A Framework for Automated Junction Monitoring

David, Rodrigo Sasse; Torp, Kristian; Sakr, Mahmoud Attia; Zimányi, Esteban

Published in:
SIGSPATIAL '24: Proceedings of the 32nd ACM International Conference on Advances in Geographic Information Systems

DOI (link to publication from Publisher):
10.1145/3678717.3691322

Publication date:
2024

Document Version
Publisher's PDF, also known as Version of record

Citation for published version (APA):
David, R. S., Torp, K., Sakr, M. A., & Zimányi, E. (2024). A Framework for Automated Junction Monitoring. In *SIGSPATIAL '24: Proceedings of the 32nd ACM International Conference on Advances in Geographic Information Systems* (pp. 304-313). Association for Computing Machinery (ACM). https://doi.org/10.1145/3678717.3691322

A Framework for Automated Junction Monitoring

Rodrigo Sasse David
Aalborg University
Aalborg, Denmark
rsd@cs.aau.dk

Kristian Torp
Aalborg University
Aalborg, Denmark
torp@cs.aau.dk

Mahmoud Sakr
Uni. Libre de Bruxelles
Brussels, Belgium
mahmoud.sakr@ulb.ac.be

Esteban Zimányi
Uni. Libre de Bruxelles
Brussels, Belgium
ezimanyi@ulb.ac.be

ABSTRACT

Monitoring roundabouts and signalized intersections in a road network is important, e.g., to reduce travel time and greenhouse gas emissions. The monitoring of such junctions is a challenging problem, and current approaches mainly use high-cost solutions for a selected few. In this work, we present a framework for the automated identification and monitoring of all junctions in a road network. The framework utilizes detailed trajectory data or high-level segment-based data to compute travel time and energy consumption for all turn directions. These metrics are then aggregated per junction to enable a fair comparison between roundabouts and intersections. The aggregated metric is used to provide an overview of all junctions and to pinpoint those performing poorly. An analysis of 1,394 junctions using 334,081 trajectories quantifies the different benefits of roundabouts and intersections, e.g., the travel time in roundabouts varies little, and turns are 21% to 155% more energy-consuming than going straight in intersections. Further, the aggregated junction metric makes it simple to monitor all analyzed junctions and detect the worst-performing. The analysis also clearly shows the benefits of trajectory data over segment-based data for junction monitoring.

CCS CONCEPTS

• **Information systems → Geographic information systems**.

KEYWORDS

Junction, Monitoring, Trajectory, Travel time, Energy consumption

ACM Reference Format:
Rodrigo Sasse David, Kristian Torp, Mahmoud Sakr, and Esteban Zimányi. 2024. A Framework for Automated Junction Monitoring. In *The 32nd ACM International Conference on Advances in Geographic Information Systems (SIGSPATIAL '24), October 29-November 1, 2024, Atlanta, GA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3678717.3691322

1 INTRODUCTION

Monitoring and evaluating junctions are among the most challenging problems in transportation [19]. Initially, junctions were monitored by humans manually counting the number of vehicles traveling through [5]. Recently, new technology has allowed junctions to be monitored with loop detectors, cameras, or wireless devices [8]. Such devices automate the task but with major drawbacks, such as low coverage and high cost [10]. Another issue is identifying which junctions to monitor, e.g., just in the US, there are over 300,000 traffic signals and 10,000 roundabouts [20].

In this work, we present a framework that applies to both roundabouts and signalized intersections. It features an algorithm for the nationwide automatic identification of junctions within a road network, along with all necessary components for evaluating and comparing them in terms of travel time and energy consumption using both trajectory and segment data. Most importantly, it allows junctions to be monitored so the ones that require attention can be easily identified.

Figure 1 shows two real-world junctions in a road network. The blue and red circles mark where we start and end measuring relevant metrics.

(a) Roundabout (b) Signalized intersection

Figure 1: Two junctions in a road network

Figure 2 shows the travel time, energy consumption, and trajectory count (in parenthesis) for them. Blue, orange, and green text indicate left, straight, and right turns, respectively.

	N	E	S	W
N	-	16.4 s 34.0 Wh (63)	11.6 s 15.4 Wh (15)	5.0 s 18.3 Wh (39)
E	4.3 s -1.8 Wh (56)	-	18.2 s 11.4 Wh (4)	11.2 s 7.5 Wh (97)
S	9.9 s 14.9 Wh (18)	4.3 s 9.1 Wh (13)	-	19.5 s 30.2 Wh (10)
W	16.3 s 30.4 Wh (48)	10.6 s 20.3 Wh (94)	5.3 s 16.3 Wh (8)	-

(a) Roundabout in Fig. 1 (a)

	N	E	S	W
N	-	8.5 s 33.9 Wh (6)	5.3 s 27.4 Wh (51)	3.8 s 4.7 Wh (7)
E	12.3 s -8.2 Wh (5)	-	16.4 s 52.7 Wh (12)	10.1 s 14.2 Wh (38)
S	7.0 s 8.2 Wh (47)	2.6 s 21.5 Wh (10)	-	10.9 s 8.7 Wh (26)
W	17.4 s 22.0 Wh (8)	10.1 s -1.2 Wh (38)	8.3 s 0.3 Wh (31)	-

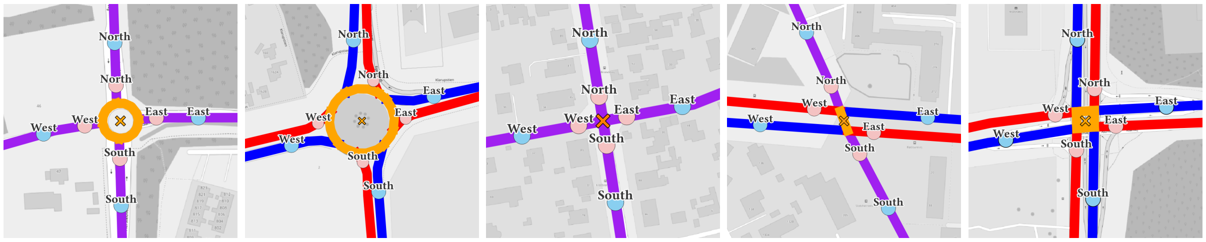
(b) Intersection in Fig. 1 (b)

Figure 2: Travel time, energy consumption, and traj. count

The remainder of the paper is organized as follows. Section 2 presents the related work. Sections 3 to 6 introduce the elements of the framework. Section 7 presents the data used in the experiments. Section 8 discusses the experiments. Finally, Section 9 concludes the paper.

SIGSPATIAL '24, October 29-November 1, 2024, Atlanta, GA, USA

ACM ISBN 979-8-4007-1107-7/24/10
https://doi.org/10.1145/3678717.3691322



(a) Bid. roundabout **(b) Uni. roundabout** **(c) Point intersection** **(d) Line intersection** **(e) Polygon intersection**

Figure 3: Different types of roundabouts and signalized intersections in a road network

2 RELATED WORK

Shirazi and Morris [16] conducted a comprehensive survey on aspects of intersection monitoring. They examined the different behaviors and safety assessments for various participants, including vehicles, pedestrians, and drivers, highlighting the unique challenges and considerations for each group.

Ziemke and Braun [23] addressed the challenging task of identifying signalized intersections in OpenStreetMap. While their method effectively achieved its intended purpose, it treats all intersections uniformly by simplifying them to a single-node representation. This approach fails to capture the intrinsic differences among various intersections.

Nematichari et al. [12] presented a system capable of continuously evaluating and forecasting the operational performance of road intersections. Their method treats all intersection types the same way, without any differentiation, and their evaluation considers intersections as a whole, without providing evaluations at the turn level.

Avşar and Avşar [1] and Dinh and Tang [5] proposed systems for camera-based automated traffic data collection for roundabouts. Both works focused on tracking the flow of vehicles at the turn level. Although both systems demonstrated accuracy of over 90%, camera-based systems are often regarded as costly and time-consuming.

Borresen et al. [2] presented a system that uses GPS-based trajectory data to calculate travel time matrices for signalized intersections. Wang et al. [17] used similarity measures and clustering techniques for analyzing trajectories at intersections. While trajectory-based systems are cheaper than camera-based systems, both methods require prior knowledge of the intersections, which limits their applicability for large-scale nationwide monitoring.

Zhou et al. [21] conducted a comparative analysis between roundabouts and signalized intersections. They developed simulation models for both, covering scenarios with varying traffic volumes. Taglieri et al. [11] investigated the network-wide conversion of signalized intersections into roundabouts with respect to free-flow speed and fuel consumption rate. The drawback of both studies is that they relied on simulation data instead of real-world data.

Demir and Demir [4] studied the applicability of modern roundabout design to a high-volume signalized intersection. The study showed an increase in capacity and a decrease in average delay and emissions. Although the study used real-world data, it was limited to only one intersection.

3 JUNCTIONS

In this section, we discuss the road network, the different types of junctions, and how we model them. This work focus on two types of at-grade junctions: roundabouts and signalized intersections.

3.1 Road Network

A road network is a directed graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, with $\mathcal{V}$ as the set of vertices, $\mathcal{E}$ as the set of edges, and $\mathcal{E} \subseteq \{(x, y) | (x, y) \in \mathcal{V}^2$ and $x \neq y\}$. The edges represent road segments and the vertices, intersections. In this work, we use OpenStreetMap (OSM)[1].

A road segment s is the tuple $(sid, len, dir, way, geo)$, in which rid is the ID of the segment, len, the length of the segment, dir, the direction of movement in the segment, way, whether the segment is bidirectional or unidirectional, and geo, the sequence of points that represents it. From now on, we call them segments for simplicity.

3.2 Roundabouts

A roundabout is represented by a set of segments. All roundabouts have the same layout regarding their center but differ in their incoming and outgoing segments. In a bidirectional roundabout, all segments are bidirectional, allowing traffic to flow in both directions. In contrast, in a unidirectional roundabout, all segments are unidirectional, permitting traffic to flow in only one direction. Figures 3 (a) and (b) show an example of a bidirectional roundabout and a unidirectional roundabout, respectively.

3.3 Signalized Intersections

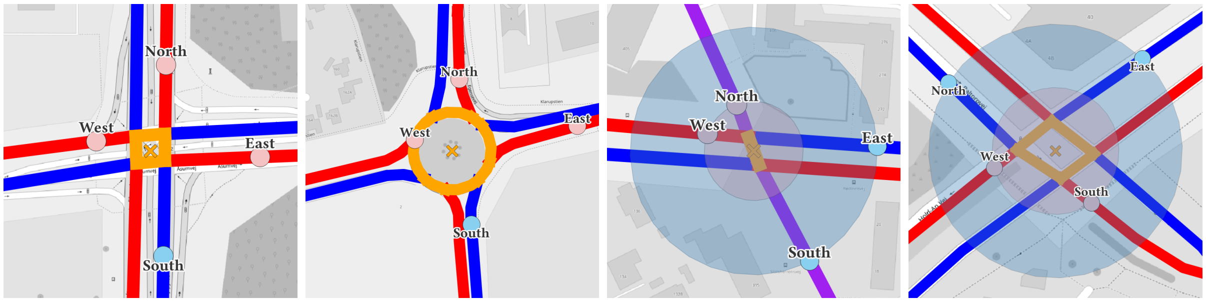
A signalized intersection is also represented by a set of segments. However, while the center of roundabouts always has the same shape, the center of signalized intersections can be either a single node (point), a single segment (line), or a set of segments (polygon).

3.3.1 Point. In this type, the incoming and outgoing segments connect through a node and there are no segments between them. The node acts as the center of the intersection. This type is accessed mostly by bidirectional segments. Figure 3 (c) shows an example.

3.3.2 Line. In this type, the incoming and outgoing segments connect through one common segment. This segment acts as the center of the intersection. This type is accessed by both unidirectional and bidirectional segments. Figure 3 (d) shows an example.

---

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



(a) Fixed distance model **(b) Fixed distance model** **(c) Distance from centroid model** **(d) Distance from centroid model**

Figure 4: In-out points placed by different placement models

3.3.3 Polygon. In this type, the incoming and outgoing segments connect through a set of segments forming a rectangle. Together, these segments act as the center of the intersection. This type is accessed almost exclusively by unidirectional segments. Figure 3 (e) shows an example.

3.4 Modeling

We use two distinct data structures to model junctions: a junction and a set of in-out pairs. This allows us to represent any junction consistently using these data structures.

A junction j represents a single junction. A junction j is the tuple $(jid, tp, cen, latc, lonc)$, in which jid is the ID of the junction, tp, the type of the junction, cen, the set of center segments, and $latc$ and $lonc$, the latitude and longitude coordinates of the centroid of the junction, respectively.

An in-out pair p represents a combination of an incoming and an outgoing segment. An incoming segment is used to enter a junction, while an outgoing, to exit it. An in-out pair p is the tuple (jid, pid, i, c, o, d), in which jid is the ID of the junction the pair belongs to, pid, the ID of the pair, i, the incoming segment, c, the shortest sequence of center segments that connects the incoming and outgoing segments, o, the outgoing segment, and d, the direction of movement for the incoming, outgoing, and center segments.

Consider Figure 3 for an example. The blue lines are unidirectional incoming, the red lines are unidirectional outgoing, and the purple lines are bidirectional incoming and outgoing segments. The orange lines are center segments. The orange X marks are centroids. The light blue and red points are called in-out points. They are presented next.

4 JUNCTION CONFIGURATION

For effective junction analysis, it is crucial to capture activity at their centers and in their surroundings. When comparing them, it is essential to ensure conditions are similar for accurate and meaningful assessments. To address both requirements, we use in-out points.

The in-out points are positioned before and after the center of junctions, along their incoming and outgoing segments, respectively. They serve as markers, guiding where to measure relevant metrics. By utilizing these points, we accurately capture activity at the junction while ensuring a fair comparison between them.

The in-out points are determined for each in-out pair of a junction. These points are fully visible in Figures 1 and 3 and partially visible in Figure 4 (in points are in light blue and out points are in light red).

4.1 Placement Models

The placement of the in-out points is flexible and determined before any analysis. In this subsection, we discuss different models we envision for determining how to place the in-out points.

4.1.1 Equal distance. This model considers the same travel distance for each in-out pair. First, we determine the position of the in points. They are placed along each incoming segment a distance d_{in} from the center of the junction. Then, from each point, we measure a fixed distance d towards each outgoing segment. The out points are placed at the end of d.

The advantage of this model is that it considers the same travel distance for all in-out pairs. The disadvantage is that, except for point intersections, the final position of the out points inevitably varies. If d is too long, some out points are placed too far from the center. If d is too short, some out points are placed too close or even at the center. Figures 4 (a) and (b) illustrate this problem. In both cases, d_{in} is 40 m and d is 80 m. Notice how in (a), out point east is placed farther from the center than out point north, which is farther than out point west. In (b), the same happens for out points east and north, while out point west is placed at the center.

4.1.2 Distance from the centroid. This model uses the centroid of the junctions to determine the placement of the in-out points. From the centroid, we draw two circles of radius d_{in} and d_{out}. The in and out points are placed where the circles intercept the incoming and outgoing segments, respectively.

The advantage of this model is that it uses a common element to provide an easy and uniform way to determine the position of the in-out points across all types of junctions. The disadvantages are twofold. First, line and polygon intersections are not perfectly shaped, meaning that the incoming and outgoing segments are not equidistant from the centroid. This results in varying distances among the in-out points to the center of the junction. Second, the same configuration can place the in-out points at different distances from the center among different types of junctions, impacting how much activity each type captures. Figures 4 (c) and (d) illustrate these problems. In both cases, d_{in} is 40 m and d_{out} is 20 m. In (c), in points south and east and out points north and west are placed 35.9, 39.6, 13.9, and 19.1 m from the center segments, respectively. In (d), in points east and north and out points west and south are placed 34.6, 30.1, 9.1, and 8.8 m from the center segments, respectively.

4.1.3 *Distance from center.*

This model places the in-out points along the incoming and outgoing segments at a distance d_{in} and d_{out} from the node where the segments meet the center of the junction.

The advantage of this model is that it ensures all in-out points are equidistant from the center of the junctions, resulting in each pair considering the same distance both before and after each junction. However, a drawback is that determining the final position of the in-out points with this model is more complex compared to the other two models.

Given that this model considers the same distance before and after the junctions for all in-out pairs and all types of junctions, it is the one we select to be used in the framework. Each combination of d_{in} and d_{out} is called a configuration. For example, in Figure 3, all in-out points are placed using this model. In all figures, d_{in} and d_{out} are 30 and 10 meters, respectively, or a 30-10 configuration.

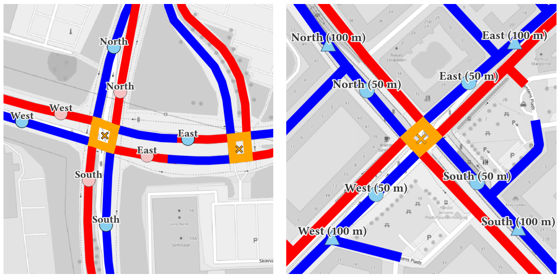
4.2 Potential Disturbances

While the in-out points add flexibility to the framework, one must be cautious when selecting configurations. As the distances d_{in} or d_{out} increase, the farther the points are placed from the center of the junction of interest. This can cause two problems.

First, if the points are placed too far from the center of the junction of interest, they might end up being placed too close to (or even at the center of) another junction. This can impact the results of the analysis as those points can capture activity from the other junction. Figure 5 (a) illustrates this problem. Notice how in point east is closer to another intersection (on the right) rather than to the intersection of interest (on the left).

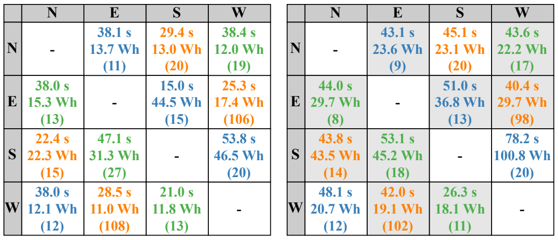
Second, for the trajectory of a vehicle to be taken into account in any analysis, the vehicle has to pass by both the in and out points. If the points are placed too far from the center, there can be other segments between the center and either of the points. In this way, there can be some trajectories that pass by only one of them. Figure 5 (b) illustrates this problem. There are two sets of in points. For the closest points, there are no segments between them and the center. For the farthest points, there is one segment between each of them and the center. This is of particular importance because, when comparing a junction with different configurations, the metrics each one yields can change not just because the travel distance is different, but also because each configuration takes into account a different set of trajectories. Figure 6 illustrates this problem. It shows the metrics for the intersection in Figure 5 (b). For Figure 6 (a), we use a 50-10 configuration, and for Figure 6 (b), a 100-10 configuration. Notice how the number of trajectories is lower in Figure 6 (b) for 9 of the 12 pairs (highlighted with gray background color). Although we use polygon intersections to illustrate both problems, all junctions are subject to them, regardless of their type.

Whenever a configuration causes a disturbance at a junction, that configuration is deemed invalid for the junction, and the junction is excluded from any further analysis.



(a) In point too far (b) Segments between in points

Figure 5: Potential disturbances



	N	E	S	W
N	-	38.1 s 13.7 Wh (11)	29.4 s 13.0 Wh (20)	38.4 s 12.0 Wh (19)
E	38.0 s 15.3 Wh (13)	-	15.0 s 44.5 Wh (15)	25.3 s 17.4 Wh (106)
S	22.4 s 22.3 Wh (15)	47.1 s 31.3 Wh (27)	-	53.8 s 46.5 Wh (20)
W	38.0 s 12.1 Wh (12)	28.5 s 11.0 Wh (108)	21.0 s 11.8 Wh (13)	-

(a) 50-10 config.

	N	E	S	W
N	-	43.1 s 23.6 Wh (9)	45.1 s 23.1 Wh (20)	43.6 s 22.2 Wh (17)
E	44.0 s 29.7 Wh (8)	-	51.0 s 36.8 Wh (13)	40.4 s 29.7 Wh (98)
S	43.8 s 43.5 Wh (14)	53.1 s 45.2 Wh (18)	-	78.2 s 100.8 Wh (20)
W	48.1 s 20.7 Wh (12)	42.0 s 19.1 Wh (102)	26.3 s 18.1 Wh (11)	-

(b) 100-10 config.

Figure 6: Metrics for the intersection in Figure 5 (b)

5 DATA FOUNDATION

In this section, we present and discuss the two datasets we use in the experiments - trajectory data and segment data.

5.1 Trajectory Data

A trajectory is a sequence of GPS points ordered by time. Each point in our dataset is the tuple (tid, lat, lon, ts, ec), in which tid is the ID of the trajectory the point belongs to, lat, the latitude coordinate, lon, the longitude coordinate, ts, a timestamp, and ec, the instantaneous energy consumption in watt-hour. Table 1 presents an example of such dataset.

tid	lat	lon	ts	ec
1	55.401815	10.425122	2012-05-27 13:25:34	2.855
1	55.401824	10.425352	2012-05-27 13:25:35	3.467
1	55.401832	10.425517	2012-05-27 13:25:36	3.763

Table 1: Example of trajectory data

The main advantage of this dataset is that it offers precise information, especially if the GPS receiver's sampling frequency is high. Also, the trajectory count can be used to identify disturbances in a configuration (see Figure 6). However, there are two main drawbacks. First, if the sampling frequency is too low, there can be just a few GPS points between the in and out points, which may diminish the confidence of the analysis. Second, it only accounts for segments that have been traveled by the vehicles, which may limit the coverage of the analysis.

To compute travel time and energy consumption with this dataset, we first place the in-out points according to the desired configuration. For each in-out pair, we identify all trajectories that travel through its correspondent in-out points. Finally, for each trajectory, we identify the GPS points between the in-out points. Consider Figure 7 for an example. It shows the trajectory of a vehicle as it travels through a signalized intersection (from west to east).

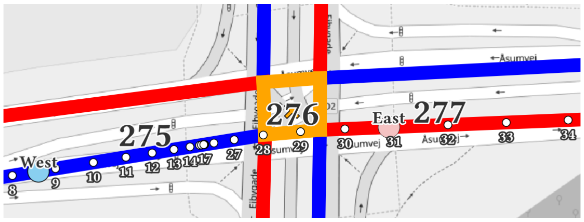


Figure 7: The GPS points of a trajectory

In the figure, there are 22 GPS points between in point west and out point east — the sequence $\langle g_9, g_{10}, ..., g_{30}\rangle$. For the travel time, we subtract the timestamp of the first point in the sequence, g_9, from the timestamp of the last point, g_{30}. For the total energy consumption, we sum the instantaneous consumption of all points in the sequence. We do this for all trajectories and then compute the average between them for each metric.

5.2 Segment Data

With segment data, the travel time and energy consumption from multiple trajectories are aggregated and become attributes of segments, commonly seen as weights [9]. Each weight in our dataset is the tuple (sid, len, dir, tt, ec), in which sid is the ID or the segment, len, its length in meters, dir, the direction of movement, tt, the travel time in seconds, and ec, the energy consumption in watt-hour. The weights we use do not have any information about time of day. Table 2 presents an example of such dataset.

sid	len	dir	tt	ec
275	152.96	forward	21.3	24.6
276	13.67	forward	1.9	5.4
277	126.31	forward	15.9	19.8

Table 2: Example of segment data

The main advantages of this dataset are that it aggregates data from multiple trajectories, mitigating the impact of the GPS receiver's sampling frequency, and it can offer coverage for the whole network. The main drawback is that there is no trajectory count, so it is not possible to identify potential disturbances.

To compute the travel time and energy consumption of a junction with this dataset, we first place the in-out points according to the desired configuration. For each in-out pair, we identify the segments involved. Consider Figure 7 for an example. In the figure, there are three segments involved — 275, 276, and 277. For each segment, we get its corresponding weight. For the incoming and outgoing segments, 275 and 277, we interpolate the metrics to account for the configuration. For the center segment, 276, we consider the metrics in full. Equation 1 shows how we calculate the total travel time for each in-out pair, in which tt_{in}, d_{in}, and len_{in} are the travel time in the incoming segment, distance of the in point, and length of the incoming segment, respectively. The same applies to tt_{out}, d_{out}, and len_{out}. Finally, tt_{cen} is the average travel time of the center segment. To calculate the total energy consumption, we replace tt with ec for each segment involved.

$$T = \frac{tt_{in} \cdot d_{in}}{len_{in}} + tt_{cen} + \frac{tt_{out} \cdot d_{out}}{len_{out}} \tag{1}$$

6 METRICS

We provide metrics at both the junction and the turn levels. At the junction level, there is one value for the entire junction. At the turn level, there is a value for each in-out pair (12 in total for a four-legged junction).

At the junction level, we calculate the weighted average of all its in-out pairs. The weight of each pair is the trajectory count of the pair divided by the trajectory count of the junction. Having a single value for each junction allows them to be compared or ranked, especially across different junction types, e.g., unidirectional roundabouts and polygon intersections [13, 15]. Consider the junction in Figure 2 (a) for an example. Each cell in the matrix is one in-out pair. The total trajectory count for the configuration used is 455. Equation 2 shows the weighted average travel time. We apply the same idea for energy consumption.

$$T = \frac{63 \cdot 16.4}{455} + \frac{15 \cdot 11.6}{455} + ... + \frac{94 \cdot 10.6}{455} + \frac{8 \cdot 5.3}{455} = 11.0\ s \tag{2}$$

For the turn level, we calculate the average as described in Section 5. When evaluating and comparing junctions, the primary reason to focus on turns rather than entire junctions is that turns are intrinsically different. These differences are not only present among the same junction type but also vary between different junction types. This approach provides a broader perspective on the advantages and disadvantages of each junction type [6, 7, 10].

For the experiments in Section 8, we exclusively use four-legged junctions. First, four-legged junctions are the most common type [19], which maximizes the number of junctions included in the study. Second, using these junctions ensures that all have similar layouts, providing consistency across our experiments. While our study focuses on these specific junctions, the framework is versatile and can be applied to junctions with any number of legs.

In a four-legged junction, there are three possible turns (U-turns are not considered). Consider Figure 1 for an example. If a vehicle coming from the south goes east, it makes a right turn. If it goes north, it goes straight. If it goes west, it makes a left turn. One can see going right, straight, or left as taking the 1st, 2nd, or 3rd possible exits, counting it counterclockwise. We have five different junction types, each having three different turn types, for a total of 15 junction-turn types:

$$\left\{\begin{array}{c}\text{Bidirectional Roundabouts (BID)}\\ \text{Unidirectional Roundabouts (UNI)}\\ \text{Point Intersections (POI)}\\ \text{Line Intersections (LIN)}\\ \text{Polygon Intersections (POL)}\end{array}\right\} \times \left\{\begin{array}{c}\text{Right Turn (R)}\\ \text{Straight Turn (S)}\\ \text{Left Turn (L)}\end{array}\right\}$$

7 JUNCTION IDENTIFICATION

In this section, we describe the algorithm for identifying junctions in a road network and present details on junctions and trajectory data.

7.1 Identification

To identify junctions in a road network, we first need to accurately identify their centers. As previously explained in Section 3, a center *cen* can be either a set of segments or a set of nodes.

In OSM, the center segments of roundabouts are tagged[2], so no particular identification is necessary. However, centers of signalized intersections are not. We use traffic signals to locate them. In OSM, a traffic signal t is a node with latitude and longitude coordinates[3]. So, for each signalized intersection type and each traffic signal, we search for:

- Point intersections: exactly one node that is common to three or more segments within a radius m from the traffic signal t that is not part of a line or polygon intersections.
- Line intersections: exactly one bidirectional segment completely within a radius n from the traffic signal t that is not part of a polygon intersection.
- Polygon intersections: exactly four unidirectional segments that together form a rectangle completely within a radius o from the traffic signal t.

Algorithm 1 shows a pseudo code that summarizes the steps for identifying polygon intersections. In the algorithm, the functions *contains* refers to PostGIS' ST_Contains[4] and *isRectangle* simply checks if the four segments form a rectangle. The algorithms for finding point and line intersections are quite similar to Algorithm 1 and they require minor modifications.

Algorithm 1

```
Input: set of traffic signals $T$, radius $o$, road network $\mathcal{G}$
Output: set of centers $C$
function $findPolygonSigInter(T, o, \mathcal{G})$
$C \leftarrow \{\}$
foreach $t$ in $T$ do
    $segs\_cen \leftarrow \{\}$
    foreach $s$ in $\mathcal{G}$ do
        if $contains(t, o, s)$ and $s.way$ is $unidirectional$ do
            add $s$ to $segs\_cen$
    if $|segs\_cen|$ is 4 and $isRectangle(segs\_cen)$ do:
        add $segs\_cen$ to $C$
return $C$
```

Once the centers are identified, for each center, we get all its incoming and outgoing segments. The incoming and outgoing segments are the segments that share nodes with the center. A bidirectional segment counts twice, both as an incoming and an outgoing segment. A unidirectional segment is incoming or outgoing if the first or last point of its sequence of points, *geo*, is at the center, respectively (Lines 1 to 12, Alg. 2).

With the incoming segments, outgoing segments, and centers, we build the junction j (Line 13, Alg. 2). Finally, for each combination of incoming and outgoing segments, we find the shortest sequence of center segments between them and build its in-out pair p (Lines 14 to 19, Alg. 2). Finding the shortest sequence of center segments only applies to roundabouts and polygon intersections. For point intersections, the sequence c is empty, and for line intersections, it is always the center road.

Algorithm 2 shows the pseudo code that summarizes the overall steps previously described.

Algorithm 2

```
Input: road network $\mathcal{G}$, junction center $cen$
Output: junction $j$, set of in-out pairs $P$
function $buildJunctionInOutPairs(\mathcal{G}, cen)$
1:  $in\_out\_segs \leftarrow getInOutSegs(\mathcal{G}, cen)$
2:  $in\_segs \leftarrow \{\}$
3:  $out\_segs \leftarrow \{\}$
4:  foreach $io$ in $in\_out\_segs$ do
5:      if $io.way$ is $bidirectional$ do
6:          add $io$ to $in\_segs$
7:          add $io$ to $out\_segs$
8:      else do
9:          if $io.geo[-1]$ in $cen$ do
10:             add $io$ to $in\_segs$
11:         else do
12:             add $io$ to $out\_segs$
13: $j \leftarrow buildJunction(in\_segs, out\_segs, cen)$
14: $P \leftarrow \{\}$
15: foreach $i$ in $in\_segs$ do
16:     foreach $o$ in $out\_segs$ do
17:         $c \leftarrow findShortestRoute(i, o, cen)$
18:         $p \leftarrow buildInOutPair(j.jid, i, o, c)$
19:     add $p$ to $P$
20: return $j, P$
```

7.2 Junction and Trajectory Details

We use the road network of Denmark to identify junctions. It consists of 749,371 edges and 1,068,605 nodes, and it has a total of 9,998 traffic signals. Identifying intersections requires different search radii, each determined empirically. These radii must be long enough to encompass the centers, but not so long that they include unintended segments or nodes. For the used road network, the radii m, n, and o that yield the most number of point, line, and polygon intersections are 8, 13, and 36 meters, respectively.

The identification is executed on a MacBook Pro equipped with a M1 Pro CPU and 16 GB of RAM. The execution times are 279 and 237 seconds for bidirectional and unidirectional roundabouts, respectively, and 614, 567, and 527 seconds for point, line, and polygon intersections, respectively.

---

[2] https://wiki.openstreetmap.org/wiki/Roundabout

[3] https://wiki.openstreetmap.org/wiki/Tag:highway%3Dtraffic_signals

[4] https://postgis.net/docs/ST_Contains.html

As previously explained, we exclusively use four-legged junctions. Table 3 presents the total number of junctions, the number of four-legged junctions, and their corresponding percentage. Figure 8 shows all the junctions found in the road network.

Junction Type	All	Four-legged	%
Bid. Roundabout	1,293	656	50.7
Uni. Roundabout	199	105	52.8
Poi. Intersection	1,098	369	33.6
Lin. Intersection	921	176	19.1
Pol. Intersection	214	88	41.1
Total	3,725	1,394	37.4

Table 3: Junction data details

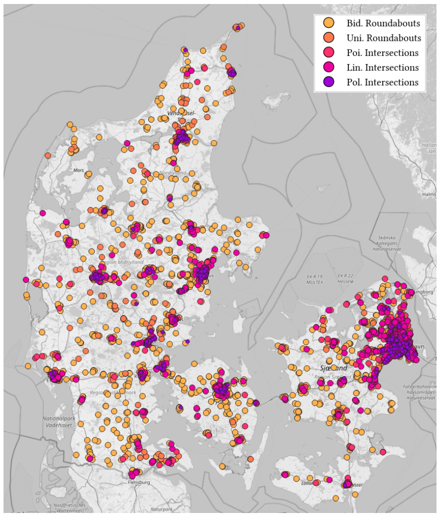


Figure 8: Junctions in Denmark

The trajectory data comes from a fleet of 177 battery electric vehicles (BEVs) equipped with both GPS and OBD devices, each sampling at a frequency of 1 Hz. The data is collected for 29 months across all regions of Denmark. The trajectory and segment data used in the experiments derive from the same dataset. Table 4 presents, for four-legged junctions, the total number of trajectories that travel through each junction type, the distance they travel at junctions in kilometers, and their count of GPS points.

Junction Type	Traj.	Dist.	GPS Points
Bid. Roundabout	84,923	44,370	3,309,561
Uni. Roundabout	12,863	7,421	501,195
Poi. Intersection	127,845	38,220	4,542,654
Lin. Intersection	64,316	30,026	2,880,159
Pol. Intersection	44,134	21,528	2,374,247
Total	334,081	141,565	13,607,816

Table 4: Trajectory data details

Although we use data from Denmark, it is important to note that the framework can be applied to any country or region mapped by OSM, provided that trajectory or segment data is available.

8 EXPERIMENTS

In this section, we compare how the two datasets perform, compare junctions regarding travel time and energy consumption, and demonstrate how the framework can be used to monitor and identify junctions that require immediate attention.

Although we use data from electric vehicles for the experiments, the framework it is compatible with data from any type of vehicle, provided it adheres to the datasets described in Section 5.

8.1 Trajectory and Segment Data

We begin by comparing the two datasets described in Section 5. We seek to determine whether they yield similar results. We start with a 30-10 configuration. This configuration encompasses 297,631 of the 334,081 trajectories in our dataset (89.1%). Figures 9 and 10 show the average travel time and energy consumption per junction and turn types for each dataset, respectively. The different background colors are used to differentiate the five junction types.

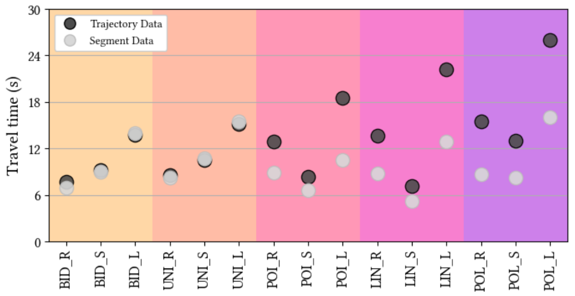


Figure 9: Travel time per turn per dataset

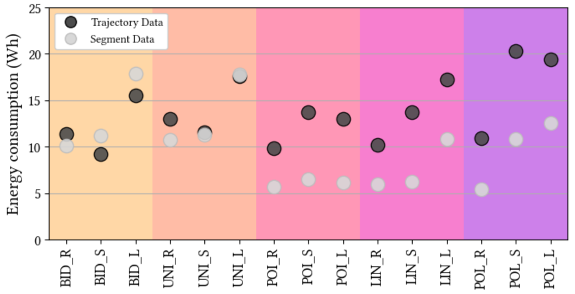


Figure 10: Energy consumption per turn per dataset

For roundabouts, both datasets exhibit similar values. However, for signalized intersections, segment data consistently shows lower values across all turns. The overall difference can be explained by the fact that vehicles stop and wait less frequently and for shorter durations at roundabouts compared to signalized intersections. With trajectory data, these moments are captured more accurately. With segment data, they can only be approximated. Therefore, both dataset yield similar results when waiting times are shorter and less frequent, while segment data tends to underestimate both travel time and energy consumption when they are longer and more common.

Another interesting finding is that, for signalized intersections, the difference between trajectory and segment data is more evident for right and left turns than for straight turns. We investigate that by looking at only the incoming segments. Table 5 presents, for each junction and turn types, the trajectory count and percentage, average travel time in seconds, and energy consumption in watt-hour for their incoming segments (considering 30 m before the center, similar to a 30-10 configuration).

Turn	Count	Perc.	Time (s)	Cons. (Wh)
BID_R	17,673	21.4%	3.0	11.4
BID_S	46,567	56.5%	2.6	10.0
BID_L	18,173	22.1%	2.7	11.9
UNI_R	2,593	26.3%	2.9	11.1
UNI_S	4,732	48.0%	2.9	9.3
UNI_L	2,527	25.7%	2.8	11.5
POI_R	19,760	17.6%	7.1	11.2
POI_S	71,429	63.6%	5.4	7.9
POI_L	21,127	18.8%	8.5	9.9
LIN_R	10,314	18.7%	7.0	11.7
LIN_S	35,926	65.2%	3.9	8.3
LIN_L	8,870	16.1%	8.5	12.0
POL_R	7,042	18.6%	7.1	12.8
POL_S	24,706	65.1%	5.9	9.4
POL_L	6,192	16.3%	10.5	12.0

Table 5: Incoming segment data

For roundabouts, there is little difference among turns. This is because vehicles behave the same before entering roundabouts regardless of the turns they intend to make [3]. For signalized intersections, the table shows vehicles can take 20-79% and 36-118% more time and use 26-148% and 21-155% more energy to travel through incoming segments when taking right or left turns compared to when going straight, respectively. The table also shows that 48-65.2% of the vehicles go straight, while 17.6-26.3% and 16.1-25.7% make right or left turns, respectively. This explains the pattern with segment data. As going straight is faster and more common than making right or left turns, when trajectory data is aggregated into segment data, it brings down the average for the whole incoming segment.

Figures 9 and 10 and Table 5, together, point to the necessity for traditional real-time path planning algorithms to add junctions to their context and consider the distinctions among different turns and between different junction types, specially signalized intersections, for more accurate estimations [14, 18, 22].

For the upcoming experiments, we will exclusively utilize trajectory data due to its capacity for more precise analysis. Additionally, we will employ the trajectory count as detailed in Subsection 8.4.

8.2 Travel Time

We now conduct an in-depth investigation regarding travel time. We begin with the same 30-10 configuration as before. Figure 11 shows the travel time per junction and turn types.

While the median travel times for roundabouts and signalized intersections are similar among the same turn types, roundabouts exhibit much lower variance compared to signalized intersections. This emphasizes the distinction between roundabouts and signalized intersections regarding traffic flow. Roundabouts allow vehicles to traverse with minimal disruption more frequently, whereas at signalized intersections, this occurs less frequently due to vehicles stopping at red lights at various times.

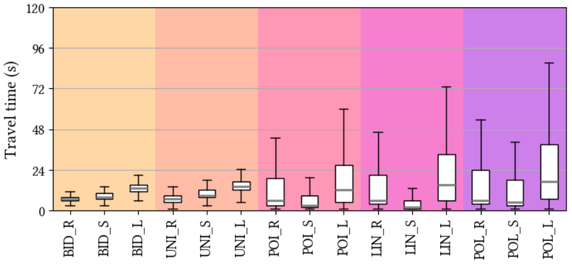


Figure 11: Travel time per turn (30-10 config.)

For roundabouts, making a right turn is faster than going straight, which is faster than making a left turn. This is anticipated as it aligns with the increase in distance traveled — vehicles cover approximately 25%, 50%, or 75% of the four-legged roundabouts for right, straight, and left turns, respectively. However, at signalized intersections, the pattern differs. While taking a left turn consumes more time, as expected, going straight is quicker than making a right turn. This highlights the distinct characteristics in design and traffic behavior between the two groups.

Now, we repeat the analysis, but with a 90-10 configuration. We do this to assess the impact of placing the in points farther away from the junctions. This configuration takes into account 276.814 of the 334.081 trajectories in our dataset (82.9%). Figure 12 shows the travel time per junction and turn types with said configuration.

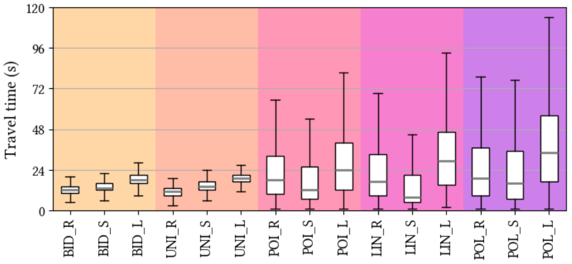


Figure 12: Travel time per turn (90-10 config.)

The figure shows an increase in the travel time for all junctions and turns, which is expected given the additional 60 meters. However, while the extra distance contributes to higher medians and variance across both roundabouts and intersections, roundabouts show smaller increases compared to intersections.

This difference can be attributed to capturing more stop-and-wait moments by placing the in points from further from the center of the junctions. Given that vehicles stop and wait more frequently and for longer periods at signalized intersections than at roundabouts, it is reasonable that placing the in points farther is more impactful for intersections than for roundabouts.

8.3 Energy Consumption

We now conduct an in-depth investigation regarding energy consumption of the junctions We begin with the same 30-10 configuration as before. Figure 13 shows the energy consumption per junction and turn.

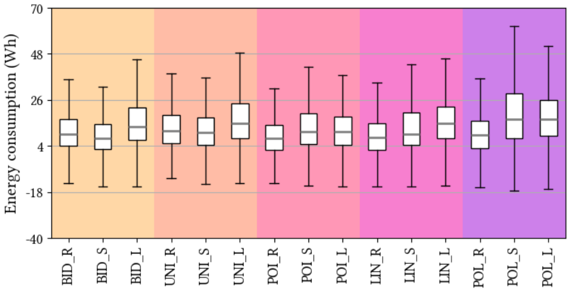


Figure 13: Energy consumption per turn (30-10 config.)

For energy consumption, the distinction between roundabouts and signalized intersections becomes less evident. Both the medians and variance are similar. There are two main causes for this.

First, electric motors do not consume energy when they are not in use, so there is no energy consumption from the motor when a BEV stop and wait. Therefore, unlike travel time, energy consumption relates to the distance traveled. Second, EVs can recover energy when braking. While vehicles brake less frequently at roundabouts, they also accelerate less. Conversely, at signalized intersections, vehicles brake more often but also accelerate more. This balance between negative consumption from braking and positive consumption from acceleration equalizes the energy consumption across different types of junctions.

Similar to our approach with the travel time, we repeat the analysis using a 90-10 configuration. Figure 14 shows the energy consumption per junction and turn types with said configuration.

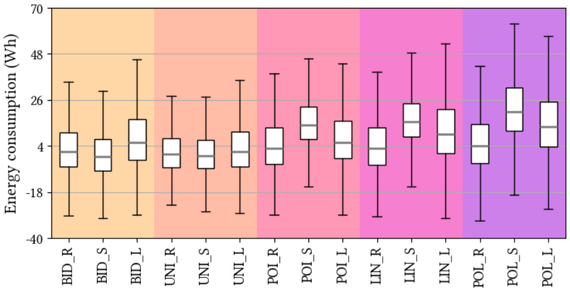


Figure 14: Energy consumption per turn (90-10 config.)

As opposed to the travel time, farther placing the in points from the junctions decreases the energy consumption for most turns. This is because the 90-10 configuration captures much more traffic before the junctions than after them. Hence, such a configuration captures more braking than acceleration. However, there are exceptions. Going straight at signalized intersections now shows an overall increase in median consumption. This is because, for signalized intersections, vehicles brake less when going straight compared to making right or left turns. For roundabouts, vehicles must brake before entering regardless of the turn they make.

8.4 Monitoring and Identification

Finally, we demonstrate how the framework can be used to monitor and identify junctions that may require further investigation or intervention. We use a 30-10 configuration. With such configuration, 16,068 turns from 1,339 junctions are taken into account - 628 bidirectional roundabouts (BID), 99 unidirectional roundabouts (UNI), 362 point intersections (POI), 169 line intersections (LIN), and 81 polygon intersections (POL).

Figure 15 shows the weighted average travel time per trajectory count per junction type. Figure 16 shows the average travel time per trajectory count per turn type. Right, straight, and left turns are represented by right (►), up (▲), and left (◄) arrows, respectively. Although not shown, the same idea can be applied to energy consumption.

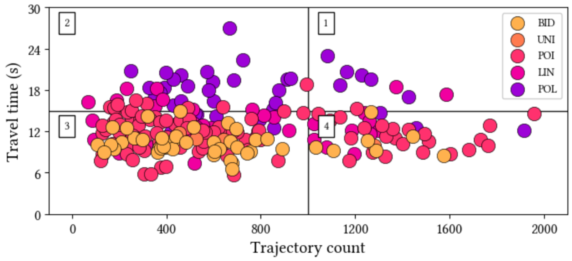


Figure 15: Travel time per trajectory count per junction

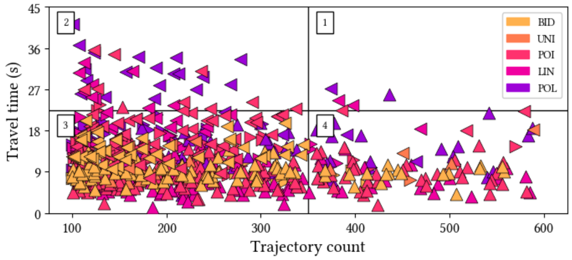


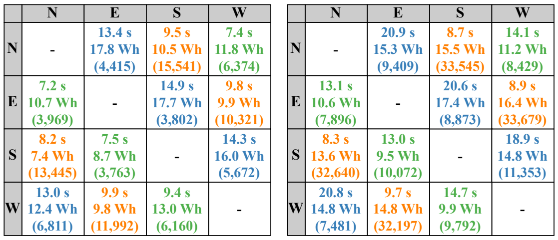
Figure 16: Travel time per trajectory count per turn

In the figures, Quadrant 4 contains junctions/turns with low travel time and high trajectory count, indicative of junctions/turns that require little to no attention. Quadrant 3 contains junctions/turns with low travel time and low trajectory count, also indicative of junctions/turns that require little attention, but may warrant occasional monitoring to ensure no emerging issues. Quadrant 2 includes junctions/turns with high travel time and low trajectory count, signaling junctions/turns that might need further investigation. Finally, Quadrant 1 contains junctions/turns with high travel time and high trajectory count, highlighting critical junctions/turns that demand immediate attention or intervention.

With such figures, one can easily identify junctions or turns of interest. The axes that form the quadrants can be adjusted according to different needs and the trajectory count can be used as a filter.

Figures 15 and 16 show a clear difference between polygon intersections and left turns, respectively. Once again, they underscore the importance of classifying roundabouts and signalized intersections accurately and diving into the turn level to better understand and address their unique traffic patterns.

Another effective method for monitoring junctions is the use of travel time matrices. These matrices provide a comprehensive overview of all possible turns within a single junction and allow for side-by-side comparisons between different junctions or groups of junctions. Figure 17 show the travel time, energy consumption, and trajectory count for all roundabouts and all intersections.



	N	E	S	W
N	-	13.4 s 17.8 Wh (4,415)	9.5 s 10.5 Wh (15,541)	7.4 s 11.8 Wh (6,374)
E	7.2 s 10.7 Wh (3,969)	-	14.9 s 17.7 Wh (3,802)	9.8 s 9.9 Wh (10,321)
S	8.2 s 7.4 Wh (13,445)	7.5 s 8.7 Wh (3,763)	-	14.3 s 16.0 Wh (5,672)
W	13.0 s 12.4 Wh (6,811)	9.9 s 9.8 Wh (11,992)	9.4 s 13.0 Wh (6,160)	-

(a) All roundabouts

	N	E	S	W
N	-	20.9 s 15.3 Wh (9,409)	8.7 s 15.5 Wh (33,545)	14.1 s 11.2 Wh (8,429)
E	13.1 s 10.6 Wh (7,896)	-	20.6 s 17.4 Wh (8,873)	8.9 s 16.4 Wh (33,679)
S	8.3 s 13.6 Wh (32,640)	13.0 s 9.5 Wh (10,072)	-	18.9 s 14.8 Wh (11,353)
W	20.8 s 14.8 Wh (7,481)	9.7 s 14.8 Wh (32,197)	14.7 s 9.9 Wh (9,792)	-

(b) All intersections

Figure 17: Metrics for roundabouts and sig. intersections

9 CONCLUSION

This paper introduces a comprehensive framework for the automated large-scale monitoring of junctions - roundabouts and signalized intersections. The framework encompasses an algorithm designed for the nationwide identification of junctions within a road network, two types of roundabouts and three types of signalized intersections, a never-before-seen approach. Additionally, it includes all necessary components to evaluate, compare, and monitor these junctions effectively.

Through a series of experiments, we analyze 1,394 junctions using 334,081 trajectories. We compare the framework's performance with detailed trajectory and high-level segment data, analyze how different junction and turn types impact travel time and energy consumption, and demonstrate how the framework can be leveraged to identify junctions that require immediate attention for optimization or maintenance.

For future work, we aim to test the framework with data from additional countries, as well as data from internal combustion engine vehicles. Additionally, we plan to develop an interactive demo tailored for local authorities, enabling them to leverage the framework to enhance their decision-making processes.

ACKNOWLEDGMENTS

This work has received funding from the Horizon 2020 - Marie Skłodowska-Curie Innovative Training Networks (H2020-MSCA-ITN-2020) framework, grant agreement No. 955895.

REFERENCES

[1] Ercan Avşar and Yağmur Ö. Avşar. 2022. Moving vehicle detection and tracking at roundabouts using deep learning with trajectory union. *Multimedia Tools and Applications* 81, 5 (2022), 6653–6680.

[2] Johannes L. Borresen, Ove Andersen, Christian S. Jensen, and Kristian Torp. 2017. Interactive intersection analysis using trajectory data. In *Proceedings of the 25th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems.* 1–4.

[3] Rodrigo S. David, Esteban Zimányi, Kristian Torp, and Mahmoud Sakr. 2023. Roundabouts and the Energy Consumption of Electrical Vehicles. In *Proceedings of the 16th ACM SIGSPATIAL International Workshop on Computational Transportation Science.* 9–18.

[4] Hatice G. Demir and Yusuf K. Demir. 2020. A Comparison of traffic flow performance of roundabouts and signalized intersections: a case study in Nigde. *The Open Transportation Journal* 14, 1 (2020).

[5] Hai Dinh and Hua Tang. 2017. Development of a tracking-based system for automated traffic data collection for roundabouts. *Journal of Modern Transportation* 25 (2017), 12–23.

[6] Douglas W. Harwood, Karin M. Bauer, Ingrid B. Potts, Darren J. Torbic, Karen R. Richard, Emilia R. Rabbani, Ezra Hauer, Lily Elefteriadou, and Michael S. Griffith. 2003. Safety effectiveness of intersection left-and right-turn lanes. *Transportation Research Record* 1840, 1 (2003), 131–139.

[7] Zhengbing He, Geqi Qi, Lili Lu, and Yanyan Chen. 2019. Network-wide identification of turn-level intersection congestion using only low-frequency probe vehicle data. *Transportation Research Part C: Emerging Technologies* 108 (2019), 320–339.

[8] Neeraj K. Jain, Rajesh K. Saini, and Preeti Mittal. 2019. A review on traffic monitoring system techniques. *Soft computing: Theories and applications: Proceedings of SoCTA 2017* (2019), 569–577.

[9] Tobias S. Jepsen, Christian S. Jensen, and Thomas D. Nielsen. 2019. Graph convolutional networks for road networks. In *Proceedings of the 27th ACM SIGSPATIAL international conference on advances in geographic information systems.* 460–463.

[10] Zihan Kan, Luliang Tang, Mei-Po Kwan, Chang Ren, Dong Liu, and Qingquan Li. 2019. Traffic congestion analysis at the turn level using Taxis' GPS trajectory data. *Computers, Environment and Urban Systems* 74 (2019), 229–243.

[11] Claudio Meneguzzer, Massimiliano Gastaldi, Riccardo Rossi, Gregorio Gecchele, and Maria V. Prati. 2017. Comparison of exhaust emissions at intersections under traffic signal versus roundabout control using an instrumented vehicle. *Transportation research procedia* 25 (2017), 1597–1609.

[12] Ali Nematichari, Tilemachos Pechlivanoglou, and Manos Papagelis. 2022. Evaluating and forecasting the operational performance of road intersections. In *Proceedings of the 30th International Conference on Advances in Geographic Information Systems.* 1–12.

[13] Irena I. Otković and Ivan Dadić. 2009. Comparison of delays at signal-controlled intersection and roundabout. *Promet-Traffic&Transportation* 21, 3 (2009), 157–165.

[14] Younes Regragui and Najem Moussa. 2023. A real-time path planning for reducing vehicles traveling time in cooperative-intelligent transportation systems. *Simulation Modelling Practice and Theory* 123 (2023), 102710.

[15] Katy Salamati, Nagui M. Rouphail, Henry C. Frey, Bin Liu, and Bastian J. Schroeder. 2015. Simplified method for comparing emissions in roundabouts and at signalized intersections. *Transportation Research Record* 2517, 1 (2015), 48–60.

[16] Mohammad S. Shirazi and Brendan T. Morris. 2016. Looking at intersections: a survey of intersection monitoring, behavior and safety analysis of recent studies. *IEEE Transactions on Intelligent Transportation Systems* 18, 1 (2016), 4–24.

[17] Chenxi Wang, Stefania Zourlidou, Jens Golze, and Monika Sester. 2021. Trajectory analysis at intersections for traffic rule identification. *Geo-spatial Information Science* 24, 1 (2021), 75–84.

[18] Longhao Yan, Ping Wang, Jingwen Yang, Yu Hu, Yu Han, and Junfeng Yao. 2021. Refined path planning for emergency rescue vehicles on congested urban arterial roads via reinforcement learning approach. *Journal of Advanced Transportation* 2021 (2021), 1–12.

[19] Bowen Zheng, Chung-Wei Lin, Shinichi Shiraishi, and Qi Zhu. 2019. Design and analysis of delay-tolerant intelligent intersection management. *ACM Transactions on Cyber-Physical Systems* 4, 1 (2019), 1–27.

[20] Jianfeng Zheng and Henry X. Liu. 2017. Estimating traffic volumes for signalized intersections using connected vehicle data. *Transportation Research Part C: Emerging Technologies* 79 (2017), 347–362.

[21] Longfei Zhou, Lin Zhang, and Chun S. Liu. 2021. Comparing roundabouts and signalized intersections through multiple-model simulation. *IEEE Transactions on Intelligent Transportation Systems* 23, 7 (2021), 7931–7940.

[22] Dan-Dan Zhu and Jun-Qing Sun. 2021. A new algorithm based on Dijkstra for vehicle path planning considering intersection attribute. *IEEE Access* 9 (2021), 19761–19775.

[23] Theresa Ziemke and Söhnke Braun. 2021. Automated generation of traffic signals and lanes for matsim based on openstreetmap. *Procedia Computer Science* 184 (2021), 745–752.