AGILE: GIScience Series, 3, 4, 2022. https://doi.org/10.5194/agile-giss-3-4-2022
Proceedings of the 25th AGILE Conference on Geographic Information Science, 2022.
Editors: E. Parseliunas, A. Mansourian, P. Partsinevelos, and J. Suziedelyte-Visockiene.
This contribution underwent peer review based on a full paper submission.

What are intersections for pedestrian users?

Jean-Marie Favreau[1] and Jérémy Kalsron[1]

[1]Université Clermont Auvergne, CNRS, Mines de Saint-Étienne, Clermont-Auvergne-INP, LIMOS, F-63000 Clermont–Ferrand, France.

Correspondence: Jean-Marie Favreau (j-marie.favreau@uca.fr)

Abstract. The increase of accessibility and pedestrian data in geographic databases such as OpenStreetMap brings with it the possibility to find a number of applications for pedestrian users.

The way in which different urban spaces are crossed obviously depends on their nature. In particular, crossing an intersection is not the same as walking along a street. Intersections are particularly complex areas, where crossing is almost mandatory, often with several possible routes.

Although there are various works in the literature that are interested in locating these intersections in a road network, to our knowledge there is no work that deals with the precise segmentation of intersections at the scale of pedestrian use.

In this article, we propose an approach that allows us to segment the OpenStreetMap street network at the pedestrian level, by precisely identifying the boundaries between intersections and other spaces.

By combining the geometry, topology and semantics of the urban automobile network of OpenStreetMap, we propose an algorithm for locating elementary intersections, and then successively assembling them in a multi-scale approach, in order to obtain the intersections as they are considered by pedestrians during their movements. In particular, our approach relies on the elements that constitute the boundaries of these intersections, such as pedestrian crossings and traffic lights.

After presenting an implementation of this approach, we offer a number of results that illustrate the robustness of the proposed approach.

Keywords. OpenStreetMap, intersections, pedestrian users, graph segmentation

1 Introduction

If for a long time large geographic databases were not able to capture information at the fine scale of pedestrian movement, the participatory contribution model of OpenStreetMap has allowed the gradual emergence in recent years of very precise data describing the precise infrastructures and devices useful for modelling pedestrian paths and their accessibility.

The progressive availability of this pedestrian accessibility data, particularly in European, East Asian and North American countries, makes it possible to envisage their use to create specific representations and to propose tools dedicated to pedestrian mobility, in its diversity of needs and uses.

1.1 Context and Motivation

Public space in urban areas is nowadays mainly dedicated to the car, especially in a large part of Europe and North America (Colville-Andersen, 2018). In these urban areas, the space dedicated to the pedestrian user is generally limited to a strip of sidewalk on each side of the road, and pedestrian crossings, which are sometimes facilitated by the presence of a car control infrastructure (traffic lights, stop signs, give way signs, etc.).

Of course, pedestrian areas and low-traffic neighborhoods (*zones de rencontre* in French), where pedestrians share the road with vehicles in a "calmed" manner, are emerging but these developments are still rare. In this article, we are interested in pedestrian mobility, and we will therefore focus on sidewalks and pedestrian crossings.

For the pedestrian user, if sidewalks are safe spaces, it is pedestrian crossings that are the main issue of an urban path, particularly because here pedestrians meet places identified as possibly dangerous. The user familiar with the city will therefore adapt his route according to the number of crossings and their quality in terms of safety or ease of crossing.

Among pedestrians, people with visual impairments do not have direct access to the overall structure of the urban space when travelling. The use of GPS routing applications for the general public is of course an interesting aid, but the finesse of what they present does not really go

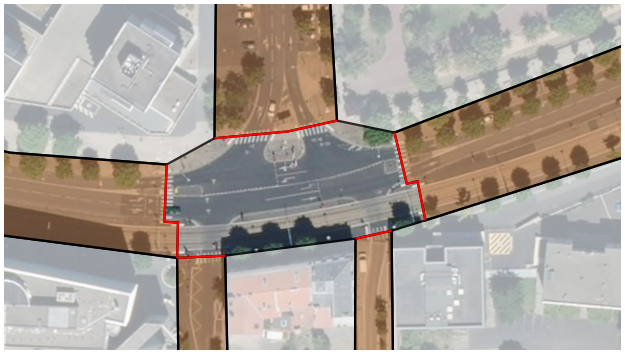


Figure 1. The area of an intersection from the pedestrian point of view, and the five branches of the intersection. The delimitation of these regions (red lines) corresponds to pedestrian crossings. Source: Géoportail © IGN.

down to the scale of the pedestrian path, leaving it up to the user to interpret crossings of intersections or the choice of sidewalk. There has been some work on the issue of accurate pedestrian routing to date (Cohen, 2017; Cohen and Dalyot, 2021), but little interest in producing a description of the environment (Kalsron et al., 2021; Boularouk et al., 2017), which is an essential tool for the visually impaired user to participate in the choice of route in the same way as other users.

In this article, we propose to extract a structure from the urban space that is modelled in such a way as to be able to construct a description that gives access to this understanding.

1.2 Intersections and branches

A first model of these spaces would be to propose a pedestrian traffic graph, where the paths would be described by polylines and nodes, augmented by semantic information identifying their typology: sidewalk, crossing, presence of traffic lights, warning strips, etc.

However, it is often necessary to have a broader and more synthetic view in order to choose an accessible route. Indeed, the quality of the pathway along the street may require a choice of one side over another. But more importantly, the crossing of intersections is often possible along several routes and requires a specific analysis. Depending on his needs, the pedestrian user will sometimes have to anticipate his route more or less according to the complexity of crossing intersections.

The identification in the urban space of the areas corresponding to these *intersections* as well as to the *branches* (i.e. adjacent streets) is therefore an essential step in understanding the pedestrian travel network (see example in Figure 1). As we will see later in this article, dealing with this issue from the point of view of the pedestrian user requires a very precise approach (at the micro-mapping scale), as the pedestrian infrastructure cannot be considered solely as the negative of that dedicated to car users.

(a) An intersection with its boundaries are defined by two pedestrian crossings (green lines) and one traffic light (red line).

(b) An intersection with its boundaries are defined by two pedestrian crossings (green lines) and one branch without marking (red line).

Figure 2. The boundaries of the intersections are located at the level of the infrastructures, if they exist. Source: Géoportail © IGN.

In the following, we will consider intersections as areas of public space where cars can change direction from one branch to another. From the point of view of pedestrian use, the boundary between the central region (the intersection) and the adjacent regions (the branches) will generally be located at the level of the pedestrian crossings for each branch near the intersection. If a branch does not have a pedestrian crossing, this separation will be located at the car stop (traffic light, yield sign, etc) as illustrated in Figure 2a. In the case where such a marking is not available, the junction will be delimited as close as possible, avoiding extending its area into the adjacent branch, as illustrated in Figure 2b.

1.3 Existing research on intersections detection and segmentation

The detection and processing of intersections in a road network is a topic that is very present in the literature, as it corresponds to important issues for many uses, starting with generalization (Mackaness and Mackechnie, 1999).

Much of the literature focuses on complex intersections, especially motorways, because their modelling requires many geometric primitives, although it may be necessary to simplify them significantly for cartographic rendering depending on the scale. Most studies therefore focus on the location of these complex structures in a road network (Li et al., 2020; Touya and Lokhat, 2020), and do not attempt to produce a precise segmentation of the region concerned.

Another more general approach is to detect patterns in the road network, for example by identifying the shape of a specific mesh such as regular grids, star structures, or ring roads (Zhang, 2004; Heinzle and Anders, 2007). These approaches can be used to locate intersections at the nodes of the mesh they identify in the road network. There are also approaches that aim to identify patterns in the road network at the intersection level, for example by detecting the characteristic shapes of triangular intersections, roundabouts or motorway interchanges (Savino et al., 2010), or by detecting intersections within their geometric shapes (T-nodes, y-nodes, fork nodes or CRS-nodes), and use

these for more global structure detection, such as areas of complex intersections (Touya, 2010). However, here again, the aim is not to produce a fine segmentation at the pedestrian level. These approaches therefore cannot be used directly in the context of our research.

While these first works mainly consider the mapping process and generalization, there are other uses that require the identification of intersections in a vehicle network. One example is the intelligent vehicle, whether it is to provide driving assistance or to go as far as the autonomous vehicle.

There is very little literature (Godoy et al., 2019) using traditional geographic databases (OpenStreetMap, BDTOPO IGN), because the precision required for these on-board techniques requires an accuracy of the order of a centimeter, which is found in so-called HD map databases (Seif and Hu, 2016), mainly reconstructed from LiDAR acquisitions or by processing photos with computer vision techniques (Guerrero et al., 2020). However, even with these data, it is interesting to note that the main subject of exploration concerns the identification of conflict zones between vehicles (Yao et al., 2020; Wang et al., 2020). This identification can be achieved using a more global model that considers the road structure, including intersections (Czarnecki, 2018), but once again, does not consider the point of view of the pedestrian, whose travel network remains in the area not covered by these models.

Another field where the identification of intersections plays an important role is the simulation of urban mobility, and the graphic representation of these traffic spaces and interactions between users. In this field, SUMO is a reference software, whose operation is precisely documented (Lopez et al., 2018). In particular, the network on which the models are based can be reconstructed from OpenStreetMap data. This includes both a model of pedestrian spaces, including sidewalks and pedestrian crossings, and a model of conflict zones, corresponding to the intersection of car traffic lanes. But again, the issue of intersections is not considered from the point of view of the pedestrian user, but only from the point of view of the car.

In summary, to our knowledge, to date, no research proposes to produce a segmentation of intersections at the level of the pedestrian user. Of course, there is work intended to use data to offer routing services to these users (Cohen and Dalyot, 2021), but these models do not propose precise segmentations of intersections. In the following sections, we propose and evaluate an algorithm dedicated to this issue, making the best use of geometric, topological and semantic data available in geographic data such as OpenStreetMap.

2 Data and Method

In this section, we start by presenting the state of the art of pedestrian mobility data modelling in OpenStreetMap, on the one hand noting the current modelling recommendations resulting from community decisions, and on the other hand reviewing the different variations that can be encountered in practice in the database.

We then propose an original method of intersection segmentation, exploiting both the geometric and semantic data available in OpenStreetMap, considering the level of modelling detail most commonly present in the level of modelling detail commonly present in the data on the European territory, particularly in France.

2.1 Pedestrian mobility data in OpenStreetMap

Historically and structurally, the primary backbone of the street infrastructure in OpenStreetMap is the street wireframe. Each street is described by at least one polyline augmented with semantic information. If the roadway has lanes separated by a fixed infrastructure, then there are as many lines as there are separate lanes. This is at least what the community expects, even if there are local variations resulting from personal interpretations of contributors, or vestiges of times when the modelling was less precisely specified.

The semantics carried by the polylines (**way**) and the points (**node**) make it possible to describe a large amount of information:

- For ways: road size, number of lanes, road width, traffic direction, traffic type,
- For nodes: pedestrian crossing, traffic lights, bicycle lanes, stop signs, yield signs, etc.

In addition to the nodes on the automobile network used to describe pedestrian crossings, the description of the pedestrian infrastructure can be modelled in several ways (Biagi et al., 2020), depending on the territory[1]:

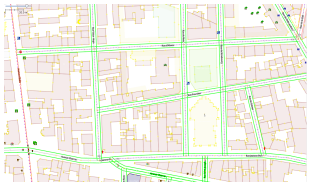
- by using semantic information on the automobile polylines[2] (Figure 3a),
- by adding pedestrian polylines[3], used to describe pedestrian crossings and sidewalks, and possibly supplemented with punctual information (node) or polylines describing the curb (Figure 3b),
- by the addition of surfaces representing the sidewalks thanks to their footprint[4] (Figure 3c).

However, the use of the two last models is still rare (table 1) and specific to cities where contributors have taken up the issue.

[1] https://wiki.openstreetmap.org/wiki/Sidewalks
[2] https://wiki.openstreetmap.org/wiki/Key:sidewalk
[3] https://wiki.openstreetmap.org/wiki/Tag:footway%3Dsidewalk
[4] https://wiki.openstreetmap.org/wiki/Tag:highway%3Dfootway#Areas

Tag	Number of objects
`sidewalk=*`	2 522 684
`footway=sidewalk`	2 693 888
`area` & `footway=sidewalk`	58 162
`crossing=*`	5 591 426
`highway=...`	132 281 352

Table 1. Number of objects contained in OpenStreetMap on February 2022. The first three lines corresponds to the three modellings of the pedestrian infrastructure. The fourth corresponds to the crossings, and the last one to the number of objects that are describing automobile ways (selected values for `highway=...`: `residential`, `service`, `primary`, `secondary`, `tertiary`, `motorway`, `living_street`, `trunk`).



(a) Semantic information in automobile polylines (light green dashed lines on each side of the polylines).

(b) Pedestrian polylines (dark green lines).

(c) Pedestrian areas (green areas).

Figure 3. OpenStreetMap data for three different neighborhoods with three different sidewalk modellings (JOSM screenshot).

In the following pages, we therefore propose to consider only the data carried by the automobile wire and the associated points to produce the segmentation. Thus, we assume that the following information (reconstructed or directly available) is available on this car line:

- The width of the road, either directly available or estimated from the type of road and the number of lanes that make it up,
- the type of road if it corresponds to an intersection infrastructure (link, roundabout),
- the name of the road,
- the type of node if it describes an item of infrastructure: pedestrian crossing, traffic light, stop sign, yield sign.

2.2 Modelling of intersections

If for the car, an intersection may be considered as a conflict zone, for the pedestrian it is an area of the urban space that cannot be crossed, and where several lanes are adjoined by sidewalks.

In this section, we propose a modelling of intersections as they can be formalised for pedestrian use, and we describe the way in which they can be reconstructed from a database as proposed in section 2.1.

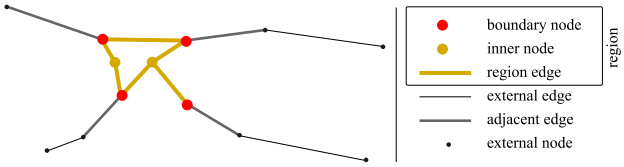


Figure 4. Example of a region defined on a geographic network.

2.2.1 Geographic network segmentation

In this part, we consider the automotive network as a non-oriented geographic network $G = (N, E)$ defined by a set of nodes N with coordinates, by a set of edges E defined as pairs of nodes of the graph, both nodes and edges being augmented with key-value semantic information. We also define the *cardinality* of a node as the number of edges that contain this node.

From the computer science point of view, this data structure can be seen as a planar graph embedded into the plane with coordinates and semantic information. Following the usages commonly associated with this formalism, we propose to describe in the following a **segmentation** of a geographic network by identifying for each **region** of this segmentation a non-empty set of nodes, and a (possibly empty) set of edges, with the constraint that if an edge belongs to the region, then its two end nodes also belong to it.

For each region, we can then identify two types of nodes: **inner nodes** (i.e. only connected to edges belonging to the region), and **boundary nodes**, i.e. connected to both at least one edge belonging to the region, and to an edge not belonging to the region (see Figure 4).

Adjacent edges to a region can also be identified as edges that are not part of the region, but one of whose ends belongs to the region.

2.2.2 Elementary intersection

The first stage of intersection detection that we propose is based on structured data as presented at the end of section 2.1. First, we propose to evaluate for each node the probability that it is an internal node of an intersection or an edge of an intersection.

A first identification allows us to easily characterise:

- Pedestrian crossings as probably strong nodes at the edge of an intersection.

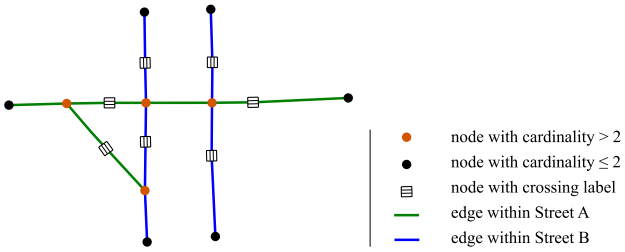


Figure 5. A graph with labels as it can be extracted from OpenStreetMap.

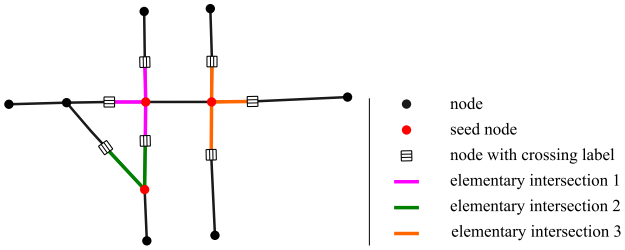


Figure 6. Elementary intersections extracted from the input graph shown in Figure 5.

- Nodes characteristic of a control of the flow of vehicles (traffic lights, stop signs, give way signs) as possibly edge nodes.

- Nodes with a cardinality greater than two, and whose adjacent edges have different road names, as most probably internal nodes of an intersection, and considered as *seed nodes* in the next steps of the algorithm.

From this first identification (see example Figure 5), we propose in section 3.2 an implementation that makes it possible to build a first segmentation into elementary intersections, using the width of the lanes and their semantics to guide the decision to extend and consolidate the segmentation from each of the *seed nodes* previously identified as internal to an intersection (see example Figure 6).

2.2.3 Multi-scale approach

However, a segmentation produced from the first markers does not fully correspond to the intuition of a pedestrian user. Indeed, this first segmentation produces small regions that are often close to each other and can be identified as sub-parts of a more complex intersection, as shown in Figure 6.

We therefore propose to model intersections not as a single segmentations, but as multi-scale structures, where *elementary intersections* are aggregated into *functional intersections*. Using the semantics, geometry and topology of the edges adjacent to each intersection, an algorithm for assembling elementary intersections into functional intersections is proposed.

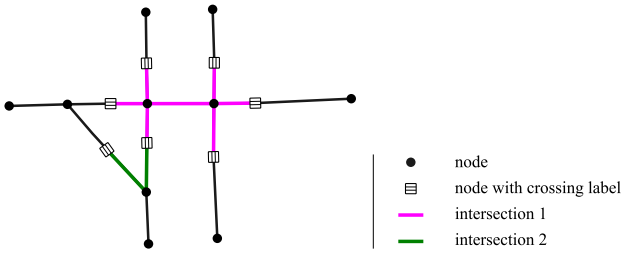


Figure 7. Result of the first assembly step from the elementary intersections shown in Figure 6.

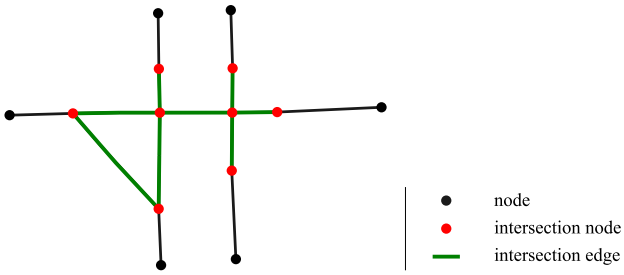


Figure 8. Result of the second assembly step from the intersections shown in Figure 7.

The first assembly step consists in identifying the pairs of intersections connected by a short path, and which have adjacent edges orthogonal to this connection path and bearing the same street name (see Figure 7).

From this first assembly, we then propose to identify the sets of intersections connected by a cycle of short paths, and to aggregate them into functional intersections. This allows us to take into account complex intersections with right-turn lanes, or intersections with multiple internal automobile routes (see Figure 8).

In these two assemblies, the notion of *short path* is constructed by considering a measure proportional to the width of the street, and by considering the semantics of the edges belonging to an intersection. For example, if an edge is labelled as part of an intersection, then its actual length is decreased. The resulting path length is compared to the road width (see details in section 3.2.4).

2.2.4 Branch segmentation

If the roadway is complex, it can happen that the branches of an intersection are composed of several edges, for example in the presence of a lane separator, or when an island separates the lane on the approach to the intersection. The final step of the segmentation consists in grouping edges by branches using semantic and geometric information (see example Figure 9).

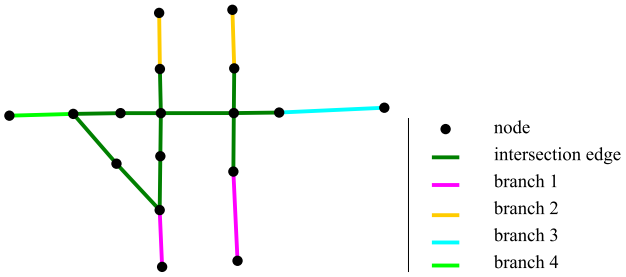


Figure 9. Branch segmentation from the functional intersection shown in Figure 8 and using the street names given in Figure 5.

3 Implementation and Experimental Results

In this section, we discuss an implementation of the multi-scale segmentation presented in section 2.2, and we present a series of experiments that allowed us to confront this implementation with real data from OpenStreetMap.

3.1 Data and software availability

Our method was applied to datasets downloaded and used without any modification in February 2022 from the OpenStreetMap collaborative database[5], available under the Open Database License (ODbL).

Under the open source license, we developed and distributed[6] a python implementation of our segmentation process using OSMnx libraryBoeing (2017), which is based on NetworkX[7].

We have also developed a randomized evaluation tool that allows us to compare the results of our segmentation with an expert eye. The tool, written in javascript, and available under open source license[8], allows a user to load the segmentation result calculated on a fixed area of interest, and then proposes to evaluate in a random order the intersections detected by the algorithm.

3.2 Implementation of the intersection segmentation

As described in the next sections, our implementation is provided with three parameters for the user to drive the segmentation:

- $\mathcal{C}_0$ to adjust the scale of the elementary intersections, used as a coefficient applied to the street's width to fix the maximum distance for a node to be considered as a boundary node of a given intersection.
- $\mathcal{C}_1$ to adjust the scale of the first merge of intersections, used as a coefficient applied to the street's widths to fix the maximum distance for two elementary intersections to be merged.
- $\mathcal{C}_2$ to adjust the scale of the final assembly for complex intersections, used as a coefficient applied to the street's widths to fix the maximum length for a link to connect functional intersections.

Several outputs are provided: an interactive displays, an output in geopackage for interoperability, and a json format as an interchange format with the evaluation tool we introduce in section 3.3.

3.2.1 Probability of belonging to an intersection or its boundary

In order to model the probability that a node or an edge belongs to an intersection and the probability that a node belongs to an intersection boundary, we propose to use a fuzzy representation: *strongly yes*, *moderate yes*, *weakly yes*, *uncertain*, *weakly no*, *moderate no*, *strongly no*.

A series of rules is then used to translate the semantics from OpenStreetMap into uncertainty:

- edges labelled as intersections belong to an intersection (*strongly yes*),
- nodes labelled as pedestrian crossings belong to the edge of an intersection (*strongly yes*),
- nodes corresponding to a traffic signal, a stop sign or a give way are moderately an intersection edge (*moderate yes*),
- nodes with cardinality greater than or equal to four belong inside an intersection (*strongly yes*),
- nodes with cardinality equal to three, and whose adjacent lane names are not unique, moderately belong to an intersection (*moderate yes*).

These rules make it possible to associate membership probabilities to certain nodes, and thus to initiate and guide the segmentation of intersections by considering a synthetic projection of the initial semantics into a simpler representation space.

3.2.2 Segmentation into elementary intersections

Each node being part of an intersection (*weakly yes* or higher), or belonging to an edge being part of an intersection (*weakly yes* or higher) is considered as a *seed node*. From each of these nodes, we propagate the construction of a region along the adjacent paths, integrating a path only if it reaches a node identified as a possible boundary of the intersection, and at a reasonable distance, depending on the width of the streets around the initial node.

If an identified possible boundary node is not strongly considered as a boundary (e.g. if it is a traffic light), the search

[5] https://www.openstreetmap.org/

[6] https://github.com/jmtrivial/crossroads-segmentation/releases/tag/agile-2022

[7] https://networkx.org/

[8] https://github.com/jmtrivial/crossroads-evaluation/releases/tag/agile-2022.

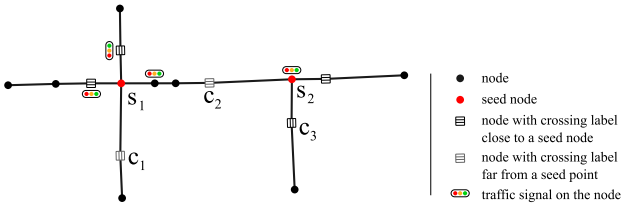


Figure 10. Two seed points for where elementary regions will be computed.

is still continued along the path, in case a strong boundary is present a little further on.

Figure 10 illustrate this process with two seed points with a representative set of configurations:

- The level of details may vary in the modelling: one intersection (denoted s_2) has a synthetic description of the traffic signal infrastructure, while the other one (s_1) is described in a more precise manner, each traffic signal being localised along the ways.
- Some pedestrian crossings (c_1 and c_2) are far from the intersections and may not be considered as boundaries of the intersections.

The proposal we made for the seed selection allows the two level of details of the modelling (high level with s_1 and low level with s_2) to be taken into account, since it only considers the topology of the network and the name of the streets.

The translation of OpenStreetMap tags into degree of membership of the boundary allows us to finely control the algorithm for generating elementary intersections without worrying about the semantics of the nodes. Thus, the intersection constructed from seed s_1 (Figure 10) will be composed of three paths, two for pedestrian crossings associated with traffic lights, and one for the traffic light without a pedestrian crossing.

In our implementation, a possible boundary node b is considered as an effective boundary of an intersection initiated by a seed node s if it satisfies the condition given in Eq. (1), where $\mathrm{length}(\cdot,\cdot)$ is the length of the path between b and s, $\mathcal{C}_0$ is a parameter of our method, E_s is the set of edges that contain s, and w_e is an estimation of the street's width.

$$\mathrm{length}(b,s) \leq \mathcal{C}_0 \max_{e \in E_s} w_e \tag{1}$$

In practice, we estimate the street's width using a typical lane width for each kind of street (3 meters for a lane in a way labelled `highway=primary`, 2.75 meters for a lane in a way labelled `highway=secondary`, etc.). We also noticed that a $\mathcal{C}_0 = 2$ gives good results in European historic centers.

In the proposed selection in Figure 10, both c_1 and c_2 are considered to be not part of the intersections, but c_3 is part of the intersection starting from s_2.

If the street connecting s_1 to s_2 had been narrower, the algorithm would not have considered c_3 as one of the nearby pedestrian crossings. Similarly, if this street had been wider, c_1 would have joined the pedestrian crossings near the intersection.

3.2.3 Merging elementary intersections by semantics and geometry

The elementary intersections in the same neighborhood are then merged by considering the geometry of their adjacent edges.

In our implementation, we compare for each pair of elementary intersections i_1 and i_2 the Euclidean distance $|s(i_1), s(i_2)$ between the seeds $s(\cdot)$ with a measure of proximity given in Eq. (2), where $\mathcal{C}_1$ is a parameter of our method, $w(i)$ is the maximum width of the streets in i, and $\alpha_{i_1,i_2} \in \{1, \frac{1}{2}\}$ depending on the presence of a possible boundary in the path connecting i_1 and i_2.

$$p_{i_1,i_2} = \mathcal{C}_1 \alpha_{i_1,i_2} \max\{w(i_1), w(i_2)\} \tag{2}$$

The pair of intersections is then considered depending on this proximity:

- if $|s(i_1), s(i_2) > p_{i_1,i_2}$, the two intersections are not neighbor intersections,
- if $|s(i_1), s(i_2) \leq \frac{1}{2} p_{i_1,i_2}$, the two elementary intersections are considered to be part of the same functional intersection,
- between these two values, we consider the angles and names of the adjacent edges to decide whether the two elementary intersections are in the same functional intersection. For each pair of neighboring intersections, we identify pairs of adjacent edges with the same name, which are oriented in a consistent direction (with a relative angle of less than 90°), and whose orientation relative to the segment defined by the seeds of the two intersections is significantly different (by an angle greater than 45°). If such a pair of edges exists, then the two elementary intersections are considered to be part of the same functional intersection. Figure 11 illustrate this computation showing two elementary intersections with two pairs of adjacent edges, each of them in the expected configuration to merge the intersections.

In practice, we have noticed that a $\mathcal{C}_1 = 2$ gives good results in European historic centers.

A new region is therefore created by assembling all the points and edges making up the initial intersections, and adding the points and edges constituting the path which connects the merged intersections.

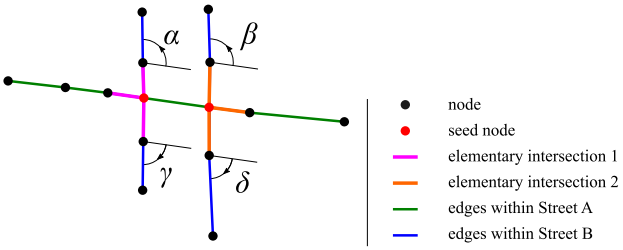


Figure 11. Two elementary intersections in which the proximity between seed points requires consideration of the configuration of adjacent edges. These two elementary intersections have two pairs of adjacent edges ($\{\alpha, \beta\}$ and $\{\gamma, \delta\}$) with the same name, consistent directions and significantly different directions with respect to the segment defined by the two initial seeds.

3.2.4 Assembling the intersections

Once all the intersections have been identified and merged by semantics, they are assembled into functional intersections.

The first step is to identify all the connected components of the complementary of the previously computed intersections. These related components constitute the regions of possible links between the different intersections. In each of these related components, we identify the set of linking paths, i.e. paths that connect two distinct intersections.

In order to drive the size of generated intersections, we define for a linking path k an adapted length defined by Eq. (3), depending on the probability that it belongs to an intersection, by weighting the length $\text{length}(k)$ of the link, where E_k is the set of edges within k, n_k is the number of nodes in k with a cardinality greater than three, and $p_e = \frac{1}{2}$ if e is labelled as part of an intersection and 1 otherwise.

$$\text{length}'(k) = \frac{\sum_{e \in E_k} \text{length}(e) p_e}{\log e^{n_k + 1}} \tag{3}$$

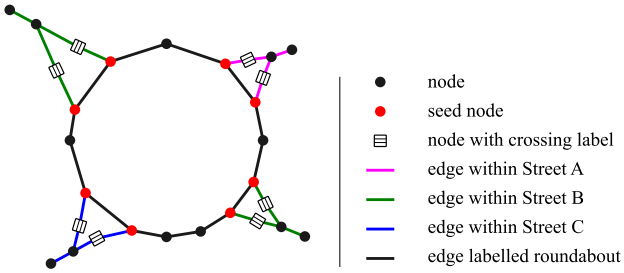
We then look for the set of cycles containing an alternation of intersections and linking paths, retaining in this process a loop l if it satisfies the following two conditions, driven by a parameter $\mathcal{C}_2$. First, we only select short linking paths k that verify Eq. (4), where w_k the maximum width of the lanes in k.

$$\text{length}'(k) \leq w_k \mathcal{C}_2 \tag{4}$$

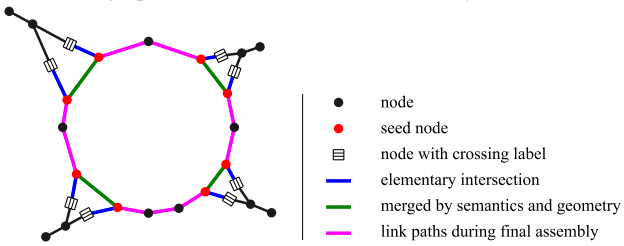
From these linking paths, we only select short cycles, using the $\mathcal{C}_2$ as a parameter of our method, as described in Eq. (5), where L_l is the set of the linking paths within l.

$$\sum_{k \in L_l} \text{length}'(k) \leq \max_{i \in I_l} \{w_i\} \pi \mathcal{C}_2 \tag{5}$$

In practice, we have noticed that a $\mathcal{C}_2 = 4$ gives good results in European historic centers, when the parameter could be increased to capture larger intersections, especially around expressways and motorways (see experimental results in section 3.4.1).



(a) Initial graph where three streets are linked by a roundabout.



(b) Step-by-step of the roundabout as an intersection. After identifying eight seed nodes, each elementary intersection is constructed adding a path to the closest pedestrian crossing. Since each pair of intersection has an adjacent lane with the same street name, they are assembled in four intersections. Finally, a loop is identified by connecting these intersections along the paths of the roundabout.

Figure 12. Illustration of the pipeline in a roundabout.

The functional intersections are then the result of the assembly of the intersections and the linking paths of these cycles.

The approach we propose here has the advantage of handling both complex intersections containing right-turn lanes (Figure 8), as well as complex intersections containing polylines describing internal paths (Figure 17), but also roundabouts (Figure 12).

3.2.5 Identification of branches

For each intersection assembled in this way, we identify its branches by constructing a list of all the external edges connected to a vertex belonging to the edge of the intersection. These edges are then assembled by grouping them together if they have the same name and are oriented in a consistent direction (with a relative angle of less than 90°), as illustrated in Figure 13.

3.3 Experimental Design

In order to evaluate the quality of the segmentation produced by our algorithm, we proceeded in two different ways.

First, by exploring the map proposed by OpenStreetMap, we identified typical intersections, more or less complex,

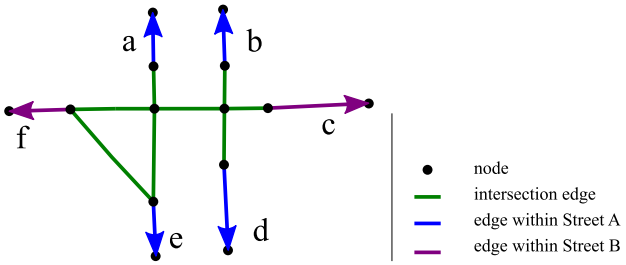


Figure 13. Each adjacent edge of this intersection is labelled with a street name. We can group them using these names and a consistent direction to compute the corresponding branches: $\{a, b\}$, $\{c\}$, $\{d, e\}$, $\{f\}$.

to challenge our algorithm. Some images are proposed in the section 3.4.1, and allow us to see these results.

As introduced in 3.1, we developed a randomized evaluation tool that allows us to compare the results of our segmentation with an expert eye. The tool allows a user to load the segmentation result calculated on a fixed area of interest, and then proposes to evaluate in a random order the intersections detected by the algorithm.

The evaluation interface (Figure 14) is composed of two panels: on the left, a simple form allows the user to indicate in a few clicks the possible defects of the segmentation:

- Existing intersection: *yes* or *no*,
- Intersection scale: *correct*, *too large* or *too small*,
- Number of branches: *correct*, *too few* or *too much*,
- Configuration of branches: *correct*, *two or more branches are merged*, *one or more branch is split* or *merge and split branches*,
- Boundary position (relatively to the intersection center): *correct*, *too close* or *too far*,
- Completeness: *correct*, *missing parts* or *excess parts*.

On the right, the intersection is represented by a set of polylines with colors corresponding to the intersection itself and to the different branches of this intersection. The entire set is drawn on an orthophotography, and a series of buttons can be used on demand to display the intersection in usual web tools (OpenStreetMap, Google Maps, Google Street View).

The tool generates an evaluation file for each area of interest, which can be explored with a dedicated interface (Figure 15), in order to have a synthetic overview of the quality of the segmentation in the considered area.

In section 3.4.2, we present a statistical evaluation of the segmentation applied in a series of large neighbourhoods from various cities in Europe.

Finally, in order to show the generality of the proposed approach, in section 3.4.3, we present some results of the

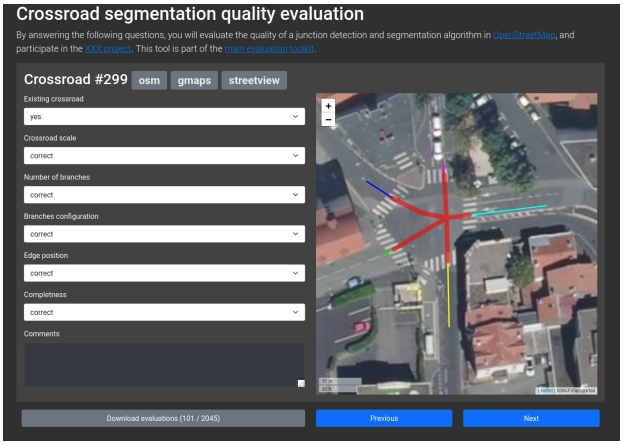


Figure 14. Interface of the evaluation tool.

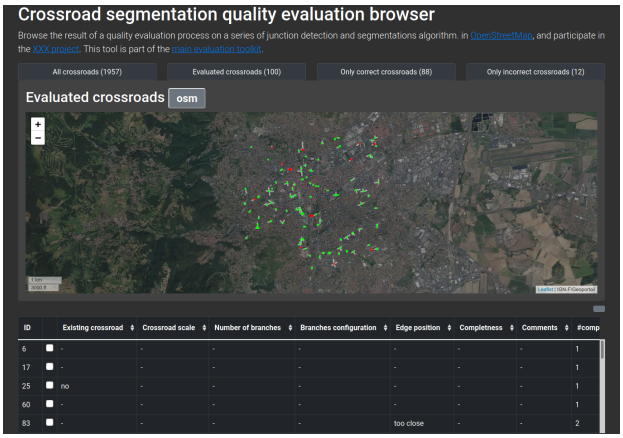


Figure 15. Interface used to explore the performed evaluations.

segmentation of intersections located outside France (our first experimentation field).

3.4 Results

In this section, we present different experimental results based on OpenStreetMap data used without modification. Depending on the area, it may be the case that the data are partial, for example missing information that could have been useful for the segmentation. In particular, one may encounter intersections where one finds:

- the semantics giving the presence of pedestrian crossings or traffic lights,
- the name of the street on some edges (or even the presence of names with diverse semantics),
- the semantics indicating that the edges are part of an intersection (rather rare data in OpenStreetMap in 2022 in a large part of the country).

However, even with these possible missing data, we observe a rather precise segmentation of intersections, as presented here.

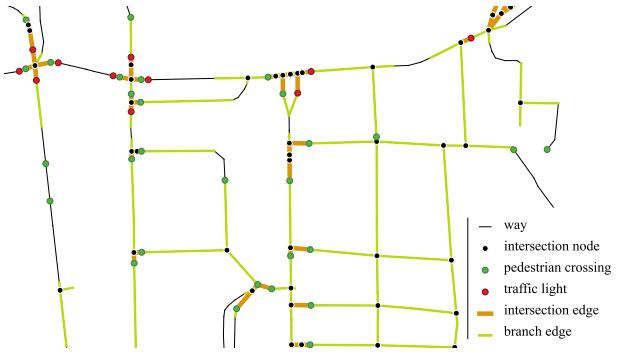


Figure 16. Result of the segmentation process in a neighbourhood.

3.4.1 Examples of segmentation

The urban structure varies greatly from one city to another, depending on the history of the city's construction. For example, medium-sized European cities have a fairly simple grid, regular in places and less aligned in others. It is especially noticeable that intersections have a very simple modelling in OpenStreetMap. On this kind of mesh, the segmentation of intersections is very simple, and the results we obtain correspond to what is expected (Figure 16).

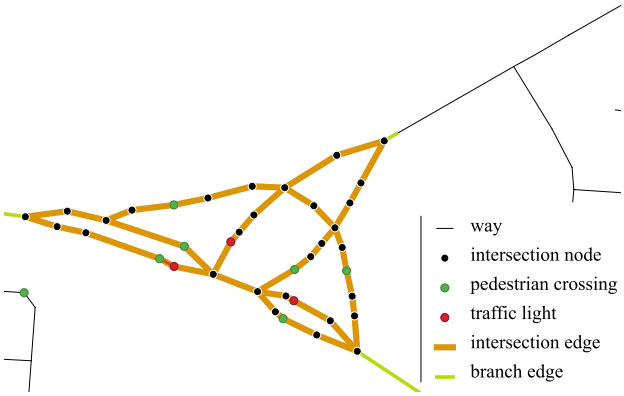
As we approach the first suburbs of these historic centers, we find areas where the space dedicated to the car increases. The intersections become larger and more complex. It is particularly for this type of intersection that our approach has been designed, as pedestrian users are frequently confronted with crossings at these intersections (Figure 17).

Roundabouts are also very classic patterns in urban settings, especially in Western Europe. Here again, our modelling allows us to capture these geometries (Figure 18).

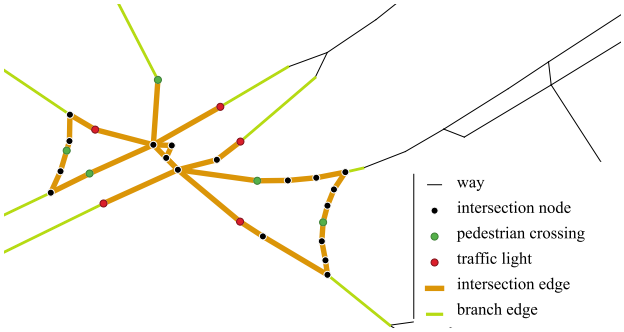
The C_2 parameter we introduced in section 3.2.4 allows us to control the extent of the reconstructed regions, in the case of complex intersections whose edges have not been labelled in OpenStreetMap as part of an intersection. This can be seen in the example shown in the figure where three small intersections are articulated around a central triangle-shaped space, which some users tend to consider as a single large intersection. The segmentation can thus render the configuration with three intersections (Figure 19a, $C_2 = 4$) or the configuration with a single large intersection (Figure 19b, $C_2 = 5$).

3.4.2 Statistical evaluation of the quality of segmentation

For the statistical evaluation, we have selected three cities of representative size for French cities: city 1 (Paris, 10,785,092 inhabitants in the urban area), city 2 (Nantes, 650,081 inhabitants in the urban area) and city 3 (Clermont-Ferrand, 268,696 inhabitants in the urban area). On each of them, we selected a point and get all the Open-



(a) An intersection with three branches.



(b) An intersection with six branches.

Figure 17. Segmentation of intersections with complex internal paths.

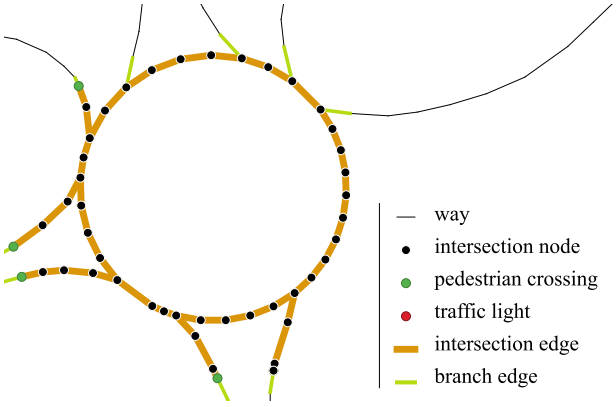
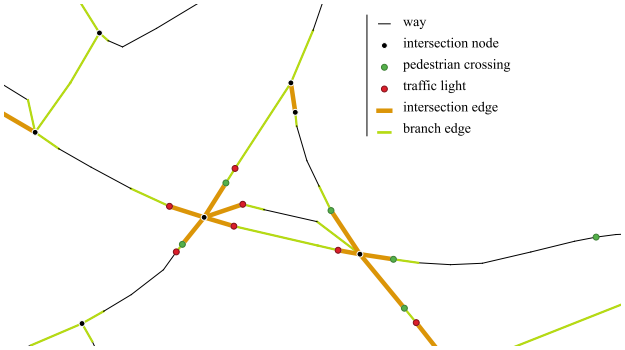
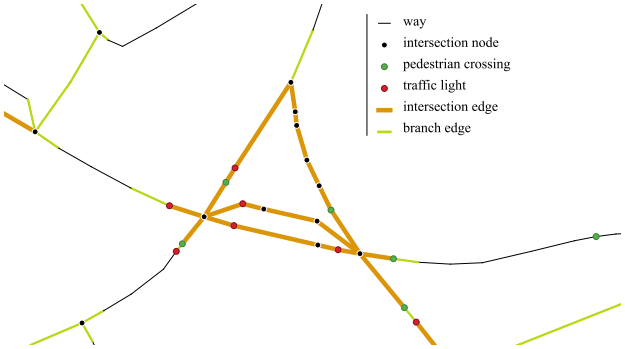


Figure 18. Segmentation of a roundabout.



(a) Segmentation on a complex area with $C_2 = 4$: three independent intersections.



(b) Segmentation on a complex area with $C_2 = 5$: a single large intersection.

Figure 19. The granularity of the segmentation of a complex area adjustable by the user through the C_2 parameter.

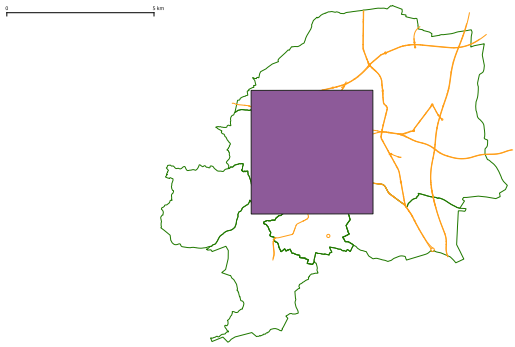
Region	#total	#simple	#intermediate	#complex
1	1,818	265	1,541	12
2	1,778	573	1,190	15
3	1,957	1,006	931	20
all	5,553	1,844	3,662	47
ratio		33.2%	65.9%	0.8%

Table 2. Complexity of the generated intersections in each region: intersections with a single node, intersections with several nodes, of which only one has a cardinality greater than 2, intersections with several nodes of cardinality greater than 2.

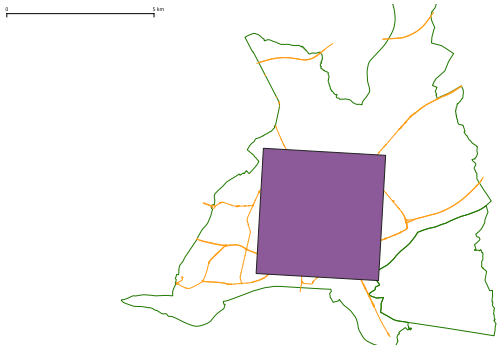
StreetMap data within a distance of two kilometers from this point (see Figure 20).

We applied the segmentation process on each of the regions, obtaining a total of 5,553 intersections. Table 2 gives for each region the distribution of intersections according to their complexity: intersections with a single node, intersections with several nodes, of which only one has a cardinality greater than 2, intersections with several nodes of cardinality greater than 2.

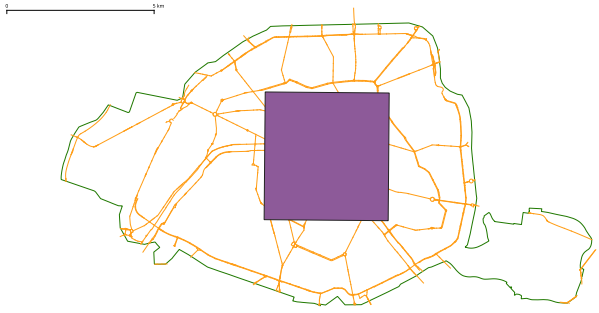
We used the evaluation tool introduced in section 3.3 in each region, randomly evaluating 100 intersections in each region. Table 3 shows that the distribution of intersections by complexity is comparable to that of all regions.



(a) Selected region centered in the historical center of Clermont-Ferrand. The region contains streets that are part of the following cities: Clermont-Ferrand, Aubière, Beaumont, Ceyrat, Royat and Chamalières.



(b) Selected region centered in the historical center of Nantes. The region also contains streets that are part of Saint-Sébastien sur Loire.



(c) Selected region centered in Paris.

Figure 20. Selected regions used for statistical evaluation.

Region	#total	#simple	#intermediate	#complex
1	100	29	70	1
2	100	17	83	0
3	100	52	45	3
all	300	98	198	4
ratio		32.7%	66%	1.3%

Table 3. Complexity of the randomly selected intersections: intersections with a single node, intersections with several nodes, of which only one has a cardinality greater than 2, intersections with several nodes of cardinality greater than 2.

Region	#total	#selected	#valid segmentation
1	1,818	100	81
2	1,778	100	76
3	1,957	100	88

Table 4. Dataset description: for each region, the number of intersections obtained by our algorithm, the number of randomly selected intersections (100), and the number of valid segmentation with respect to the orthophotography.

Of these 300 intersections, 245 were evaluated as valid by comparing the segmentation with the orthophotography proposed by the interface (see Table 4).

Of the remaining 55 intersections, 26 are affected by a boundary adjustment problem, with the orthophotography showing a pedestrian crossing that should have been included in the segmentation. After analysis, 11 are affected by missing pedestrian crossings in OpenStreetMap, 3 because of an exceeded or incorrectly positioned pedestrian crossing, and 2 others because an adjacent street is a pedestrian street according to OpenStreetMap, but which is not suggested by the orthophotography. The remaining 10 misplaced boundaries would require local adjustment of the C0 parameter.

Of the remaining intersections, 21 were considered to have missing parts or to be too large or too small in scale. There are several causes here:

- for 2 of them, it is a missing lane in the OpenStreetMap modelling,
- for 3 others, an adjacent street was identified in OpenStreetMap as not accessible to cars,
- for 4 others, they are roads bordering or entering a square, spaces not taken into account by the algorithm,
- for 12 of them, the local adjustment of the C1 or C2 parameter would have allowed the segmentation to be corrected.

Two intersections were evaluated as non-existent. After analysis, these were approximations in the OpenStreetMap modelling (incorrectly labelled private yard service, inaccurate modelling of a multiple lane crossing).

One intersection had its three branches combined into one branch. The algorithm failed to correctly interpret a T-junction in a residential neighbourhood where each branch had the same name.

One intersection had one of its branches split into two branches. After analysis, these two lanes corresponded to the service of a car park, and were without names, preventing the algorithm from being able to associate them.

Finally, we also identified an intersection located in the middle of a park, and 3 intersections in the middle of car parks, which invite further work on the filtering of the OpenStreetMap data.

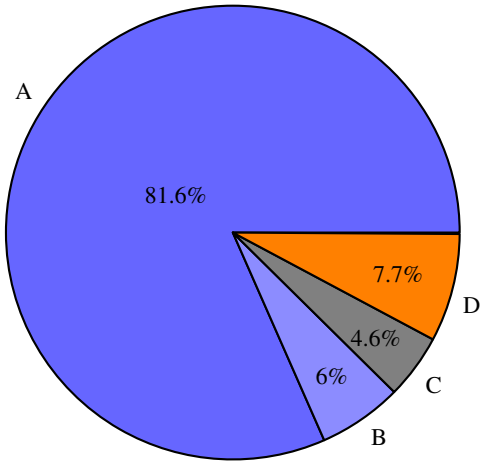


Figure 21. Distribution of intersections considered during our evaluation process (Table 4) according to their typology. A (245 intersections): segmentation corresponding to the orthophotography, B (18 intersections): lack or inaccuracy in OpenStreetMap, C (14 intersections): not supported regions, D (23 intersections): requires local adjustment of parameters.

In summary (Figure 21), the segmentation of 23 intersections can be corrected by locally adjusting one of the three parameters of the method, 18 intersections are inaccurate due to lack or inaccuracy of data in OpenStreetMap, and 14 intersections illustrate the lack of generality of our algorithm, which does not take into account lanes closed to cars, unnamed lanes, intersections with the same names on all branches, or the context (square, park, car park).

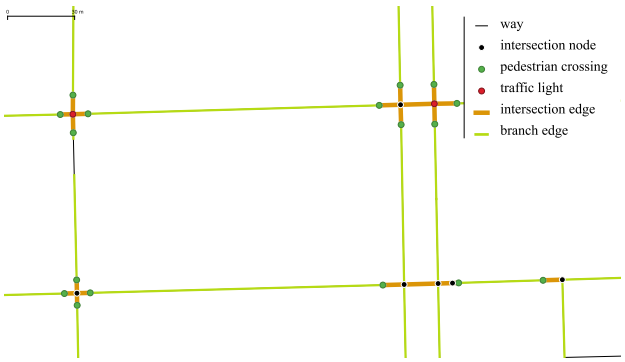
3.4.3 Additional examples

In order to illustrate the genericity of the proposed approach, we applied our implementation on several locations, using the default parameters discussed in the previous section.

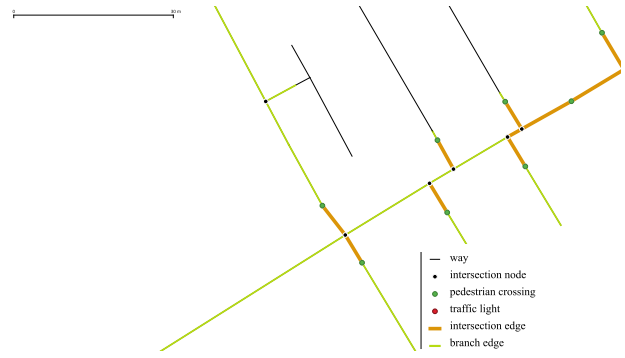
During these experiments, we identified that large cities in Europe and North America were described in OpenStreetMap with the same degree of quality as the cities we studied in France. Without changing the parameters of our methods, we obtained good segmentation qualities (Figure 22).

Conversely, small villages in France, but also places such as Tokyo suburbs or Conakry where the habits of contributing to OpenStreetMap are different, are not documented as precisely. In particular, few pedestrian crossings are observed in the manipulated data, resulting in fewer operational segmentations of intersections (Figure23).

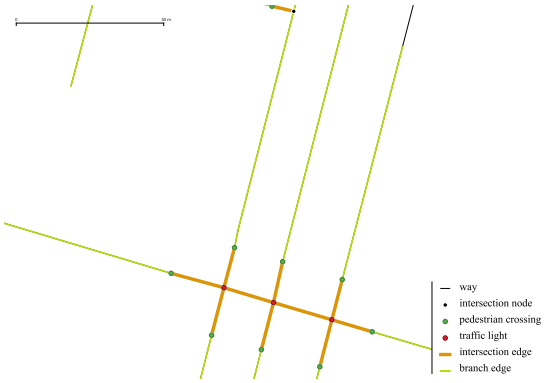
While the stability of the segmentation obtained does not seem to require adjustment of the parameters of our method in the examples covered during this exploration, it is likely that if the structure of the network is different (for example, on the approaches to motorways), our algorithm may need to be adjusted. However, we have chosen



(a) New York (USA)

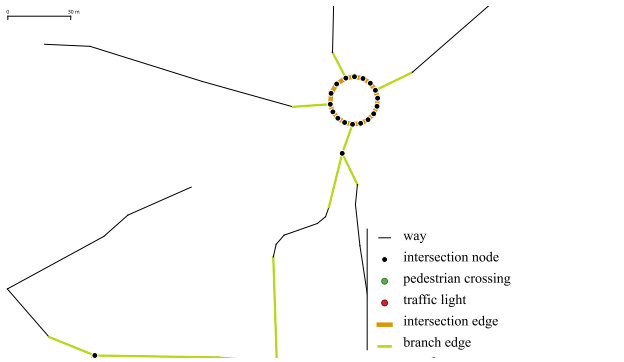


(b) Warsaw (Poland)

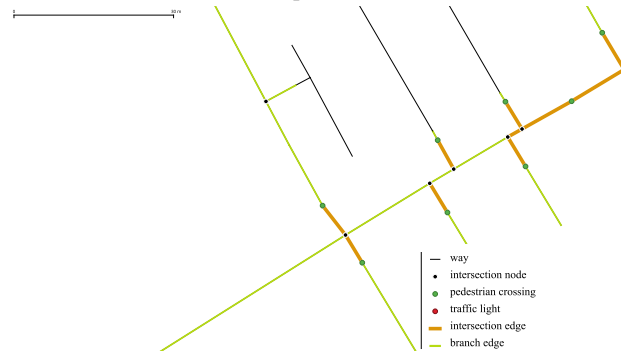


(c) Genoa (Italy)

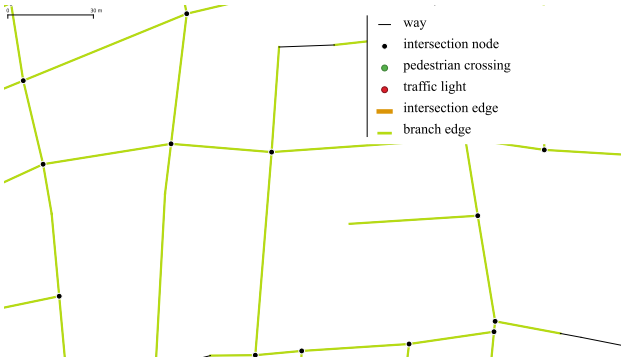
Figure 22. Segmentation results on neighborhoods where pedestrian crossings are described in OpenStreetMap data.



(a) Champeix (France)



(b) Conakry (Guinea)



(c) Tokyo (Japan)

Figure 23. Segmentation results on neighborhoods where pedestrian crossings are described in OpenStreetMap data.

not to focus on these areas, as they do not correspond to pedestrian use.

4 Conclusion and Future Work

In this paper, we have presented a method for segmenting intersections taking into account the needs of pedestrians. After having identified the intersection cores, the intersection boundaries are adjusted to the existing infrastructures (pedestrian crossings, traffic lights, etc.), then an aggregation approach of these elementary intersections provides functional intersections, i.e. corresponding to the practice of pedestrian users. After presenting a detailed implementation of this method, we tested it against a dataset taken directly from OpenStreetMap. We have thus shown the good performance of the algorithm, and presented some limitations.

Therefore, the next steps in improving the algorithm could concern the consideration of specific configurations, such as squares and pedestrian street approaches. Greater care will have to be taken in filtering the data, in order to eliminate possible persistent regions, corresponding to parks and car parks.

Beyond these first observations, we plan to improve the estimation of the width of the lanes, in particular by integrating into this evaluation the neighbouring buildings, or the items of infrastructure in the immediate surroundings, such as parking areas or cycle lanes and paths.

The results of this segmentation can be improved, but by allowing the user to act on the three parameters of our method, or even by proposing to modify the data if they do not reflect the reality of the terrain, we can now envisage the use of this segmentation algorithm in practical applications.

In particular, for our team, this tool is part of a larger processing chain, where the objective is to produce a textual description of a crossroads for users with visual impairments. The results of the segmentation, permit not only the identification of the intersection region, but also of the branches, and will be a useful addition to the OpenStreetMap data to assist with information extraction and textual restitution algorithms.

Furthermore, this segmentation algorithm and its implementation could have several direct uses, notably for the OpenStreetMap community, which could integrate it into its data quality monitoring process, in order to quickly identify areas where the modelling is not consistent with the infrastructure visible on the orthophotography.

This segmentation could also feed into tools already based on OpenStreetMap, such as pedestrian routing tools, or micro-traffic simulation tools.

5 Acknowledgements

The authors of this article would like to thank Guillaume Touya for the various exchanges that helped to improve the presentation of this work. This work was funded by the French National Research Agency, via the ACTIVmap project (ANR-19-CE19-0005).

References

Biagi, L., Brovelli, M. A., and Stucchi, L.: Mapping the Accessibility in Openstreetmap: a Comparison of Different Techniques, ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 43B4, 229–236, https://doi.org/10.5194/isprs-archives-XLIII-B4-2020-229-2020, 2020.

Boeing, G.: OSMnx: New methods for acquiring, constructing, analyzing, and visualizing complex street networks, Computers, Environment and Urban Systems, 65, 126–139, 2017.

Boularouk, S., Josselin, D., and Altman, E.: Ontology for a voice transcription of OpenStreetMap data: the case of space apprehension by visually impaired persons, in: World Academy of Science, Engineering and Technology, London, United Kingdom, https://hal.archives-ouvertes.fr/hal-01533064, 2017.

Cohen, A.: Building a weighted graph based on OpenStreetMap data for routing algorithms for blind pedestrians, Technion-Israel Institute of Technology, 2017.

Cohen, A. and Dalyot, S.: Route planning for blind pedestrians using OpenStreetMap, Environment and Planning B: Urban Analytics and City Science, 48, 1511–1526, 2021.

Colville-Andersen, M.: The Arrogance of Space, pp. 90–95, Island Press/Center for Resource Economics, Washington, DC, https://doi.org/10.5822/978-1-61091-939-5_9, 2018.

Czarnecki, K.: Operational world model ontology for automated driving systems–part 1: Road structure, Waterloo Intelligent Systems Engineering Lab (WISE) Report, 2018.

Godoy, J., Artuñedo, A., and Villagra, J.: Self-generated osm-based driving corridors, IEEE Access, 7, 20 113–20 125, 2019.

Guerrero, J., Chapuis, R., Aufrère, R., Malaterre, L., and Marmoiton, F.: Road Curb Detection using Traversable Ground Segmentation: Application to Autonomous Shuttle Vehicle Navigation, in: 2020 16th International Conference on Control, Automation, Robotics and Vision (ICARCV), pp. 266–272, https://doi.org/10.1109/ICARCV50220.2020.9305304, 2020.

Heinzle, F. and Anders, K.-H.: Characterising Space via Pattern Recognition Techniques: Identifying Patterns in Road Networks, in: The Generalisation of Geographic Information : Models and Applications, edited by Mackaness, W. A., Ruas, A., and Sarjakoski, T., Elsevier, 2007.

Kalsron, J., Favreau, J.-M., and Touya, G.: Le carrefour dont vous êtes le héros, Spatial Analysis and Geomatics (SAGEO) 2021, https://hal.archives-ouvertes.fr/hal-03263279, poster, 2021.

Li, C., Zhang, H., Wu, P., Yin, Y., and Liu, S.: A complex junction recognition method based on GoogLeNet model, Transactions in GIS, 24, 1756–1778, https://doi.org/10.1111/tgis.12681, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/tgis.12681, 2020.

Lopez, P. A., Behrisch, M., Bieker-Walz, L., Erdmann, J., Flötteröd, Y.-P., Hilbrich, R., Lücken, L., Rummel, J., Wagner, P., and Wießner, E.: Microscopic traffic simulation using sumo, in: 2018 21st international conference on intelligent transportation systems (ITSC), pp. 2575–2582, IEEE, 2018.

Mackaness, W. A. and Mackechnie, G. A.: Automating the Detection and Simplification of Junctions in Road Networks, Geoinformatica, 3, 185–200, https://doi.org/10.1023/a:1009807927991, 1999.

Savino, S., Rumor, M., Zanon, M., and Lissandron, I.: Data enrichment for road generalization through analysis of morphology in the CARGEN project, in: Proceedings of 13th ICA Workshop on Generalisation and Multiple Representation, Zurich, Switzerland, https://kartographie.geo.tu-dresden.de/downloads/ica-gen/workshop2010/genemr2010_submission_21.pdf, 2010.

Seif, H. G. and Hu, X.: Autonomous driving in the iCity—HD maps as a key challenge of the automotive industry, Engineering, 2, 159–162, 2016.

Touya, G.: A Road Network Selection Process Based on Data Enrichment and Structure Detection, Transactions in GIS, 14, 595–614, https://doi.org/10.1111/j.1467-9671.2010.01215.x, 2010.

Touya, G. and Lokhat, I.: Deep Learning for Enrichment of Vector Spatial Databases: Application to Highway Interchange, ACM Transactions on Spatial Algorithms and Systems, 6, 21, https://doi.org/10.1145/3382080, publisher: ACM, 2020.

Wang, Y., Cai, P., and Lu, G.: Cooperative autonomous traffic organization method for connected automated vehicles in multi-intersection road networks, Transportation research part C: emerging technologies, 111, 458–476, 2020.

Yao, Z., Jiang, H., Cheng, Y., Jiang, Y., and Ran, B.: Integrated Schedule and Trajectory Optimization for Connected Automated Vehicles in a Conflict Zone, IEEE Transactions on Intelligent Transportation Systems, pp. 1–11, https://doi.org/10.1109/TITS.2020.3027731, 2020.

Zhang, Q.: Modelling Structure and Patterns in Road Network Generalization, in: ICA Workshop on Generalisation and Multiple Representation, Leicester, UK, https://kartographie.geo.tu-dresden.de/downloads/ica-gen/workshop2004/Zhang-v2-ICAWorkshop.pdf, event-place: Leicester, UK, 2004.