Data Processing and Enrichment of LiDAR-Derived Traffic Data

Ana Peykova [1], Dessislava Petrova-Antonova [2], Kaloyan Karamitov [2]

[1] Minerva university, San Francisco, California, USA – ana@uni.minerva.edu
[2] GATE Institute, Sofia University "St. Kliment Ohridski", Sofia, Bulgaria – (dessislava.petrova, kaloyan.karamitov)@gate-ai,eu

Keywords: LiDAR Traffic Data, Traffic Trajectory Data Enrichment, Semantic Mapping of Intersection Objects, Traffic Violation Detection, Traffic Flow Density.

Abstract

With the escalating demand for efficient traffic management and the increasing complexity of traffic control, diverse sensor technologies have been implemented to measure traffic in real-time. The road-side LiDAR emerges as a novel technology addressing the data gap in multimodal traffic analyses. LiDAR sensing return time to precisely capture distance and reflectivity, generating point cloud data encompassing all traffic trajectory information. It overcomes challenges posed by illumination conditions like light, dust and fog, which often affect camera sensor performance. In addition, LiDAR sensing minimises the effect of changing object position and angles, simplifying object detection and recognition.
This paper tackles the challenges of analysing LiDAR-derived traffic data by proposing a method for traffic trajectory data enrichment. The methodology followed includes creating a semantic map, bridging the physical space and raw data, transforming from a local to a standard Coordinate Reference System (CRS) and enriching data trajectory representation. Three use cases are presented based on the dataset obtained after enrichment: object classification, permissible directions violation detection, and traffic flow density. The proposed method is validated using traffic data from a LiDAR system of 6 sensors located in one of the busiest intersections in Sofia, Bulgaria. The raw sensor data is processed by a fusion box called the Augmented LiDAR Box, delivering time series frames with labelled moving objects in .osef format. The results prove that the proposed data enrichment method successfully transforms the trajectories into semantic sequences, opening up new avenues for trajectory analysis and intersection traffic micro-modelling.

1. Introduction

Intelligent transport systems (ITS) have undergone rapid development in recent years. Data plays a crucial role in evaluating transportation system performance and facilitating the development of innovative solutions to address traffic-related challenges. However, the acquisition and processing of traffic data, as highlighted by Barceló et al., present inherent challenges (2011). While synthetic error-free data is valuable for theoretical research, its limited practical applicability necessitates a shift towards the labour-intensive task of collecting, formatting and processing real-world data. Such data is derived from various measuring tools, a recent but promising one being Light Detection and Ranging (LiDAR) sensors, which allow for all-traffic object tracking and detecting road boundaries, road facilities, and traffic lanes (Lakshmanan, 2023). Despite its commercial availability, there's a gap between the data it provides and actionable insights, highlighting the need for methodologies to analyse and utilise LiDAR-derived traffic trajectory data effectively.

With the escalating demand for efficient traffic management and the increasing complexity of traffic control, diverse sensor technologies have been integrated into networks to measure the actual traffic state. Barceló et al. list conventional measuring tools for traffic analysis. LiDAR laser light return time to capture distance and reflectivity precisely, generating point cloud data encompassing all traffic trajectory information. This differs from traditional sensors, like radars, induction loops or infrared detectors, which offer limited macro-level vehicle or pedestrian data. LiDAR sensors exhibit high accuracy and frequency in detecting surrounding objects, overcoming challenges posed by illumination conditions like light, dust and fog that often affect camera sensor performance. Unlike camera sensors, LiDAR is minimally affected by changing object positions and angles, simplifying object detection and recognition.

Beyond technical capabilities, LiDAR addresses privacy concerns in data-driven ITS. As highlighted by Zhang et al. (2011b), the field must adopt privacy awareness and a people-centric approach for widespread acceptance of data-driven ITS. LiDAR data, being entirely anonymous, alleviates concerns related to the collection and storage of private information, enabling a shift towards human-centric analyses focusing on interactions and behavioural understanding rather than punitive actions.

The potential of LiDAR technology in various use cases has spurred significant research efforts in processing point cloud data and refining object detection and recognition algorithms (Zhang, 2021a; Xu, 2018; Song, 2020). These advancements have led to the commercial availability of LiDAR sensor systems from various vendors, equipped with embedded programming for real-time object clustering, classification, and metadata provision, effectively tracking vehicle and pedestrian movements. Despite the market presence of 3D LiDAR sensors with preprocessors packaged as user-friendly plug-and-play spatial intelligence solutions, the output, micro-traffic object trajectories, often lacks interpretability and straightforward insights, falling in the gap between data and actionable knowledge.

This paper proposes a method for effectively processing LiDAR-derived traffic trajectories to analyse vehicle traffic flow. It aims to utilise simple, straightforward, and maximally relevant tools and approaches to the field. GIS tools and machine learning modelling are combined to provide interpretability not only in the results but also in the working process followed. The applicability of the proposed method is shown by three use cases,

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-X-4-W5-2024-263-2024 | © Author(s) 2024. CC BY 4.0 License.

including object classification, permissible directions violation detection, and traffic flow density.

The remainder of this paper is organised as follows: section 2 gives a brief overview of the current state of the field, section 3 describes the area and the data used for this study, section 4 discusses the methodology employed for the elaboration of the proposed method, results and conclusions are presented in sections 5 and 6 respectively.

2. Related Work

A significant amount of research has been done in processing motion trajectory data. Much of the work is focused on macro-scale GPS-derived data like flight or ship trajectories; notable contributions focus on analysis based on distance measures, geometric features and heading (Laube, 2005; Rintoul and Wilson, 2015). These traditional features are less effective at micro-scale trajectories where motion is regulated, smoother and of varying lengths. Xi et al. propose a novel feature, a trajectory directional histogram (TDH), and effectively implement dominant-set clustering on micro-traffic intersection trajectory data (2006). Using spectral clustering, Xin et al. employ the TDH method to hierarchically cluster unreliable raw motion trajectories from micro-traffic video sequences (2011). Xu et al. iteratively shrinks trajectories to their cluster means to increase robustness and cluster separation, given incompleteness and noise in the data (2018).

The most established methods operate within an abstract data space, solely considering trajectory properties in relation to other trajectories. Few researchers have delved into connecting data to the physical space and analysing it within its real-world context. Extracting video sequence-derived data poses challenges, and aligning it with real-world features can be problematic due to factors like camera angles and depth.

In contrast, trajectories derived from LiDAR data extracted from point clouds offer accurate and consistent distance measurements, ensuring more precise mapping to real spaces (Gao, 2023). This characteristic of LiDAR-derived trajectories facilitates a simpler, more intuitive approach that connects trajectories to the spatial geometry of traffic intersections. Enriching trajectory data with semantic information tied to real-world spatial contexts simplifies trajectory processing and analyses. This approach diversifies clustering and classification problems by offering new data representation methods. Leveraging LiDAR's contextual information is close to how humans think of vehicle motion, making traffic flow analysis accessible without deep statistical or machine learning expertise. This is especially useful for interdisciplinary projects like ITS and data-driven urban planning. For example, a field experiment was conducted to collect high-precision vehicle trajectories using roadside LiDAR devices (Cao, 2024). A high-definition digital map of an urban road without signalised intersection in the city of Chengdu is created. The proposed method accurately divides lanes for 97.83% of trajectory points. The sensitivity analyses show that it is robust to the noise in trajectory data and the error of calibrating the roadside LiDAR devices.

The roadside LiDAR systems are sensitive to misdetections, resulting in discontinuous and shortened trajectories of objects. To address this, a Joint Detection And Tracking (JDAT) scheme is proposed to mitigate miss-detections at the vehicle detection stage (Zhang, 2022). The road users are separated by moving point semantic segmentation and instance clustering. Then, object detection and object tracking are conducted in parallel. The trajectories are identified from the object detection results using only a certain number of representatives for each trajectory. To address the occlusions of vehicles caused by the location of the installation on the lamppost, Lakshmanan et al. propose a robust vehicle detection model based on LiDAR sensor data (2023). A synthetically augmented transfer learning method is employed, and a synthetic LiDAR data generation tool is implemented, delivering a variety of vehicle shapes offered by the ShapeNet dataset along with ray casting. A rule-based method to identify a partially occluded vehicle in the parking lot is proposed by Thornton et al. (2014), considering any object with a length less than a predefined threshold, such as 2.5 meters, as an occluded vehicle. The occlusion might also be identified by checking whether the background curve can be seen between a given pair of vehicles (Lee, 2012). If not, the farther vehicle is suspected of being occluded. A similar approach is applied to detect pedestrians and estimate their trajectories from the occlusion time, not to observe the background points (Saki, 2019). The proposed method makes it possible to detect more pedestrians located far from sensors and in rather crowded situations compared with an existing scheme based on point clouds.

3. Study Area and Data

This project focuses on the traffic intersection of “Cherni Vruh” Bulevard and “Sreburna” Street in Sofia, Bulgaria. This intersection is one of the busiest in the capital city, connecting the ring road with Lozenets District, a large residential and downtown area. The neighbourhood is densely populated and undergoing intensive regeneration, making it a constant hotspot in the city. There are a big shopping mall, a hypermarket, multiple office buildings, multiple public transport stops and a large park in immediate proximity to the intersection. The road has 13 lanes for incoming traffic and 9 for outgoing traffic with intricate alignments.
The intersection regularly experiences congestion, particularly during rush hours. Furthermore, there are frequent accidents of different magnitudes, ranging from minor infractions and disregarding right-of-way to extreme speeding and illegal street racing.

The data comes from a LiDAR sensor system based on remote measurement technology used to track the movement of both pedestrians and motor vehicles. This technology operates by emitting infrared (invisible spectrum) laser light towards targets and measuring the time it takes for the reflected light to return to the sensor. By detecting variations in reflection times, instances of the laser beam, and different wavelengths, it generates point cloud data for the identified targets.

The LiDAR system comprises 6 sensors – one Ouster OS1 (Gen 2) Mid-range 128 and five Ouster OS1 (Gen 2) Mid-range 64s. The sensors are strategically placed at different traffic poles around the intersection. The raw point cloud data collected by the sensors is processed by a fusion box called the Augmented LiDAR Box (ALB). The software used for the fusion process groups points and recognises objects based on a 5% sample of the point cloud (Outsight, 2021). It distinguishes between static and moving objects. Figure 1 shows a visualisation of the static objects in the intersection.

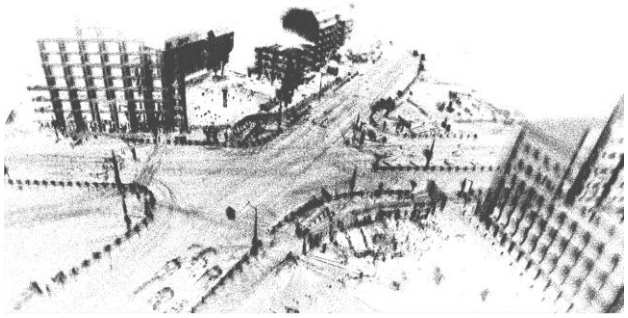


Figure 1. Point cloud visualisation of static objects in the intersection.

The data output by the preprocessor of the fusion box is organised into time series frames, each containing object IDs that persist across frames. In addition to IDs, each object in a frame is labelled with its class (such as car, truck, unknown, pedestrian, or two-wheeler), coordinates, estimated speed, volume, and more. Timestamps are according to the GMT time zone, while the local time zone is GMT+3; x, y, z coordinates are in a local coordinate reference system (CRS). The pre-processed data adheres to a format defined by Outsight (Open SErialization Format or OSEF), a nested Tag-Length-Value (TLV) tree, and can be downloaded in real-time via VPN through a TCP stream. Following the download, binary files can be parsed into a standard .csv format by traversing the TLV tree. The TLV tree with the actual node names is shown in Figure 2.

```
frame
  |____ tracked_objects
  |____ timestamp (Datetime)
  |____ number_of_objects (Integer)
  |____ object_ids (List)
  |____ object_classes
  | |____ class_name (String)
  | |____ class_code (Integer)
  |____ poses (List)
  | |____ translation
  | | |____ x (Float)
  | | |____ y (Float)
  | | |____ z (Float)
  | |____ rotation
  |____ bounding_boxes
  | |____ bbox_x (Float)
  | |____ bbox_y (Float)
  | |____ bbox_z (Float)
  |____ speed_vectors (List of Lists)
  |____ zones
      |____ bindings
      |   |____ object_id (Integer)
      |   |____ zone_index (Integer)
      |____ definitions (List of Objects)
          |____ zone_name (String)
```

Figure 2. Reconstruction TLV tree.

The dataset used for this study is a 7-minute sample from 10:01 to 10:08 on Monday, January 9, 2023. The dataset contains information about 2260 unique detected objects' trajectories described by 379 927 points. These objects, presented in Table 1, include vehicles and pedestrians. Among them, 1142 objects are consistent, including 11 cars, 3 persons, 1 truck, and 2 two-wheelers, and 1125 objects are unknown.

	Car	Truck	Two-wheeler	Person	Unknown
consistent	11	1	2	3	1125
recognised	676	13	82	254	-
undefined	-	-	-	-	-
total	687	14	84	267	1125

Table 2. Dataset description.

Since the Fusion box only samples 5% of the collected data for metadata production, the dataset has inaccuracies. This leads to issues such as most objects labelled as 'unknown', inconsistent object recognition (e.g., object initially labelled as 'person' then 'car' then 'two-wheeler'), incomplete trajectories, fluctuating volumes, etc.

4. Methodology

Figure 3 shows the workflow of the proposed method for traffic trajectory data enrichment with three potential use cases outlined.

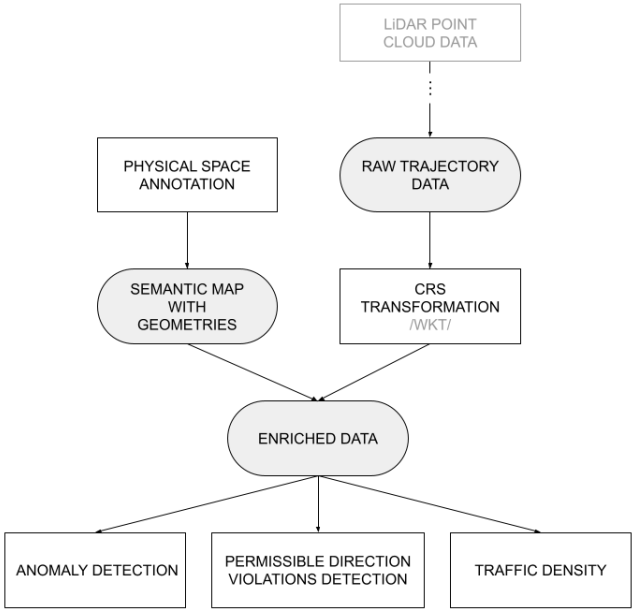


Figure 3. Data enrichment workflow. Rectangle shapes present actions, and obround boxes denote results.

4.1 Semantic Map

Bridging physical space and raw data involves digitising the space by creating a semantic map encompassing areas of interest. A semantic map comprises geometric objects such as polygons, points, or polylines, each annotated to represent specific features like stop lights, sidewalks, lanes, traffic islands, and more. Geometric features are outlined using satellite imagery with high-resolution satellite images of urban areas freely available online. The outlined geometries are extracted as polygons as follows:

$$P = \{(x_1, y_1), (x_2, y_2) \dots (x_n, y_n)\}, \quad (1)$$

where (x_i, y_i) = longitude and latitude in a known CRS.

4.2 Local to Standard CRS Transformation

A trajectory of a tracked object is a set of *n* positions, sampled at some rate, and can be represented as follows:

$$T = \{t_1, t_2, \dots t_n\} = \{(x_1, y_1), (x_2, y_2) \dots (x_n, y_n)\}, \quad (2)$$

where (x_i, y_i) = centroids of the object at the time *i*.

The coordinates are in respect to a local CRS. Given the precision and consistency of LiDAR measurements, the transformation between different CRS is a linear one. In the context of CRS transformations, an affine transformation involves a combination of translations, rotations, scaling and shearing, represented by a

https://doi.org/10.5194/isprs-annals-X-4-W5-2024-263-2024

matrix multiplication followed by a vector addition. Let's consider a basic 2D affine transformation matrix:

$$\begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix}, \quad (3)$$

To transform the trajectory *T*, the transformation matrix is applied to each position (x_i, y_i) as follows:

$$\begin{pmatrix} x_i' \\ y_i' \\ 1 \end{pmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x_i \\ y_i \\ 1 \end{pmatrix}, \quad (4)$$

The transformed positions (x_i', y_i') constitute the trajectory in the new coordinate system. The parameters *a*, *b*, *c*, *d*, *e* and *f* depend on the specific transformation. A minimum of three reference points are needed to derive the transformation matrix between two CRS. These points are pairs whose coordinates are known in both coordinate systems. Given original points (x_1, y_1), (x_2, y_2), (x_3, y_3) and transformed points (x_1', y_1'), (x_2', y_2'), (x_3', y_3'), a system of equations is set based on the general affine transformation equation:

$$\begin{pmatrix} x_1' & x_2' & x_3' \\ y_1' & y_2' & y_3' \\ 1 & 1 & 1 \end{pmatrix} = \begin{pmatrix} x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{pmatrix}, \quad (5)$$

Solving the system of equations gives the values of *a*, *b*, *c*, *d*, *e* and *f*. The inverse of the transformation matrix *M* represents the reverse transformation.

$$Orogonal = M^{-1} \times Transformed, \quad (6)$$

This is assuming that the matrix *M* is invertible, which is generally the case for valid affine transformations.

Well-Known Text (WKT) is a text representation format used to describe spatial objects and their geometries in the context of Geographic Information Systems (GIS). WKT provides a standardised way to represent and exchange spatial data as it includes the coordinate system and map projection needed to accurately interpret spatial data. For a local CRS, its definition can be encoded into WKT-CRS format using the transformation matrix *M* aligning it with known CRS standards such as EPSG:3857 or EPSG:4326, widely adopted by platforms like Google Earth and OpenStreetMap. This encoding process is easily done through GIS software like QGIS or Python's pyproj library.

4.3 Enriched Data Trajectory Representation

Once the CRS transformation is applied, connecting the data with the space is as simple as a point-to-polygon check. Given polygons of interest $P_1, P_2, \ldots P_n$, T can be expressed as $T = \{t_1 \in P_i, t_2 \in P_{i,}, \ldots t_n \in P_i\}$. A short-hand $T = \{1,1,1,0, \ldots 1\}$ can be interpreted as the tracked object is in polygon 1 for 3 timesteps, then it moves to polygon 0, and so on. Given that the trajectories represent the movement of objects at a traffic intersection, where regulated movement in lanes is expected, the object's motion within a polygon can be inferred.
A semantic map should have detailed and comprehensive annotations. For instance, lane annotations could encompass attributes like street affiliation, cardinal direction, entry or exit points at intersections, permissible turns, and other relevant characteristics. These annotations contribute to a richer dataset allowing for diverse data representations and classification based on pattern recognition.

For example, entering and leaving the intersection can be encoded as a binary variable. Then trajectories crossing the intersection are expected to follow the pattern $R = \{1, \ldots 1,0,0, \ldots\}$ in their representation, denoting being present in a lane headed toward the intersection, then moving in a lane exiting the intersection. A right or a left turn can be detected based on cardinal directions by detecting a shift in direction. For example, a trajectory representation $R = \{\ldots N, E, E, \ldots\}$ constitutes a right turn.

4.4 Vehicle and Pedestrian Classification

At an intersection, vehicles are anticipated to adhere to traffic regulations. Objects of different classes tend to occupy specific areas within the intersection. For instance, vehicles typically use roads, while pedestrians are expected to be on sidewalks or crosswalks. Thus, an object's class (vehicle or pedestrian) can be inferred from the path. It's important to note that in this scenario, classification is simplified into two categories – vehicle or pedestrian – without distinguishing between specific vehicle types (such as a truck or a car).

Given the noisy nature of the data, where trajectories can be unclear, and objects like pedestrians and cars might stray from their designated areas (e.g., pedestrians not strictly sticking to sidewalks), a practical approach for classification is to set a time threshold for an object's presence in a pedestrian or non-pedestrian area. The classification is based on heuristics aiming to perform a cheap-and-easy distinguishing of pedestrians and vehicles. Setting a threshold of 75% road area for vehicles (that is, if an object spends 75% or more of its time on the road area, it is classified as a vehicle) can produce a seemingly good classification. Since the crosswalk trajectories might be mostly located on the road, some of the pedestrian objects might be labelled as vehicles.

Vehicle trajectories can be recognised by following some legal paths. A complete vehicle trajectory is a trajectory that begins in an incoming lane and ends in a lane leaving the intersection. If such a trajectory is detected, the object it belongs to is labelled as a vehicle. The remainder are either pedestrian or incomplete.

4.5 Permissible Directions Violation Detection

Beyond sequence patterns that are generally true across all traffic intersections, trajectories can be screened for patterns based on intersection-specific rules. Mapping out lanes' permissible directions allows for violation screening and detection. The permissible directions for each lane were gathered from the lane markings as seen in recent satellite images. One assumption was that if, for example, a turn is permissible, then any outgoing lane to the left is a legal terminating lane for the turn.

Based on the known intersection rules, a hash table of valid patterns can be created. Any pattern not found in the hash table is flagged as illegal. Such analyses bring attention to lanes where drivers violate the permissible directions frequently and spur further investigation of the cause such as poor signage, poor infrastructure, or high traffic density.

4.6 Traffic Flow Density

Traffic flow quantifies the volume of traffic in relation to road length. It's determined by the number of objects *n* observed in a

specific lane L_i at a given time t. As trajectories in the dataset may differ in length, origin, and endpoint, and considering the non-uniform nature of lane geometries, road length assessment should be conducted lane by lane.

$$U(l_i) = max_{(x,y),(u,v)\in P} dist((x,y),(u,v)), \quad (7)$$

where $U(l_i)$ = the maximum distance between two points in the set of points P_i within lane L_i.

This mitigates differences in sensor range in respect to each lane.

5. Results

This section presents the results of the data enrichment workflow, including constructing a semantic map, object classification, violation detection and visualisation of traffic flow density.

5.1 Semantic Map

The constructed semantic map contains information about 12 incoming and 10 outgoing traffic lanes, 4 crosswalks, 7 metro exits, 3 traffic islands and 1 bus bay (Figure 4). All features are annotated with several properties, described in Table 2.

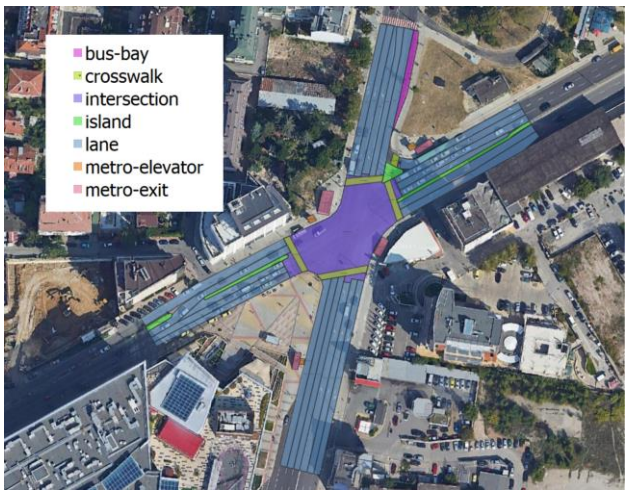


Figure 4. Visualisation of the semantic map of "Cherni Vruh" Boulevard and "Sreburna" Street intersection. Features are color-coded by type.

The semantic map is exported as a geoJSON file, which can be easily opened and manipulated in GIS software as a vector layer and in Python using geopandas. Using the transformation derived above, the geometry of the intersection is available both in standard EPSG 3857 and local CRS.

Property	Description
id	Unique ID of the polygon
type	"lane", "crosswalk", "island", "intersection", "bus-bay", "metro-exit", "metro-elevator"
street	Name of the street the polygon belongs to
direction	Cardinal direction of the lane
to_inter	Lanes: 1 if moving toward the intersection, 0 otherwise
left_turn	Lanes: 1 if left turn is legal, 0 otherwise
right_turn	Lanes: 1 if right turn is legal, 0 otherwise
straight	Lanes: 1 if continuing straight is legal, 0 otherwise

Table 2. Properties of polygons of interest, included in the semantic map. The properties are focused on describing vehicle movement.

Having the semantic map, each point of the data can be located within the outlined geometries, giving context to the trajectories in the physical intersection.

5.2 Local to Standard CRS Transformation

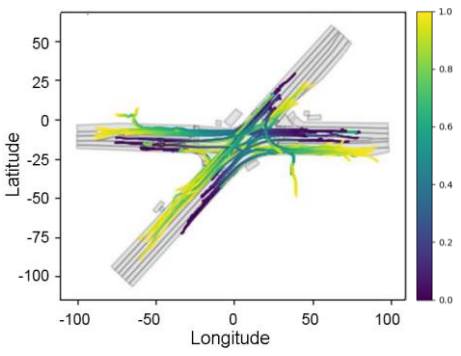
To check the accuracy of the local to standard CRS transformation, a 2D projection of the LiDAR data, presenting the static objects of the intersection is used. Figure 5 shows that the static objects' points well map out to actual physical objects at the intersection, proving the transformation WKT is sufficiently accurate.

Figure 5. A 2D projection of a 50 000-point sample from the point cloud with static geometry transformed from a local CRS to EPSG 3857.

5.3 Vehicle and Pedestrian Classification

The initial dataset of 2260 objects were filtered to separate out the complete vehicle trajectories. Such trajectories contain the pattern *{1,0}* of heading direction relative to the intersection area (*to_inter*). The filtered dataset consisted of 78 vehicles (Figure 6a) and was used in the use cases, presented in the next sections. The remaining trajectories were split based on location among two groups: incomplete vehicle trajectories (Figure 6b) and pedestrian trajectories (Figure 6c). For the incomplete vehicle trajectories, only the entry and exit lane is known. Note, there are no crosswalk trajectories in Figure 6c, since they are likely classified as incomplete instead of pedestrian.

(a)



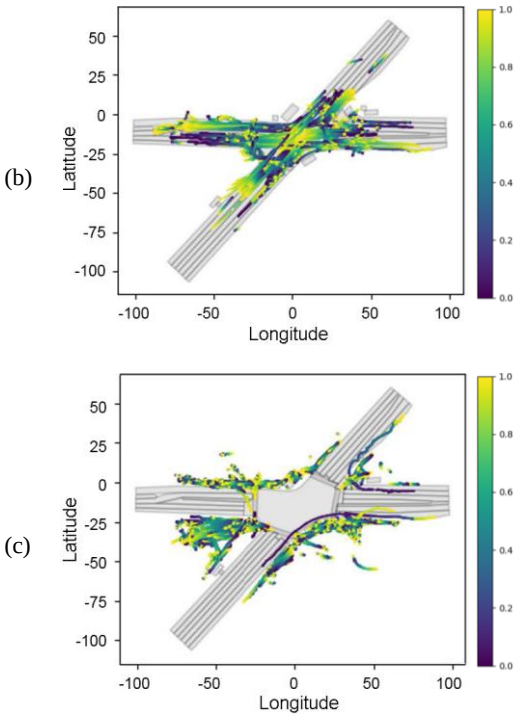


Figure 6. Trajectories color-coded based on normalised interpolation between the start (purple) and end (yellow). (a) Complete vehicle trajectories; (b) Incomplete vehicle trajectories; (c) Pedestrian trajectories.

5.4 Permissible Direction Violations

The 78 complete trajectories were compared to a list of permissible patterns. Out of them, 3 were flagged as illegal, constituting 2 illegal straights and 1 illegal right turn. All 3 violations belonged to the same incoming street, highlighting a potentially problematic area (Figure 7).

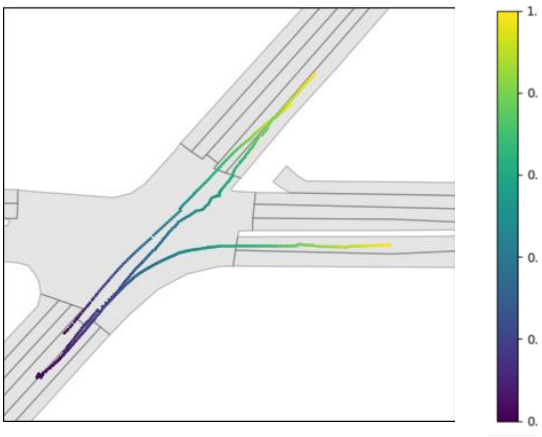


Figure 7. Detected violations in the data sample. Trajectories are color-coded based on normalized interpolation between start (purple) and end (yellow). All three violations (two illegal straight and one illegal right turns) occur on the northbound entry lanes of the intersection.

5.5 Traffic Flow Density

The count of objects passing through a lane is used as a proxy for traffic density. This analysis includes all 2260 objects, recognising that even incomplete trajectories contribute to traffic flow. The object locations are compiled to count the number of vehicles passing through each lane. The resulting counts are visualised in a heatmap (Figure 8). Traffic lanes' object count ranges from 20 to 70 vehicles over the whole period. Some of the busiest lanes are the northeast- and north-bound outgoing lanes of the intersection. This could be due to the fact that both of these directions have only 2 outgoing lanes that need to handle the traffic coming from 3 or 4 incoming lanes.

Figure 8. Heatmap of lane traffic density for an interval of 7 minutes.

6. Conclusion and Future Work

This data enrichment process transforms trajectories into semantic sequences, opening up new avenues for trajectory analysis. Particularly with advancements in language processing, this becomes an intriguing area for researchers. In these three use cases, subsequence search techniques are employed to detect semantic patterns. This method facilitates the identification of abnormal behaviours and accident-prone zones by leveraging predefined rules of the intersection.

The proposed method for data representation rests upon the assumption that objects generally follow the same motion within different polygons. This is not an unreasonable assumption for large and highly-regulated intersections when tracking vehicle traffic. However, its applicability may not translate to pedestrian trajectories, which tend to follow less strict patterns.

The effectiveness of the proposed methods is strongly influenced by the presence of noise in the data. Despite the theoretical high accuracy and precision of LiDAR measurements, the preprocessing steps, such as object detection and tracking, introduce notable errors. To enhance computational efficiency, preprocessors often operate on partial data extracted from the original point cloud, which can result in misdetection and other issues. Recalculating the object's centroid at each frame contributes to noisy trajectories.

To address complex intersections or multiple intersections simultaneously, there is considerable potential in exploring unsupervised machine learning models to infer traffic regulations directly from the data without the need for manual encoding. Noisy data can be mitigated by applying polyline simplification algorithms or smoothing filters, such as the Kalman filter.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-X-4-W5-2024-263-2024 | © Author(s) 2024. CC BY 4.0 License.

Finally, the data sample used for this study covers a very short time period of 7 minutes. More data is required to further explore and make conclusive statements about traffic patterns at the intersection, including comparing data from different times of day. This could cover rush hour, workday and weekend dynamics, and illegal nighttime racing.

Acknowledgements

This research is part of the GATE project funded by the Horizon 2020 WIDESPREAD-2018-2020 TEAMING Phase 2 programme under agreement no. 857155, the enrRichMyData project, funded by Horizon Europe research and innovation programme under agreement no. 101070284, and the INTEND project, funded the European Union's Horizon 2020 research and innovation programme under grant agreement no. 101135576.

References

Barceló, J., Kuwahara, M., & Miska, M. 2010. Traffic Data Collection and its standardization. *International Series in Operations Research & Management Science*, 1–10. https://doi.org/10.1007/978-1-4419-6070-2_1

Cao, P., Wang, Y., & Liu, X. 2024. Trajectory-Based Method for Dividing Lanes of Vehicle Trajectories Collected by Roadside LiDAR. *Transportation Research Record*, 2678(3), 816-828. https://doi.org/10.1177/03611981231182987

Gao, Y., Zhou, C., Rong, J., Wang, Y., 2023. High-accurate vehicle trajectory extraction and denoising from roadside LIDAR sensors. *Infrared Physics & Technology* 134, 104896. https://doi.org/10.1016/j.infrared.2023.104896

He, X., Li, Q., Wang, R., & Chen, K. 2022. A spatio-temporal feature trajectory clustering algorithm based on deep learning. *Electronics*, 11(15), 2283. https://doi.org/10.3390/electronics11152283

Lakshmanan, K., Roach, M., Giannetti, C., Bhoite, S., George, D.;,Mortensen, T., Manduhu, M., Heravi, B., Kariyawasam, S., Xie, X. 2023. A Robust Vehicle Detection Model for LiDAR Sensor Using Simulation Data and Transfer Learning Methods. *AI* 2023, 4, 461-481. https://doi.org/10.3390/ai4020025

Laube, P., van Kreveld, M., & Imfeld, S. 2005. Finding remo — detecting relative motion patterns in geospatial lifelines. *Developments in Spatial Data Handling*, 201–215. https://doi.org/10.1007/3-540-26772-7_16

Lee, H. and Coifman, B. 2012. Side-fire lidar-based vehicle classification. *Transportation Research Record*, 2308(1), pp.173-183.

Rintoul, M. D., & Wilson, A. T. 2015. Trajectory analysis via a geometric feature space approach. Statistical Analysis and Data Mining: *The ASA Data Science Journal*, 8(5–6), 287–301. https://doi.org/10.1002/sam.11287

Saki, T., Hiromori, A., Yamaguchi, H., Higashino, T. 2019. Occlusion-based trajectory estimation for pedestrians using LiDAR sensors. 10th International Conference on Indoor Positioning and Indoor Navigation - Work-in-Progress Papers (IPIN-WiP 2019) co-located with the Tenth International Conference on Indoor Positioning and Indoor Navigation (IPIN 2019), Pisa, Italy, September 30th - October 3rd, 2019. Volume 2498 of CEUR Workshop Proceedings, pp. 400-407, CEUR-WS.org, 2019.

Song, Y., Zhang, H., Liu, Y., Liu, J., Zhang, H., & Song, X. 2020. Background filtering and object detection with a stationary lidar using a layer-based method. *IEEE Access*, 8, 184426–184436. https://doi.org/10.1109/access.2020.3029341

Thornton, D.A., Redmill, K. and Coifman, B. 2014. Automated parking surveys from a LIDAR equipped vehicle. *Transportation research part C: emerging technologies*, 39, pp.23-35

Xi Li, Weiming Hu, & Wei Hu. 2006. A coarse-to-fine strategy for vehicle motion trajectory clustering. 18th International Conference on Pattern Recognition (ICPR'06). https://doi.org/10.1109/icpr.2006.45

Xin, L., Yang, D., Chen, Y., & Li, Z. 2011. Traffic flow characteristic analysis at intersections from multi-layer spectral clustering of motion patterns using raw vehicle trajectory. 2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC). https://doi.org/10.1109/itsc.2011.6082850

Xu, Hao, Tian, Z., Wu, J., Liu, H., & Zhao, J. 2018. (rep.). High-Resolution Micro Traffic Data From Roadside LiDAR Sensors for Connected-Vehicles and New Traffic Applications (p. V–VIII, 1–3, 31–36). Reno, Nevada: Department of Transportation.

Xu, Hongteng, Zhou, Y., Lin, W., & Zha, H. 2015. Unsupervised trajectory clustering via adaptive multi-kernel-based shrinkage. 2015 IEEE International Conference on Computer Vision (ICCV). https://doi.org/10.1109/iccv.2015.492

Zhang, J., Pi, R., Ma, X., Wu, J., Li, H., & Yang, Z. 2021. Object classification with Roadside Lidar data using a probabilistic neural network. *Electronics*, 10(7), 803. https://doi.org/10.3390/electronics10070803

Zhang, J., Wang, F.-Y., Wang, K., Lin, W.-H., Xu, X., & Chen, C. 2011. Data-Driven Intelligent Transportation Systems: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 12(4), 1624–1639. https://doi.org/10.1109/tits.2011.2158001

Zhang, J.; Xiao, W.; Mills, J.P. 2022. Optimizing Moving Object Trajectories from Roadside Lidar Data by Joint Detection and Tracking. *Remote Sens.* 2022, 14, 2124. https://doi.org/10.3390/rs14092124

Appendix

Table 2 presents the optical performance of the Long-Range High-Resolution Imaging LiDAR sensors used for data acquisition.

Characteristic	Description
Range (80% Lambertian reflectivity, 2048 @ 10 Hz mode)	100 m @ >90% detection probability, 100 klx sunlight 120 m @ >50% detection probability, 100 klx sunlight
Range (10% Lambertian reflectivity, 2048 @ 10 Hz mode)	45 m @ >90% detection probability, 100 klx sunlight 55 m @ >50% detection probability, 100 klx sunlight
Minimum Range	0.3 m for point cloud data

Characteristic	Description
	0 m - 0.3 m blockage detection flag to indicate object within 0.3 m (v2.0 beta feature)
Range Accuracy	±3 cm for lambertian targets, ±10 cm for retroreflectors
Precision (10% Lambertian reflectivity, 2048 @ 10 Hz mode, 1 standard deviation)	0.3 - 1 m: ± 0.7 cm 1 - 20 m: ± 1 cm 20 - 50 m ± 2 cm >50 m: ± 5 cm
Range Resolution	0.3 cm
Vertical Resolution	32, 64, or 128 channels

Characteristic	Description
Horizontal Resolution	512, 1024, or 2048 (configurable)
Field of View Vertical	45° (+22.5)
Angular Sampling Accuracy	Vertical: ±0.01° / Horizontal: ±0.01°
False Positive Rate	1/10,000
Rotation Rate	10 or 20 Hz (configurable)
Number of Returns	1 (strongest)

Table 2. Optical performance of the Long-Range High-Resolution Imaging LiDAR sensors.