Journal of Internet Services and Applications, 2023, 14:1, doi: 10.5753/jisa.2023.3075

User-centered analysis of a safe bus routing strategy

João Marcos A. M. Ramos [**Universidade Federal de Viçosa - Florestal** | ***joao.m.ramos@ufv.br***]
Vinícius G. J. Almeida [**Universidade Federal de Viçosa - Florestal** | ***vinicius.jesus@ufv.br***]
Henrique S. Santana [**Universidade Federal de Viçosa - Florestal** | ***henrique.s.santana@ufv.br***]
Thais R. M. Braga Silva [**Universidade Federal de Viçosa - Florestal** | ***thais.braga@ufv.br***]
Fabrício A. Silva [**Universidade Federal de Viçosa - Florestal** | ***fabricio.asilva@ufv.br***]

Universidade Federal de Viçosa - Campus Florestal, Rodovia LMG 818, km 06, s/n, Campus Universitário, Florestal - MG, 35690-000

Received: 23 December 2022 • **Accepted:** 28 March 2023 • **Published:** 20 June 2023

Abstract Context-aware mobility has the potential to make the way we travel more efficient, safer, and more sustainable. Among the possible contexts, safety, in terms of crime levels in city regions, is one that has been used to calculate safer routes. Making a bus route safer is important to improve the quality of life of the passengers, who often are victims of criminals during their journey.However, existing studies focus only on private vehicles and do not assess the impact for citizens as a whole. In this work, an existing solution for calculating safe routes is evaluated in the context of public bus transport in terms of the impact caused to passengers. The results showed that, in general, changing a bus route to make it safer increases the distance traveled by a few kilometers for most passengers. This small increase in distance is not harmful to the passengers, given that they will be at less risk to face any kind of criminal situation. In addition to this analysis, a scalable tool for extracting mobility flow was also developed.

Keywords: context-aware mobility, public transportation, safe routing, flow extraction

1 Introduction

Context-aware mobility is a growing research area, aiming to use data about people and their environment to help improve their movement (Santos *et al.*, 2017). Thus, it is expected that citizens displacement can become more efficient, safer, more sustainable, and customized. Among those goals, safety has increasingly become more relevant, and it consists of tracing mobility routes crossing regions of lower criminality rates, in order to protect vehicles or people following such routes.

Currently, existing works that calculate safe routes have focused their efforts on building strategies geared towards private vehicles. Overall, those solutions involve tracing routes for a car or motorcycle that needs to travel from an origin to a destination point. Despite being useful for mobility, those proposals have not been evaluated for public transportation, which is responsible for carrying numerous people around big cities. Therefore, criminality context awareness should also be integrated with bus routes to benefit their passengers.

To the best of our knowledge, (Almeida *et al.*, 2022) is the only work found in the literature that aims to build safe routes for public transportation. The authors use an objective equation to calculate criminality scores for regions of different shapes. The solution was implemented for the bus routes of the São Paulo city in Brazil, changing the paths through which buses move between stops to make them safer. However, that study only evaluated its impact in terms of the routes themselves, overlooking the passengers perspective – the users who are in fact affected by the route changes.

As such, the purpose of the present work is to conduct a user-centered assessment of (Almeida *et al.*, 2022).

The objective of this evaluation is to determine whether the solution to find safer routes would affect a considerable number of people in a real scenario, and whether the changes made would positively or negatively affect the largest portion of the population. Our hypothesis is that it is possible to find safer routes to keep passengers less vulnerable to criminal situations while not increasing the length of the route significantly. We evaluated the impact on passengers after the proposed changes to make bus routes safer, taking as input a real dataset composed of more than 300,000 mobile users. Results show that, in general, changes in paths affected negatively only a small portion of citizens, which would have to travel for up to one extra kilometer to get to their destination.

In order to achieve the intended results, a mobility flow matrix of thousands of users was calculated, then mapped to bus route segments. Since we are dealing with large amounts of data, a scalable tool for flow extraction was implemented, which managed to reduce processing time up to seven times when compared to an already existing equivalent tool. This large-scale flow extraction tool is also an important contribution of our work.

This work is an extension of our previous study published in Portuguese at the Brazilian Symposium on Ubiquitous and Pervasive Computing (SBCUP 2022). In this version, we included new content regarding the related works, the formal description of the solution, and the data description and characterization. Also, we included new metrics in the results, as well as a more detailed discussion on them. With the new results, it was possible to observe that the changes proposed to the bus routes in order to make them safer affect only a few people and that this impact is not significant. Therefore, the adoption of a context-aware solution to make the routes of public transportation safer may be a good opportunity to improve the citizens quality of life.

The remainder of the text is organized as follows: Section 2 presents related works found in the literature. Section 3 describes the implementation of a tool for mobility flow extraction. The safe bus routing strategy proposed by Almeida *et al*. (2022) is combined with a user-centered analysis to show the relevance of creating safer routes to avoid or reduce passenger exposure at bus stops and criminally dangerous regions, since these end up being the places with the greatest risks to their integrity. The details and results of these implementations are presented in Sections 4 and 5. Finally, in Section 6, conclusions and future directions are given.

2 Background and related work

The baselines for this work can be divided in two categories, according to their contributions: safe route analysis and flow extraction.

Regarding safe route analysis, there are different approaches in the literature to handle this problem. In (Liu *et al*., 2017), SafeRNet is presented as a framework based on Bayesian networks to create safety probability estimation. Criteria are added to the network, such as vehicle flow, road conditions, weather, and traffic collisions, in order to provide route variations and lower the number of accidents to private vehicles. Similarly, Mata *et al*. (2016) present a Bayesian networks framework for private vehicles used to build safe routes from a hybrid recommendation system, consuming data from official criminal records and from Twitter. The work allows observing the impacts caused by the use of time windows, in addition to the improvements in route safety and adaptation to schedules caused by them. SafePaths (Galbrun *et al*., 2016) is another route recommendation system, which uses the Gaussian Kernel Density Estimation (KDE) clustering approach. Chicago and Philadelphia are the cities used as input, and the strategy to build routes is made on small regions, such as neighborhoods and streets, allowing to estimate the relative criminal probability on any road segment. The authors state their solution is robust in terms of execution time and update of new criminal records.

In (Ladeira *et al*., 2019), one more safe route identification solution was proposed for private vehicles, using the Density-Based Spatial Clustering of Applications with Noise (DBSCAN), well known for its good results even in the presence of noise and diverse cluster shapes and sizes (Santos *et al*., 2018). This solution is combined with a probability density function called Parzen Window. Unfortunately, there is no published study written in English that contemplates the use of the Parzen Window function. Results show a significant drop in the number of clusters through which vehicles should go, after opting for safer routes instead of shorter ones.

Lastly, Hot Routes (Tompson *et al*., 2009) is a solution focused on public transport – more specifically, buses –, whose goal is to identify unsafe regions in the city of London, based on criminal data and a KDE clustering technique. The resulting information is presented on a map as bus stops and route segments. However, improvements regarding stop locations are not shown, nor safer route solutions are proposed.

Table 1 presents a summary of the described related works, highlighting the type of vehicles targeted by their solutions, the implemented algorithms, the usage of temporal windows, and their application context. Among all the presented solutions, (Almeida *et al*., 2022) - the base of this study - is the only work that focuses on calculating safer routes for buses.

The city of São Paulo has been the subject of other studies about its population mobility. There are works such as (Moreno-Monroy *et al*., 2017), in which an analysis is conducted using public transport data from the city and its metropolitan area, in order to evaluate how accessible are the schools of São Paulo. The authors describe a metric consisting of the spatial distribution of students, the school locations, and the public transport vehicles that serve the influence area of schools. Martins *et al*. (2021) also use data from the metropolitan area of São Paulo, due to its intense traffic, to create a trajectory visualization solution with *Trail Bundling*, a technique to group trajectories next to each other in a simplified representation, differing from the concept of mobility flow in the way trajectories are grouped.

The other main concept dealt with in the present work is the origin-destination (OD) matrices, used during the mobility flow calculation step. The definitions of OD matrices are presented in works such as (Barbosa *et al*., 2018), which gives an overview of multiple approaches to studying human mobility. In (Iqbal *et al*., 2014), origin-destination matrices are created from Call Detail Records (CDR) and limited traffic counts.

Regarding the mobility flow calculation, the authors of (Guo *et al*., 2012) deal with the spatial grouping of massive amounts of GPS points to identify potentially significant locations, and extract and map aggregated flow metrics. Those, in turn, are used to understand the spatial distribution and temporal tendency of movements. As for (Kon *et al*., 2021), the authors describe the use of mobility flow to analyze movement patterns in a bike-sharing system, employing a method capable of processing millions of trips.

There are some known tools and libraries to work with mobility data. We can highlight MovingPandas (Graser, 2019), which deals with individual trajectory data but lacks aggregated metrics for flow extraction. On the other hand, there is also Scikit-Mobility (Pappalardo *et al*., 2021), providing methods to handle both trajectory and flow data. However, neither of those libraries perform well for big volumes of data since the entirety of datasets must be loaded into memory at once, and they lack support for parallel or distributed computation. This problem is addressed in the proposed flow extraction tool developed in this work.

3 A tool for mobility flow extraction

In the area of trajectory data analysis, an important task is to extract mobility flow, which consists of counting the number of collective movements between regions to determine areas of intense traffic. This metric is interesting for both private and public applications, helping with urban planning, advertisement, recommendation systems, and environmental impact studies, among others (Iqbal *et al*., 2014; Guo *et al*., 2012).

In this work, we use mobility flow to evaluate which bus

Table 1. Characteristics of state-of-the-art solutions.

	Target	**Algorithm**	**Temporal window**	**Context**
Baseline (Almeida *et al.*, 2022)	**Buses**	DBSCAN and Parzen Window	**Yes**	**Criminal**
(Ladeira *et al.*, 2019)	Private vehicles	DBSCAN and Parzen Window	Yes	Criminal
(Santos *et al.*, 2018)	Private vehicles	DBSCAN and Google Maps	Yes	Criminal
SafeRNet (Liu *et al.*, 2017)	Private vehicles	Bayesian Networks	Yes	Traffic (accidents)
(Mata *et al.*, 2016)	Private vehicles	Bayesian Networks	Yes	Criminal
SafePaths (Galbrun *et al.*, 2016)	Private vehicles	KDE	No	Criminal
Hot Routes (Tompson *et al.*, 2009)	Buses	KDE	No	Criminal

routes segments – as defined in Section 4.2 – are most used by citizens, and consequently how many people are affected by each route relocation. Given the need for large-scale flow extraction, we have developed a tool to tackle this problem, whose design is summarized in Figure 1 and detailed in the following sections.

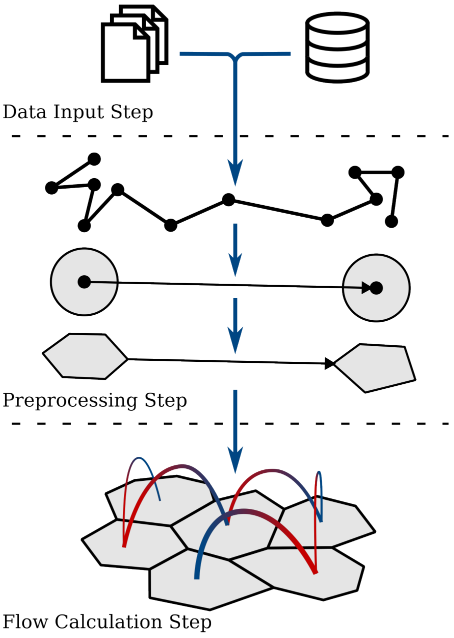


Figure 1. Summary of designed steps on the mobility flow calculation tool.

3.1 Design

First, the Apache Spark (Zaharia *et al.*, 2012) framework, alongside the Apache Sedona (Yu *et al.*, 2019) library, were used as the tool back-end in order to benefit from its computational efficiency and parallelism. We also opted for implementing it with the Scala programming language for performance gains.

We define three main steps for the scope of our tool: data input, preprocessing, and mobility flow calculation. Each of those steps has a particular input and output tabular format, in which each data record is a row and features are represented as columns.

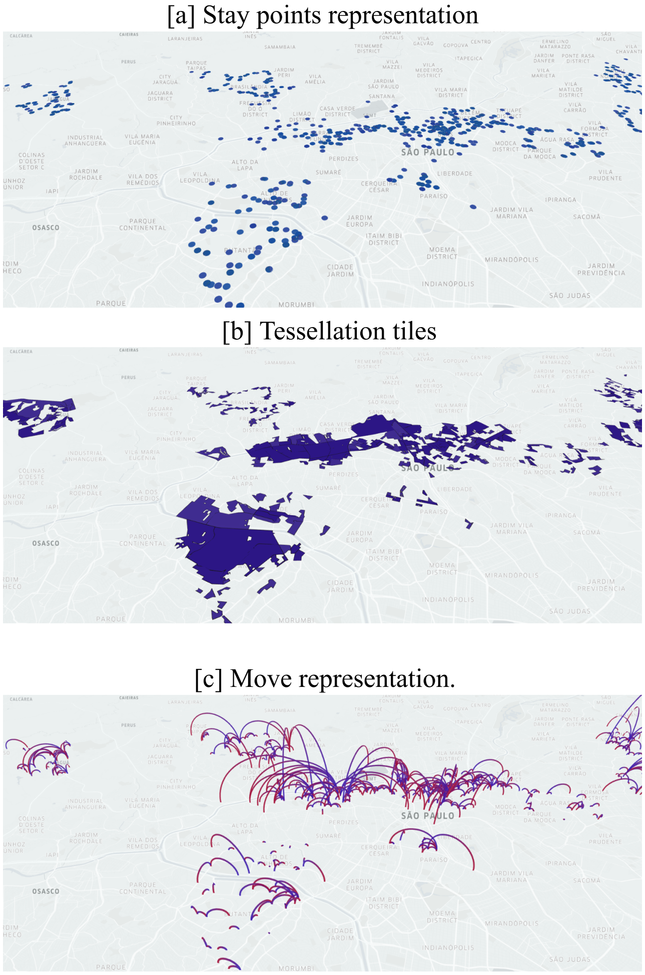
In the first step, each data entry consists of instantaneous temporal information associated with a specific *point in space*. Thus, each row p_i can be defined as a tuple $(u_i, \phi_i, \lambda_i, t_i)$, where u_i is a user identifier, ϕ_i and λ_i are respectively the latitude and longitude, and t_i is a timestamp.

However, this instantaneous point representation does not convey the meaning of end-to-end movements, because multiple entries can be recorded as an object moves. Consequently, these points must be transformed during the preprocessing step into *stay points* (Fig. 2a) through the algorithm presented in (Montoliu *et al.*, 2011), which was implemented in our solution. This algorithm uses three parameters to filter out points detected as intermediates in a movement, and aggregates nearby points into one. One parameter determines the maximum distance points can be far from each other to be considered in the same stay point. Also, there are two time-related parameters to establish a minimum and maximum permanence time for each stay point. By doing so, each data record sp_i, i.e., each stay point, is defined as a tuple $(u_i, \phi_i, \lambda_i, ts_i, te_i)$, where u_i, ϕ_i and λ_i have the same meaning as the previous format, and ts_i and te_i respectively represent the start and end timestamps for the permanence of that user on the indicated point.

Nonetheless, it is not useful to represent moves as stay points to calculate flow, because we aim for an aggregated view of movements, to count the number of moves between *regions*. In this sense, the notion of tessellations is convenient: a division of space into regions, usually expressed as a set of polygons. Still during the preprocessing step, we used the Apache Sedona library to implement a method to find in which polygon of a tessellation each geographical point is located, so that its exact coordinates can be labeled by the identifier of that polygon. Therefore, each data record st_i is then defined by a tuple $(u_i, \theta_i, ts_i, te_i)$, where u_i, ts_i and te_i have the same meaning as in the previous format, and θ_i is a unique identifier for the polygon that contains the respective ϕ_i and λ_i coordinates from sp_i.

For this work, we used as tessellation the polygons of census sectors provided by the Brazilian Institute of Geography and Statistics (IBGE). Figure 2b shows the polygons associated with the points from the previous image.

To finish the preprocessing step, data is finally converted into a more explicit representation for *movements*, in which each row m_i is defined as a tuple $(u_i, \theta_{oi}, \theta_{di}, t_o, t_d)$, where u_i is a user identifier, t_o is a timestamp indicating when that user departed from their origin region θ_{oi}, and t_d is a timestamp indicating when the user arrived at their destination region θ_{di}. We assume there is a move from two regions when there are consecutive stay points for the same user between those regions. That is, given two consecutive stay points sp_k and sp_{k+1} that belong to the same user, a movement record m_k is constructed as $(u_k, \theta_k, \theta_{k+1}, te_k, ts_{k+1})$. Figure 2c shows the same points of previous images, but with each curve representing a move, with the origin colored in red, and the destination colored in blue.


[a] Stay points representation

[b] Tessellation tiles

[c] Move representation.

Figure 2. Example of data representation in the preprocessing step.

On the last step inside the scope of the tool, mobility flow is finally calculated. Mobility flow can be defined as the number of moves that occurred between pairs of origin and destination regions during a given time interval, considering both outflow and inflow. Outflow refers to the movements going outwards from a region and inflow refers to the ones going towards a region. Formally, given a time interval $[T_0, T_f]$ and a set of move records $M = \{m_1, ..., m_k\}$, outflow for a pair of regions θ_A and θ_B is defined as $f_{AB,out} = |\{(u_i, \theta_{oi}, \theta_{di}, t_o, t_d) | \theta_{oi} = \theta_A, \theta_{di} = \theta_B, T_0 \leq t_o \leq T_f\}|$. Similarly, inflow in defined as $f_{AB,in} = |\{(u_i, \theta_{oi}, \theta_{di}, t_o, t_d) | \theta_{oi} = \theta_A, \theta_{di} = \theta_B, T_0 \leq t_d \leq T_f\}|$. Note that the only variable that changes in those two definitions is which timestamp is considered: either the departure or the arrival time. Thus, if only a single time interval is used, comprising the whole dataset, both outflow, and inflow will be the same value. For simplicity sake, we used a single time interval in this work, and so outflow and inflow are referred to as simply *flow* – denoted by $f_{AB} = f_{AB,out} = f_{AB,in}$ for a pair of regions θ_A and θ_B.

Flow is usually expressed as an origin-destination (OD) matrix, mathematically defined as $F = (f_{ij})_{1 \leq i \leq n, 1 \leq j \leq n}$, where n is the number of unique regions in the dataset. In our implemented solution, after associating stay points to identifiers of their corresponding regions, flow is calculated by using these identifiers as aggregation keys, and counting the number of records. So, the OD matrix is implemented in a tabular format, where each row is a tuple $(\theta_A, \theta_B, f_{AB})$.

3.2 Performance analysis

To assess the efficiency of the implemented flow extraction tool, we compared its execution time against the already known Scikit-Mobility library. The objective here is to measure how efficient, in terms of computational time, the proposed tool is to extract mobility flow from large datasets when compared to Scikit-Mobility. It is important to state that this tool can be used in different problems that require mobility flow and is not responsible for route calculation. Considering the stay point detection algorithm used by Scikit-Mobility is not the same as the one we implemented and could thus lead to different outputs, the scope of this performance analysis only takes into account the process of using a tessellation to associate points to region identifiers, and the aggregation step to calculate the flow matrix itself, in order to provide a fair comparison.

Every test instance was run on a single computer, with the following specifications: 2 Intel Xeon CPU X5650 (12M Cache, 2.66 GHz, 6.40 GT/s Intel QPI, 6 cores, 12 threads) processors, 24 GB DDR3 1333 MHz RAM and 512 GB of storage. Each was run 33 times to avoid disparities in execution time due to uncontrolled factors such as other processes running on the same machine.

The input dataset chosen for the performance analysis was offered by a partner company, and is different from the one used on the bus route study, consisting of 8.5 million records, with around 49 thousand unique users. Each data record was collected through apps installed on users mobile phones, capturing geographic coordinates with timestamps.

As our implementation uses the parallelism leveraged by Apache Spark, we tested it using different numbers of execution threads, varying from 1 to 16. Table 2 shows the result of the comparative evaluation, with ”skmob” referring to Scikit-Mobility and ”original” to our own tool. While Scikit-Mobility took around 15 minutes on average to perform the mobility flow calculation, our tool completed the task in about 2 minutes with 8 or 16 threads. This shows the

Table 2. Execution time for flow extraction.

		Execution time (minutes)			
Solution	**Threads**	**Min.**	**Avg.**	**Max.**	**StdDev.**
skmob	-	15.1	15.2	15.3	0.04
original	1	7.6	7.6	7.8	0.05
original	2	4.0	4.7	5.3	0.45
original	4	2.2	2.7	3.1	0.19
original	8	1.6	1.9	2.2	0.21
original	16	1.5	1.9	2.3	0.21

efficiency of our implementation and how parallelism can outperform traditional methods.

4 User-Centered Analysis of Safe Bus Routes

This section presents the user-centered assessment of the impact of using a safe route solution, more specifically the work of Almeida *et al.* (2022), on the movement of bus users. So far, existing solutions focus on evaluating the route itself, without concern about how changes in routes affect passengers. For the user-centered analysis conducted here, real data from thousands of users was used to extract mobility flow, and the impact on them was revealed.

4.1 Contextualization

A city, in general, has several distinct bus lines, each of which is responsible for covering certain metropolitan regions. Each line has a number of bus stops, which are mandatory and sequential places where the bus of that line must pass through to board and disembark passengers. Between two bus stops, a bus can follow different routes, such as the shortest, the safest, or the fastest. In addition to the bus stops, each street corner through which a bus needs to pass is also considered a vertex of the city road network built and used in this work.

Traditionally, between two bus stops, the bus tends to take the shortest route, that is, considering the limitations of city roads, the one that travels the shortest distance between its origin and destination bus stops, i.e., bus stops in sequence. However, the shortest route is not necessarily safe, that is, one that avoids areas with high crime rates. In this context, the citizen is often exposed to violence both inside vehicles and at the bus stops. Therefore, an alternative approach would be to use safe routes, which avoid criminal areas as much as possible.

4.2 Construction of Safe Routes

The work of Almeida *et al.* (2022) has a flexible scheme to find criminal regions, as well as a function capable of summarizing criminal data to calculate the safety score of a route. After identifying criminal regions, the considered safe routes solution also tries to merge the location of certain bus stops that are in highly dangerous areas, with others that are close to them but at less dangerous places. A bus stop is only merged with another if they are within a viable distance from each other. The use of an already existing bus stop avoids investment in infrastructure for the bus stops – e.g., coverage, monitoring panels, etc.

When changing the route of a given bus line, or relocating one of its bus stops, passengers may be affected by the distance to be traveled, since the safest route is not always the shortest one. If we calculate the difference in length between the safest and the shortest routes, considering the complete bus line – i.e., from its first to its last bus stop –, we find the impact of the safe route solution on the line as a whole. But this does not necessarily reflect on the impact perceived by the passengers themselves, as each of them has different boarding and disembarking points, which rarely coincide with the starting and ending points of the lines. In other words, a given line may have sections with more passengers boarding and disembarking. Therefore, certain segments of the route are more relevant to the population than others, and so mobility flow is used to identify these segments, as described in Section 4.3.

In order to create safety-based route options, the first step is to define the criteria for identifying unsafe regions. For this, after preparing the criminal datasets used and leaving them with only the useful attributes in the process, such as the crime heading, date, time, latitude, and longitude, it is already possible to build clusters through the geolocation of the registers. To generate these clusters across the city, the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm was used. It has two parameters: $\varepsilon = 100m$ (maximum radius between neighbors) and $\upsilon = 36$ (minimum number of neighbors for a cluster to be valid). These values vary according to the dataset used and the size of the city. After carrying out numerous empirical combinations, these were the ones that were most consistent with our model and scenario, in the city of São Paulo.

In this way, after modeling and defining the safety criteria, it is possible to define the strategy for identifying and classifying safe routes and combine it with criminal information. Almeida *et al.* (2022) present in their work formal definitions for the correct understanding of important concepts. For this work, only a summarized overview will be presented.

Definitions: In the public transportation data used, the N available **bus lines** are represented by an ordered list of **bus stops**. Therefore, originally there are no routes that connect such bus stops. The **routes** are the paths that connect all pairs of bus stops of a bus line. The journeys between two pairs of adjacent bus stops are called **route segments**. For a single bus line, there may be different types of routes, such as: shortest, safest, and least safe, each of them having different route segments in their construction.

Equation 1, first defined in (Babu and Viswanath, 2008) was used by Almeida *et al.* (2022) to calculate the safety level of the created routes. This equation, called *Parzen Window*, is a non-parametric way to estimate the probability density function of a random variable.

The term **route segment** is important because Equation 1 is applied over each route segment constructed between pairs of bus stops. As a result, each segment receives a score k, which, when added to the complete route, returns a total value of K. This total value represents the safety index of the route, so the lower it is, the safer the route can be considered. In this way, it is possible to classify among all the routes created for each bus line which ones are the safest and the least safe.

In Equation 1, m represents the number of street corners between the begin and ending of a segment, σ is the standard deviation of the number of crimes per *cluster*, x_a represents the distance of each vertex to the center of the nearest *cluster*, and x represents the distance from the edge of the closest *cluster* to its center.

$$k_{(p_i,p_j)} = \frac{1}{m}\sum_{a=1}^{\mathrm{m}} \frac{1}{\sqrt{2\pi\sigma}} exp\left(-\frac{x_a - x}{2\sigma^2}\right) \quad (1)$$

To complement the understanding of the Equation 1, Figure 3 shows how each stage of the route is related to the nearest criminal cluster. Therefore, the points in blue represent the street corners of the route, and the arrows, the city streets. The polygon highlighted in light red is used to represent a criminal cluster. Finally, the red dot indicates where the center of the cluster is located. From this, two distances are calculated, **A** and **B**. **A** represents the distance of the analyzed vertex in relation to the nearest cluster center, and **B** represents the distance from the nearest edge of the analyzed vertex to the center of the criminal cluster. In Equation 1, **A** is equivalent to x_a and **B** is equivalent to x. However, in practice, the formed clusters do not have the regular format illustrated in Figure 3, so it is necessary to identify the closest edge to characterize the proximity to the center of the cluster, as done by Ladeira *et al.* (2019).

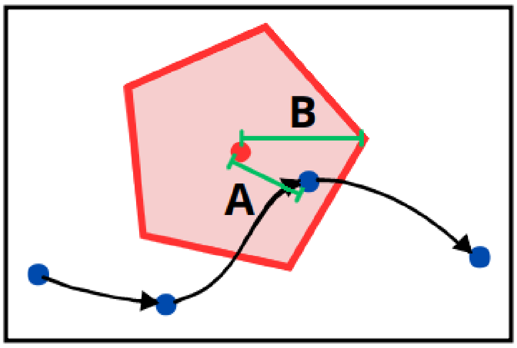
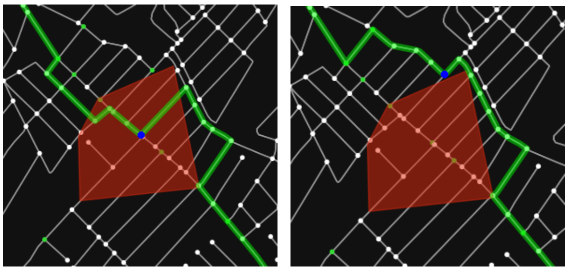


Figure 3. Example of applying the equation to a route section.

With this equation, it is now possible to execute the aforementioned method and obtain the safest routes. This is done by applying it iteratively to each street corner of the route and summing the partial results in the K variable, which in the end represents the route safety index. The safest route is the one with the lowest K. As a comparison parameter, the shortest route of each bus line was generated to serve as a reference, representing the way buses currently travel, since the exact route of the lines is not available in the datasets. Thus, at the end of this part, we have the shortest and safest route between pairs of consecutive bus stops of all lines.

However, according to Almeida *et al.* (2022), this solution alone did not prove to be fully efficient in terms of safety, and there may be cases in which changing the route between two bus stops does not make much difference, since the main problem is at the bus stop located inside a criminal region. Thus, in addition to changing the route, Almeida *et al.* (2022) added a strategy to solve this problem, proposing the relocation of bus stops as an alternative to increasing route safety.

This strategy adopted the following criteria: a bus stop can only be relocated if it is inside a criminal cluster and there is another bus stop outside the cluster, but within a radius of 250 meters. If this condition is met, it is also verified if there is a route (round trip) between the current bus stop and the candidate one. If such route exists and its length is less than or equal to 1000 meters, this candidate bus stop becomes valid and able to replace the current one. In Figure 4, it is possible to see how this process occurs. It can be seen in 4(a) that even if the algorithm searches for safer routes, the fact that the bus stop (blue marker) is located inside the cluster means that the route has to go through it, reducing safety and exposing the passenger to a greater risk. In 4(b), it is possible to work around the problem since there is a candidate bus stop (blue marker) that satisfies the required conditions. In this way, it is noticed that when relocating the bus stop to a region outside the cluster, the constructed route avoids the cluster as a whole, increasing its safety.



(a) Original bus stop within a criminal cluster.

(b) Bus stop relocated to a safe region

Figure 4. Example of a bus stop replaced by a valid candidate.

It is important to state that other strategies can be used to select safer routes, such as spatial operations. Still, it is necessary to formulate how to compute how safe a route is. The use of Equation 1 is a good alternative that was proven efficient by Ladeira *et al.* (2019). In addition, the possibility to change a bus stop from one place to another is a novelty that also contributes to making the routes even safer.

Thus, given these alternatives for building safe routes, using clustering and relocation, different possibilities and combinations of routes can be created. Two of these scenarios, presented in Table 3, were created and implemented to analyze the case studies and generated results. Essentially, Scenario 1 represents the current model for public transport and Scenario 2 is the solution of Almeida *et al.* (2022), with all the safety issues mentioned. These scenarios will be used to reference the results later.

Table 3. Implemented scenarios.

Scenario	Displacement Type	Uses Relocation?
1	Shortest Route	No
2	Safest Route	Yes

4.3 Flow and line segments

As described in Section 3, after completely processing input data with the implemented tool, a flow matrix is calculated and each of its cells can be identified by an origin and a destination census sector. The next step consists of searching a bus line segment that connects each of those origin-destination pairs.

Each bus line b has an identifier id_b and an ordered list of bus stops $bs_1, ..., bs_n$. Each bus stop has a geographical location, through which it is possible to associate a corresponding census sector, thus transforming the list of bus stops into a list of sectors $s_1, ..., s_n$ where the bus line b travels through.

For each occurrence of two distinct census sectors s_i and s_j in the bus stops of a line b, if $i < j$, i.e., if s_i is present on the list before s_j, this line b is marked as usable for traveling from s_i to s_j. Also, we keep track of how many bus stops have to be crossed for this travel.

Performing this calculation for every pair of origin-destination sectors extracted from mobility flow, it is possible to choose the bus line with the least amount of intermediate bus stops. Movements from and to the same sector were discarded as either data collection noise or too short to use public transportation.

For each line segment selected in the previous step, the shortest and the safest route were built. These two types of routes were compared in terms of total distance traveled, number of changed vertices (i.e., street corners of stops), and number of impacted people.

5 Results

In this section, we present the results with the objective of assessing how changing routes to make them safer affects the passengers. We consider the two scenarios from Table 3, where Scenario 1 considers the original route, which is assumed to be the shortest one, while Scenario 2 considers the safest route selected by the process described in Section 4.2.

Before presenting the results obtained by applying the steps detailed in Section 4, we describe the data used.

5.1 The Data

5.1.1 Bus Routes and Crime

For the bus routes, an open dataset from the *Interscity*[1] platform was used. It is composed of 2,089 eligible bus lines in the city of São Paulo/SP, with an average of 43 stops per line, a minimum of 5 and a maximum of 132, in addition to a median of 40 stops.

For the construction of the clustering model of the criminal areas, an open dataset, available on the *Data World*[2] platform, was also used. This has approximately 945,000 criminal records in the city of São Paulo during the year 2014. However, the crimes with no geolocated information were removed, resulting in 732,000 crimes. Finally, a crime heading filter was applied to select crimes more consistent with the public transportation scenario. Of these, only items such as theft, simple homicide, stealing, and drug traffic were maintained. In this way, the final dataset kept approximately 522,000 crimes. With the criminal data filtered and cleaned, it was possible to apply the clustering techniques. Table 4 presents some information on the resulting clusters. A visual example of how a criminal cluster is represented in the city graph can be seen as the red area in Figure 4.

Table 4. Cluster statistics were created for filtered crimes.

Clusters	Formed by all crimes filtered
Amount	2,030
Average Area (m²)	54,397.47
Average crimes	103.48
Standard deviation of crimes	164.90

The main results obtained in Almeida *et al*. (2022), which evaluates the entire lines, are presented in the following as part of data characterization.

An important factor when proposing improvements to an existing route solution is the distance that will be added to the route to obtain safer routes. In Figure 5 it is possible to observe the distribution of distances. On average, there are 20,076.69 meters for scenario 1 and 22,399.46 meters for scenario 2. In percentage terms, it represents approximately only a 12% increase.

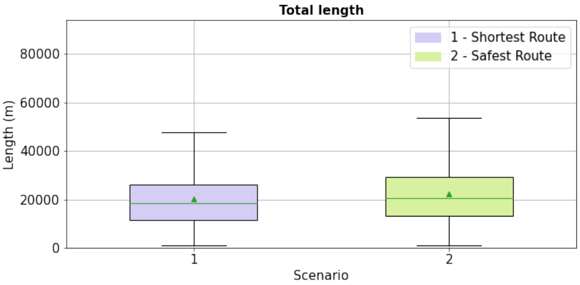


Figure 5. Total length between bus routes for both scenarios.

Regarding the benefits obtained with this addition, it is clear that with the strategy used, buses need to travel less in unsafe regions, reducing the number of vertices (street corners) of the route on which the bus travels, located in criminal regions, from 27.93% (scenario 1) to 22.01% (scenario 2). This difference can be seen in Figure 6.

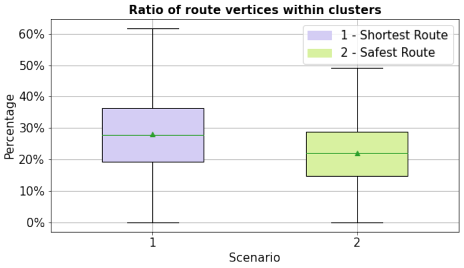


Figure 6. Percentage of route vertices located within criminal clusters.

[1] https://interscity.org/open_data/

[2] https://data.world/maszanchi/boletins-de-ocorrencia-sp-2014

In addition to this result, it is also important to evaluate only the bus stops located within criminal clusters. With the original solution (scenario 1), on average, 25.46% of the bus stops of a route were located within a criminal cluster. With the solution presented, this number dropped to 8.12%, significantly reducing the exposure of passengers waiting for their bus in a criminal area. In practical terms, it represents an average of 7.4 bus stops of a line removed from criminal clusters. Considering that the average number of bus stops is 43, this represents approximately 17% of the entire route.

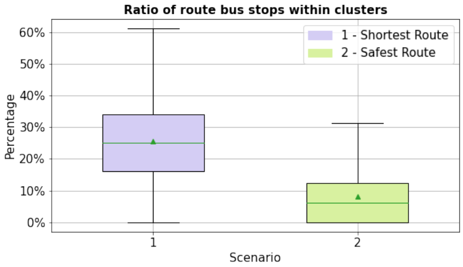


Figure 7. Percentage of bus stops located within criminal clusters.

5.1.2 Displacements

For a user-centered analysis, real geolocation data were provided by a private company under a confidentiality agreement. Data from 356,725 users were made available, generating a total of 11,351,545 records over 6 months of 2021. Most users contain around 200 records, which is a reasonable amount to extract knowledge.

Mobility flow was extracted from these data points, resulting in 51,240 unique pairs of regions. The average flow among those pairs is 3.68 movements, the highest value is 292, and the lowest value is 1.

The three most important sectors, through which a larger number of people move, are in the central region of the city, as shown in Figure 8. In addition, two other import sectors are located far from the central region, as shown in Figure 9, which correspond to a region where the *Neo Química Arena* stadium is located, together with the *Shopping/Metrô Itaquera*, two points of great interest in the city.

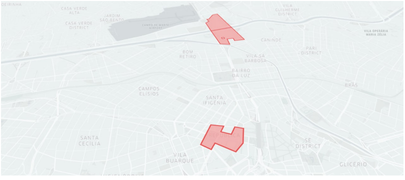


Figure 8. Sectors located in the central region

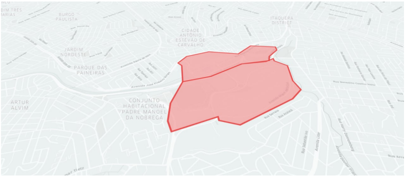


Figure 9. Sectors located further away

5.2 Traveled Distance and Number of Vertices

The distance traveled measures the length of the segment between the origin and destination points of the flow matrix, and not the route as a whole. In other words, it is the distance traveled by the passengers on their journey. We also calculate the extra distance traveled, which is given by subtracting the size of the safest route from the size of the shortest one, the former being always greater than or equal to the latter. The number of vertices, on the other hand, consists of the number of stopping points and corners that constitute the route of that specific bus line segment.

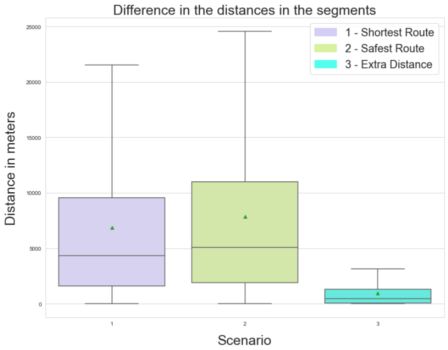


Figure 10. Distribution of the distance of the segments

We can observe the distance of the sections in Figure 10. In this figure, the outliers have been removed for better visualization. As expected, the safest routes are longer than the shortest ones, with the median of the shortest routes being below 5km, while close to 5km for the safest ones. However, if we look at the extra distance traveled, we see that most of the increases occur in the 1km range. That is, most segments have an increase of a maximum of 1km when the route becomes safer, with little impact on users.

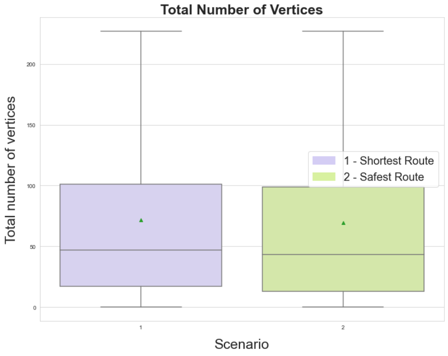


Figure 11. Size of segments and the total number of vertices.

Regarding the total number of vertices, Figure 11 shows that the increase in vertices is not as significant as the distance. Thus, the impact for users on the path in terms of the number of corners is practically none.

5.3 Changes in Vertices and Bus Stops

When making a route safer, either by changing the path or by relocating its bus stops, the number of vertices (i.e., corners), as well as bus stops, may change. To measure the impact of

Table 5. Same and different vertices or bus stops.

Measures	Different Vertices	Different Bus Stops	Same Vertices	Same Bus Stops
Mean	15.49	3.76	45.75	6.39
Minimum	0.00	0.00	0.00	0.00
Maximum	547.00	83.00	493.00	90.00
1º quartile	0.00	0.00	1.00	0.00
Median	4.00	0.00	24.00	1.00
3º quartile	13.00	2.00	66.00	9.00
Standard Deviation	35.72	8.65	59.51	10.45

such changes, the number of stops and vertices of the shortest and the safest routes for each segment was compared, as can be seen in Table 5.

It can be observed that, on average, few vertices are changed between the shortest route and the safest one, and the great part is maintained or presents little modification. We can also observe that, in certain segments, there is no change at all, since the shortest route is already the safest. For 75% and 50% of the segments, the change of vertices is at most 13 and 4, respectively. Regarding the bus stops, we see that 75% of the segments have a maximum of 2 changed bus stops, that is, it maintains the integrity of the original route, not significantly affecting passengers.

5.4 Number of Affected Users

To understand the number of people impacted by the changes in routes, we check the mobility flow matrix, since each record indicates the aggregation of individual trips.

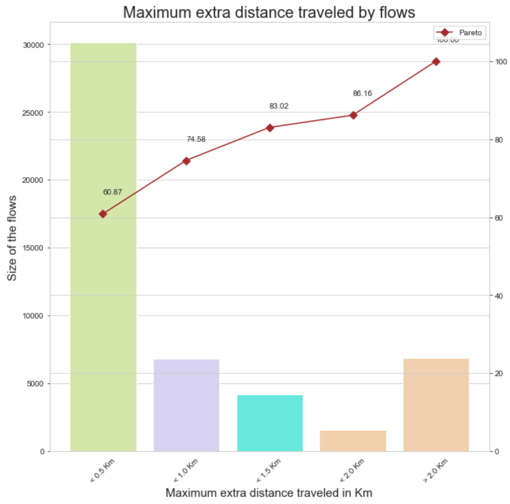


Figure 12. Maximum extra distance traveled by the number of people.

Figure 12 shows the extra distance traveled per segment regarding the number of people on it. It can be noticed that the segments with greater extra distance affect a small number of people, while most users experience little impact. In fact, 74.58% of users were affected in less than 1km and 86.16% in less than 2km. It is worth noting that this figure includes all segments and that some of them were greatly affected when the safer route is used, traveling up to extra 2km. However, they represent a minimal number of segments and affected users.

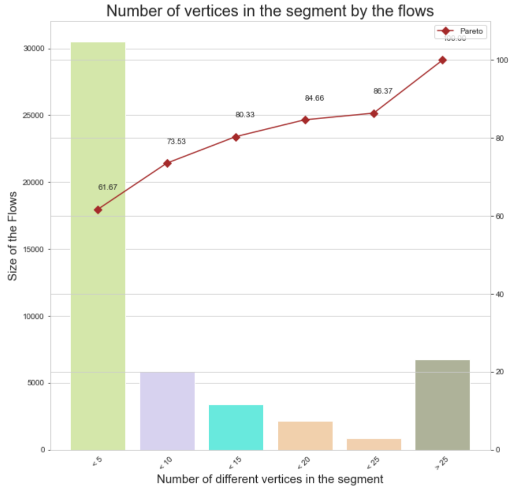


Figure 13. Number of different vertices by the number of people.

Figure 13 shows the result of the analysis regarding the number of different vertices in the segment by the number of people. Once again, the segments with the highest number of people were the least altered routes, while the routes where there are significant differences are those with the least number of people. As can be seen, only a small part (around 13%) of the passengers travel through segments with more than 25 changes in the number of vertices.

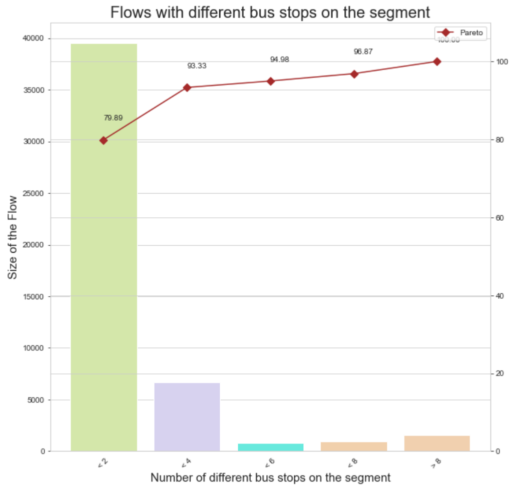


Figure 14. Number of relocated bus stops by the number of people.

Finally, in Figure 14 we can observe the number of people who were affected by the relocations, and the number of bus stops that were relocated in the segment. It can be seen that most of the significant changes affect a small portion of the passengers, while the least impact changes are those that affect the most relevant displacements. In fact, 79.89% of the passengers had their routes with one or zero changes in the bus stops, while 94.98% of them had their routes affected by less than 6 changes in the bus stops.

6 Conclusion and Future Work

This work presented a user-centered analysis of the application of a safe route solution for public bus transport in the city of São Paulo/Brazil. The solution used was proposed by Almeida *et al.* (2022) and has only been evaluated under transportation metrics, such as integral line length and number of replaced stops. In this work, we went further and evaluated how passengers are affected by route changes by considering real data from thousands of users.

It has been shown that changing a bus route to make it safer increases its length. However, the impact on the passengers themselves is not significant, and most of the route segments have their length increased in less than 1 km. If we consider the benefits of having a safer route, this distance is acceptable.

For this analysis to be carried out, a tool was developed to extract flow from large volumes of mobility data. This tool was able to reduce the processing time from the 15 minutes required by Scikit-mobility, a solution from the literature, to 2 minutes when 8 parallel processes are used. This tool, which can be used in other scenarios, is also a contribution of this work.

Regarding future works, we can list the need to expand the implemented tool with new functionalities related to the extraction of mobility flow or to facilitate the calculation with new metrics. It is also expected more case studies to be carried out, taking into account other cities.

Acknowledgment

The authors thank the Brazilian agencies CNPq, Fapemig, and CAPES for their financial support.

Declarations

Authors' Contributions

All authors contributed to the writing of this article, read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Availability of data and materials

Data can be made available upon request.

References

Almeida, V. G. J., Silva, T. R. M. B., and Silva, F. A. (2022). Se for, vá na paz: Construindo rotas seguras para veículos coletivos urbanos. In *Anais do Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, Porto Alegre, RS, Brasil. SBC. DOI: 10.5753/sbrc.2022.221978.

Babu, V. S. and Viswanath, P. (2008). An efficient and fast parzen-window density based clustering method for large data sets. In *International Conference on Emerging Trends in Engineering and Technology*, pages 531–536. DOI: 10.1109/ICETET.2008.166.

Barbosa, H., Barthelemy, M., Ghoshal, G., James, C. R., Lenormand, M., Louail, T., Menezes, R., Ramasco, J. J., Simini, F., and Tomasini, M. (2018). Human mobility: Models and applications. *Physics Reports*, 734:1–74. Human mobility: Models and applications. DOI: 10.1016/j.physrep.2018.01.001.

Galbrun, E., Pelechrinis, K., and Terzi, E. (2016). Urban navigation beyond shortest route: The case of safe paths. *Information Systems*, 57:160–171. DOI: 10.1016/j.is.2015.10.005.

Graser, A. (2019). MovingPandas: Efficient structures for movement data in Python. *GI_Forum – Journal of Geographic Information Science 2019*, 7:54–68. DOI: 10.1553/giscience2019$_01_s$54.

Guo, D., Zhu, X., Jin, H., Gao, P., and Andris, C. (2012). Discovering spatial patterns in origin–destination mobility data. *Transactions in GIS*, 16. DOI: 10.1111/j.1467-9671.2012.01344.x.

Iqbal, M. S., Choudhury, C. F., Wang, P., and González, M. C. (2014). Development of origin–destination matrices using mobile phone call data. *Transportation Research Part C: Emerging Technologies*, 40:63–74. DOI: 10.1016/j.trc.2014.01.002.

Kon, F., Ferreira, E., Souza, H., Duarte, F., Santi, P., and Ratti, C. (2021). Abstracting mobility flows from bike-sharing systems. *Public Transport*. DOI: 10.1007/s12469-020-00259-5.

Ladeira, L., Souza, A., Pereira, G., Silva, T. H., and Villas, L. (2019). Serviço de sugestão de rotas seguras para veículos. In *Anais do XXXVII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, pages 608–621, Porto Alegre, RS, Brasil. SBC. DOI: 10.5753/sbrc.2019.7390.

Liu, Q., Kumar, S., and Mago, V. (2017). Safernet: Safe transportation routing in the era of internet of vehicles and mobile crowd sensing. *2017 14th IEEE Annual Consumer Communications & Networking Conference (CCNC)*. DOI: 10.1109/ccnc.2017.7983123.

Martins, T., Lago, N., Zambom Santana, E. F., Telea, A., Kon, F., and Souza, H. (2021). Using bundling to visualize multivariate urban mobility structure patterns in the são paulo metropolitan area. *Journal of Internet Services and Applications*, 12:6. DOI: 10.1186/s13174-021-00136-9.

Mata, F., Torres-Ruiz, M., Guzmán, G., Quintero, R., Zagal-Flores, R., Moreno-Ibarra, M., and Loza, E. (2016). A mobile information system based on crowd-sensed and official crime data for finding safe routes: A case study of mexico city. *Mobile Information Systems*, 2016:1–11. DOI: 10.1155/2016/8068209.

Montoliu, R., Blom, J., and Gática-Pérez, D. (2011). Discovering places of interest in everyday life from smartphone data. *Multimedia Tools and Applications*, 62:179–207. DOI: 10.1007/s11042-011-0982-z.

Moreno-Monroy, A., Lovelace, R., and Ramos, F. (2017). Public transport and school location impacts on educational inequalities: Insights from são paulo. *Journal of Transport Geography*, 67. DOI: 10.1016/j.jtrangeo.2017.08.012.

Pappalardo, L., Simini, F., Barlacchi, G., and Pellungrini, R. (2021). scikit-mobility: a Python library for the analysis, generation and risk assessment of mobility data. DOI: 10.48550/arXiv.1907.07062.

Santos, F. A., Rodrigues, D. O., Silva, T. H., Loureiro, A. A. F., Pazzi, R. W., and Villas, L. A. (2018). Context-aware vehicle route recommendation platform: Exploring open and crowdsourced data. In *2018 IEEE International Conference on Communications (ICC)*, pages 1–7. DOI: 10.1109/ICC.2018.8422972.

Santos, F. A., Rodrigues, D. O., Silva, T. H., Loureiro, A. A. F., and Villas, L. A. (2017). Rotas veiculares cientes de contexto: Arcabouço e aná lise usando dados oficiais e sensoriados por usuários sobre crimes. In *Anais do XXII Workshop de Gerência e Operação de Redes e Serviços*, Porto Alegre, RS, Brasil. SBC. Available at:`https://sol.sbc.org.br/index.php/wgrs/article/view/2591`.

Tompson, L., Partridge, H., and Shepherd, N. (2009). Hot routes: Developing a new technique for the spatial analysis of crime. *Crime Mapping: A Journal of Research and Practice*, 1(1):77–96. Available at:`https://discovery.ucl.ac.uk/id/eprint/20057/`.

Yu, J., Zhang, Z., and Sarwat, M. (2019). Spatial data management in Apache Spark: The GeoSpark perspective and beyond. *Geoinformatica*, 23(1):37–78. DOI: 10.1007/s10707-018-0330-9.

Zaharia, M., Chowdhury, M., Das, T., Dave, A., Ma, J., McCauly, M., Franklin, M. J., Shenker, S., and Stoica, I. (2012). Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing. In *9th USENIX Symposium on Networked Systems Design and Implementation (NSDI 12)*, pages 15–28, San Jose, CA. USENIX Association. Available at:`https://www.usenix.org/system/files/conference/nsdi12/nsdi12-final138.pdf`.