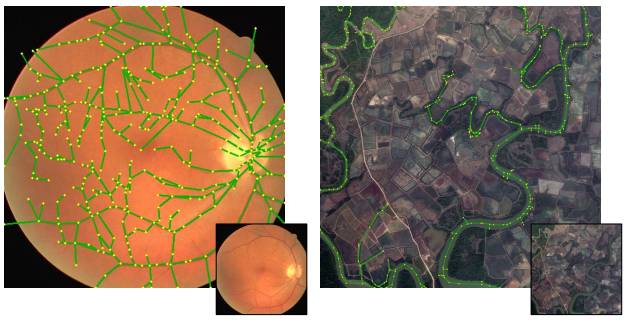


Figure 7. Applicability to other problems. Our algorithm can be used for other tasks, including the vectorization of organic objects such as blood vessels in retina images (left) and river networks in aerial images (right).

5. Conclusion

We proposed an algorithm to extract road networks in remote sensing images under the form of low-complexity graphs. Although conceptually simple, our algorithm relies upon mature geometric bricks that allow us to improve the geometric quality of the output graphs compared to prior works. In particular, it offers several interesting properties including (i) flexibility and generalization potential, (ii) geometric guarantees, and (iii) preservation of geometric regularities. We showed in our experiments the algorithm produces output graphs with a higher geometric quality while approaching the (node-based) accuracy scores of the best methods in the field.

In future work, we will investigate the use of more generic parametric curves than line-segments. In particular, we will study how to reach an optimal graph-complexity with Bézier curves. We will also investigate how to extend our approach to non-local geometric regularities with alternative optimization frameworks.

Acknowledgements

This work was carried out within the framework of the Intelligent_Mapping project, a component of the targeted IRIMA Platforms project, funded by the French Research Agency (ANR-22-EXIR-0008).

References

Agarwal, P. K., Har-Peled, S., Mustafa, N. H., Wang, Y., 2002. Near-linear time approximation algorithms for curve simplification. *Proc. of the Annual European Symposium on Algorithms*.

Ameri, F., Valadan Zoej, M. J., 2015. Road vectorisation from high-resolution imagery based on dynamic clustering using particle swarm optimisation. *The Photogrammetric Record*, 30(152).

Bahl, G., Bahri, M., Lafarge, F., 2022. Single-Shot End-to-end Road Graph Extraction. *Computer Vision and Pattern Recognition Workshops (CVPRW)*.

Bauchet, J.-P., Lafarge, F., 2018. KIPPI: KInetic Polygonal Partitioning of Images. *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Biagioni, J., Eriksson, J., 2012. Inferring Road Maps from Global Positioning System Traces: Survey and Comparative Evaluation. *Transportation Research Record*, 2291(1).

Boyer, M., Youssefi, D., Lafarge, F., 2024. Linefit: A geometric approach for fitting line segments in images. *European Conference on Computer Vision (ECCV)*.

Chen, Z., Deng, L., Luo, Y., Li, D., Marcato Junior, J., Nunes Gonçalves, W., Awal Md Nurunnabi, A., Li, J., Wang, C., Li, D., 2022. Road extraction in remote sensing data: A survey. *International Journal of Applied Earth Observation and Geoinformation*, 112.

Chu, H., Li, D., Acuna, D., Kar, A., Shugrina, M., Wei, X., Liu, M.-Y., Torralba, A., Fidler, S., 2019. Neural Turtle Graphics for Modeling City Road Layouts . *International Conference on Computer Vision (ICCV)*.

de Goes, F., Cohen-Steiner, D., Alliez, P., Desbrun, M., 2011. An Optimal Transport Approach to Robust Reconstruction and Simplification of 2D Shapes. *Computer Graphics Forum*, 30(5).

Douglas, D. H., Peucker, T. K., 1973. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: the international journal for geographic information and geovisualization*, 10(2).

Etten, A. V., Lindenbaum, D., Bacastow, T. M., 2018. SpaceNet: A Remote Sensing Dataset and Challenge Series. *CoRR*, abs/1807.01232.

Favreau, J.-D., Lafarge, F., Bousseau, A., Auvolat, A., 2020. Extracting Geometric Structures in Images with Delaunay Point Processes. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 42(4).

He, S., Bastani, F., Jagwani, S., Alizadeh, M., Balakrishnan, H., Chawla, S., Elshrif, M. M., Madden, S., Sadeghi, M. A., 2020. Sat2Graph: Road Graph Extraction through Graph-Tensor Encoding. *European Conference on Computer Vision (ECCV)*.

Hetang, C., Xue, H., Le, C., Yue, T., Wang, W., He, Y., 2024. Segment Anything Model for Road Network Graph Extraction . *Computer Vision and Pattern Recognition Workshops (CVPRW)*.

Ke, Z., Tan, B., Zheng, X., Shen, Y., Wu, T., Xue, N., 2025. Scalelsd: Scalable deep line segment detection streamlined. *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., Dollár, P., Girshick, R., 2023. Segment anything. *International Conference on Computer Vision (ICCV)*.

Li, T.-M., Lukáč, M., Michaël, G., Ragan-Kelley, J., 2020. Differentiable Vector Graphics Rasterization for Editing and Learning. *ACM Trans. on Graphics*, 39(6).

Li, Y., Dang, B., Li, W., Zhang, Y., 2024. Glh-water: A large-scale dataset for global surface water detection in large-size very-high-resolution satellite imagery. *Proc.of the AAAI Conference on Artificial Intelligence*, 38number 20.

Li, Z., Wegner, J. D., Lucchi, A., 2019. Topological Map Extraction From Overhead Images . *International Conference on Computer Vision (ICCV)*.

Lu, X., Weng, Q., 2025. Deep learning-based road extraction from remote sensing imagery: Progress, problems, and perspectives. *ISPRS Journal of Photogrammetry and Remote Sensing*, 228.

Menten, M. J., Paetzold, J. C., Zimmer, V. A., Shit, S., Ezhov, I., Holland, R., Probst, M., Schnabel, J. A., Rueckert, D., 2023. A skeletonization algorithm for gradient-based optimization. *International Conference on Computer Vision (ICCV)*.

Otsu, N., 1979. A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1), 62-66.

Pautrat, R., Barath, D., Larsson, V., Oswald, M. R., Pollefeys, M., 2023. DeepLSD: Line Segment Detection and Refinement with Deep Image Gradients . *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Perron, L., Didier, F., n.d. Cp-sat.

Song, J., Miao, R., 2016. A Novel Evaluation Approach for Line Simplification Algorithms towards Vector Map Visualization. *ISPRS International Journal of Geo-Information*, 5(12).

Sotiris, A., Lucchi, A., Hofmann, T., 2023. Mastering spatial graph prediction of road networks. *International Conference on Computer Vision (ICCV)*.

Staal, J., Abràmoff, M. D., Niemeijer, M., Viergever, M. A., Van Ginneken, B., 2004. Ridge-based vessel segmentation in color images of the retina. *IEEE trans. on Medical Imaging*, 23(4).

Tian, X., Günther, T., 2024. A Survey of Smooth Vector Graphics: Recent Advances in Representation, Creation, Rasterization, and Image Vectorization. *IEEE Transactions on Visualization and Computer Graphics*, 30(3).

Wang, B., Liu, Q., Hu, Z., Wang, W., Wang, Y., 2023a. TERNformer: Topology-Enhanced Road Network Extraction by Exploring Local Connectivity. *Trans. on Geoscience and Remote Sensing*, 61.

Wang, L., Dai, M., He, J., Huang, J., 2023b. Regularized primitive graph learning for unified vector mapping. *International Conference on Computer Vision (ICCV)*.

Xu, Z., Liu, Y., Sun, Y., Liu, M., Wang, L., 2022. RNGDet++: Road Network Graph Detection by Transformer with Instance Segmentation and Multi-scale Features Enhancement. *arXiv preprint arXiv:2209.10150*.

Yin, P., Li, K., Cao, X., Yao, J., Liu, L., Bai, X., Zhou, F., Meng, D., 2025. Towards Satellite Image Road Graph Extraction: A Global-Scale Dataset and A Novel Method. *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhang, Z., Zhao, Z., Wang, D., Wang, L., 2024. Graphmorph: Tubular structure extraction by morphing predicted graphs. *Conference on Neural Information Processing Systems (NeurIPS)*.