Accelerometer and Gyroscope Synthetic Data Calculation based on Driver Smartphone GPS

Varvara Shushkova[1], Alexey Kashevnik[2], Larisa Bakeeva[1]
[1]ITMO University, St. Petersburg, Russia
[2]SPC RAS, St. Petersburg, Russia
shushkova.vv@gmail.com, alexey.kashevnik@iias.spb.su, larisabakeeva@yandex.ru

***Abstract*—Determining driver behavior during road traffic is an actively researched task, with numerous systems aimed at identifying dangerous maneuvers, detecting driver fatigue, and monitoring the driver's condition. The simplest way to obtain data for studying driver behavior is to collect vehicle movement data using a smartphone that utilizes built-in sensors such as GPS, accelerometers, and gyroscopes. These sensors provide comprehensive data on the car's speed, location, acceleration, and angles of deviation, enabling in-depth research of driver behavior. However, it is not always possible to obtain accelerometer and gyroscope data. This paper proposes an algorithm for calculating accelerometer and gyroscope synthetic data based on GPS that provide possibilities to classify driver behaviour based on these synthetic data for devices that do not have accelerometer / gyroscope sensors. The method involves constructing quaternions to describe rotations, followed by conversion to Krylov-Euler angles. We evaluated proposed method on the dataset that has been captured in-the-wild during driving vehicles by several drivers. Developed software that implements the proposed method is available as open-source.**

I. Introduction

The task of detecting driver behavior during road traffic is actively being addressed today. Numerous systems aim to detect dangerous maneuvers [1] and detecting driver state [2], [3], [4].

The simplest way to obtain data for studying driver behavior on the road is to collect vehicle movement data using a mobile device. Modern smartphones are equipped with numerous built-in sensors, including GPS, accelerometers, gyroscopes, magnetometers, etc. With these sensors, one can determine the vehicle's speed, location, acceleration, and angles of deviation. This data set provides a wide scope for researching driving behavior and analyzing road situations.

One possible approach to solve such problems is by tracking the vehicle's path and predicting subsequent maneuvers. In this case, the input data consists of GPS data, which can be used to reconstruct the driver's route and attempt to train a system to predict future actions.

In our previous research [5] we addressed the task of investigating the correlation between the driver's experienced emotion and the type of maneuver being performed. This paper enhance our previous work and uses captured in-the-wild dataset DriverMVT (In-Cabin Dataset for Driver Monitoring including, Video and Vehicle Telemetry Information) [6]. GPS, gyroscope, and accelerometer data are used to determine the type of maneuver. The driver's emotions are identified through video recordings of their face. The built-in phone sensors was used for collecting accelerometer and gyroscope data. However, collecting accelerometer and gyroscope data is not always feasible. To expand the scope of research and increase the number of test drivers, this work aims to calculate synthetic accelerometer and gyroscope data based on GPS. This method allows for a broader study and facilitates the inclusion of more test drivers, even when specific sensor data is unavailable.

Accelerometer data measures the vehicle's acceleration along three perpendicular axes (x, y, and z). It provides information on the vehicle's speed changes and direction of movement. Gyroscope data, on the other hand, measures the angular velocity around these three axes. It captures the vehicle's rotational movements, helping to understand the orientation and maneuvering of the vehicle. Together, these data provide a comprehensive view of the vehicle's dynamics and the driver's behavior. With accelerometer and gyroscope data, it is possible to construct the movement trajectory and determine whether a maneuver is aggressive.

This paper describes a method that allows for the calculation of accelerometer and gyroscope data based on GPS data. It also outlines techniques for visualizing and validating the obtained values. The implemented method is available as open-source https://github.com/shushkova/gps2gyro.

The rest of the paper is organized as follows. Section 2 provides an overview of existing approaches to processing GPS, accelerometer, and gyroscope data. Section 3 describes the proposed method for converting GPS data into gyroscope and accelerometer data, outlines the data visualization method, and presents the methodology for testing the obtained results. In Section 4, the methodology from the previous section is applied to a real dataset. Section 5 summarizes the results, discusses future directions, and outlines limitations.

II. Related work

This section describes the data from GPS, gyroscope, and accelerometer sensors, and their applications in driver behavior detection systems. It also reviews existing methods for converting accelerometer and gyroscope data.

A. Driver behavior monitoring systems

Detecting driver behavior is a popular task today. Authors approach this problem in various ways and propose different algorithms to enhance road safety. One of the existing approaches is the use of GPS, accelerometer, and gyroscope data to detect anomalies in driver behavior.

There are two types of gyroscope: mechanical and electrical. In the context of analyzing vehicle motion, particularly with smartphones, data from electrical devices are used. Mechanical gyroscopes have a fixed z-axis pointing north, while electrical devices have built-in axes that can be oriented in different ways. Smartphones, often used in cars, provide a convenient platform due to their embedded sensors, such as accelerometers and gyroscopes, which provide valuable data for monitoring and analyzing driver behavior.

In the paper [7], the authors calculate the Trajectory Aggressiveness Indicator (TAI) based on GPS data to identify aggressive driving behavior. This indicator ranges from 0 (no aggressive behavior) to 100 (extremely aggressive behavior). To improve accuracy, additional factors such as weather and road conditions are also considered in the calculation of this indicator.

Another study [8] classifies driving styles for public transport. The authors propose a pattern recognition approach to automatically classify driving styles without expert evaluation, utilizing accelerometer data from repeated routes driven in different styles. By employing statistical features of 3-axis accelerometer signals as classifier inputs, the study achieved 100% precision in distinguishing between aggressive and normal driving styles on the same route.

Authors of the paper [9] also utilize accelerometer and gyroscope data embedded in mobile devices to detect the type of maneuver being performed. Based on accelerometer and gyroscope data, the start and end points of driving events are identified. Then, the accelerometer and gyroscope data during the maneuver are compared with reference values for various maneuvers to determine the type of maneuver being performed.

To address the issue of driver identification and verification in car-sharing services, the authors of [10] utilize accelerometer data. The paper proposes a recurrent neural network architecture. The results achieved by the authors are comparable to those in studies that use not only accelerometer data but also gyroscope and magnetometer data for similar tasks [11].

Another way to use GPS, accelerometer, and gyroscope data is proposed in [12]. In this research, the authors create a new system for real-time road quality estimation. To reduce noise in the accelerometer and gyroscope data, they use a Low-Pass Filter, which effectively removes high-frequency noise by applying a suitable threshold to the filtered signal. As a result, they developed a classifier that predicts road quality based on this data. GPS data is used for path visualization.

To sum up, the reviewed approaches underscore the effectiveness of leveraging sensor data, such as GPS, accelerometer, and gyroscope information, for detecting and analyzing driver behavior. These methods, including calculation of aggressiveness scores, classification of driving styles, and identification of maneuvers, highlight the potential for the developing driver monitoring systems.

B. Approaches to accelerometer and gyroscope data transformation

Gyroscope and accelerometer data are often used to solve human activity classification tasks. To work with data from accelerometer and gyroscope sensors, the authors of the work apply various transformations. The authors [13] propose a descriptor-based approach for activity classification using smartphone sensors, employing features such as histograms of gradients and Fourier descriptors. Their method achieves 97.12% accuracy on the UCI HAR dataset and 96.83% on a physical activity sensor dataset, outperforming existing methods that achieve 96.33

The authors of [14] address the task of classifying human activities (walking, running, ascending, and descending stairs) using accelerometer and gyroscope data. Based on data obtained from mobile devices, the authors build a classifier to determine the type of activity. This work addresses the problem of the unknown position of the mobile device during activity, which consequently leads to difficulties in identifying the axes of the built-in sensors. The authors use Euler angles to rotate to a reference coordinate system. As a result, the accuracy of the prediction system increases by 17%.

In the article [15], an algorithm for generating gyroscope and accelerometer data based on GPS data is described. The author uses quaternions to calculate the angle of rotation of the aircraft and subsequently determine the gyroscope data. A limitation of this article is the synthetic nature of the data. The data is manually generated, with the z-axis oriented to the north, and the data does not contain any gaps, appearing regularly at a fixed time intervals.

Another approach to processing accelerometer and gyroscope data is proposed in [16]. The article discusses a method for improving the accuracy of data obtained from MEMS IMUs. The authors dynamically select the most accurate data with minimal systematic errors, instead of using the traditional Bayesian approach, which combines all data. The proposed Best Axes Composition (BAC) method discards the remaining data, thereby improving trajectory estimation accuracy. The authors extend this approach by adding BAC for accelerometers to the previously developed BAC for gyroscopes and demonstrate that both methods enhance accuracy over different time intervals compared to traditional methods of averaging data from multiple IMUs.

The research [17] addresses the problem of driver identification based on data obtained from mobile phone sensors. To increase the volume of input data from accelerometers and gyroscopes, the authors propose using a Generative Adversarial Network (GAN). To improve the quality of the generated data, the authors suggest applying Discrete Wavelet Transform (DWT) before feeding the data into the neural network for

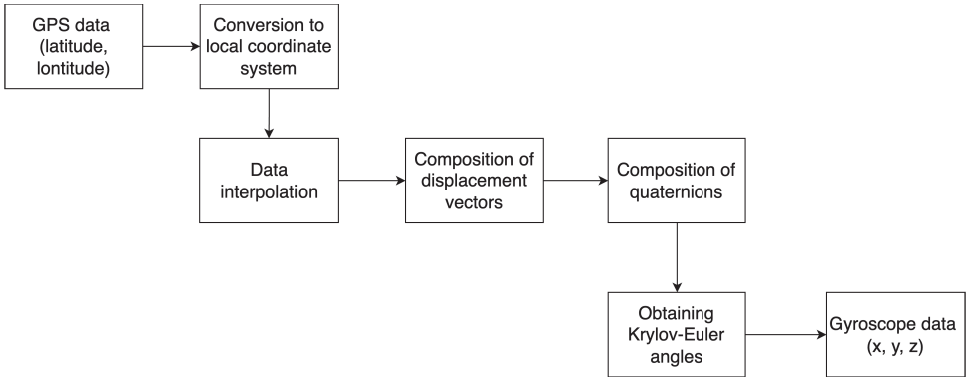

Fig. 1. Gyroscope data calculation based on GPS data

training. As a result of these transformations, precision reaches 98%, and recall is 97%.

Many studies use GPS, accelerometer, and gyroscope data to detect driver behavior, analyze road conditions, and assess human activity. Typically, these researches focus on transforming GPS, gyroscope, and accelerometer data to address specific tasks, employing various methods such as additional filtering, augmentation, and transformation. However, collecting accelerometer and gyroscope data is not always possible. This research proposes an algorithm for solving the reverse problem: estimating accelerometer and gyroscope data based on GPS sensors. The developed method can expand training datasets in several studies, providing practical benefits for improving existing results.

III. Methodology

In this study, the task of calculating accelerometer and gyroscope data based on GPS sensor data is addressed. Accelerometer data contains information about acceleration, while gyroscope data reflects angular velocity.

A. Data preprocessing

The input data used in this study is collected through sensors embedded in Nvidia Jetson Nano devices. The data includes latitude and longitude coordinates. Ideally, new records appeared at a frequency of 0.1 seconds. However, gaps often exist in the data, so interpolation is necessary to create a smooth trajectory and achieve a record frequency of 0.1 seconds. Missing values for latitude, longitude, and speed are filled using linear interpolation.

B. Gyroscope data calculation

We propose our method for synthetic gyroscope data simulation based on GPS (see Fig. 1).

As input, we use latitude and longitude data from the GPS sensor. The first step of the proposed method is to obtain the coordinates in a local coordinate system, where the x and y axes are tangent to the Earth's surface at the origin point of the path, and the z axis is perpendicular to the surface. Conversion to the local coordinate system is necessary to measure linear velocity in meters per second and to obtain angular velocity data. The limitation of the global coordinate system is the difficulty in applying formulas for calculating angular and linear velocities.

The following section applies all formulas to three-dimensional orthogonal systems. It is important to note that the local coordinate systems constructed by the above method for the initial point and any subsequent point do not coincide, as the tangent directions differ at various points on the Earth. However, given that the Earth's radius is much larger than the distances of the routes under consideration, we can analyze the route in the local coordinate system of the initial point.

In our system, the z-coordinate is directed perpendicular to the Earth's surface and corresponds to the altitude above sea level. However, the elevation change data in our task has an accuracy of 1 meter, which is not precise enough to account for minor changes. Furthermore, for the task of detecting the type of maneuver, changes in the z-coordinate are not significant, so these changes are neglected.

Thus, to convert GPS coordinates to a local coordinate system, it is necessary to calculate the x and y coordinates of each point relative to the initial point. Figure 2 illustrates the logic of calculating the coordinates in the local coordinate system of point 2 with respect to point 1 (Points 1 and 2 are obtained using GPS). To achieve this, we take the latitude of point 2 and the longitude of point 1. These are the coordinates of point P. Therefore, the distance between point P and point 2 represents the x-coordinate, while the distance between point P and point 1 represents the y-coordinate.

The distance between GPS points will be understood as coordinates in the local coordinate system. The following formula is used to calculate the distance between GPS points:

$$A = \sin^2\left(\frac{lat_2 - lat_1}{2}\right),$$

$$B = \cos(lat_2) \cdot \cos(lat_1) \cdot \sin^2\left(\frac{lon_2 - lon_1}{2}\right),$$

$$\phi = 2\arcsin\left(\sqrt{A + B}\right),$$

where lat_2, lat_1 are values of latitude in radians, lon_2, lon_1 are values of longitude in radians.

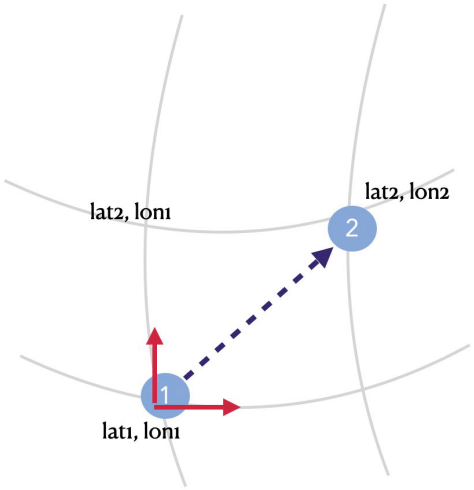

Fig. 2. The logic of calculating coordinates in the local coordinate system

To convert to linear distance the following formula is used:

$$L = \phi \times 6367000 \quad \text{(where 6367000 m is the Earth's radius)}$$

Thus, a set of coordinates is obtained, based on which it is necessary to construct vectors. Each component of the vector is a value in meters, corresponding to the distance traveled at a given moment in time.

Next, it is necessary to calculate the rotation angles of each vector with respect to the previous vector. To determine the angles, the vectors are normalized, since only the rotation angles are needed.

To determine the gyroscope data, the angles of rotation at each moment around each axis, i.e., the Euler angles, have to be determined. These angles can be calculated using quaternions. A quaternion describes the rotation of a vector from one position to another by specifying the axis around which to rotate the vector and the angle of rotation. Next, the method for calculating the quaternion for two vectors will be described.

In Figure 3, two consecutive vectors u_1 and v_1 are shown. These vectors are constructed from the initial points of the path. The angle of rotation from u_1 to v_1 is considered the clockwise rotation angle and is highlighted in red.

The quaternion for this rotation is constructed based on the axis of rotation, which is obtained by the cross product of u_1 and v_1 ($vector = u_1 \times v_1$), and the angle between the vectors , determined by the formula: $angle = \arccos(vector\ u_1 * vector\ v_1)$. Before creating a quaternion for a vector, it is necessary to normalize the vector. With the axis and the angle, we get the following quaternion:

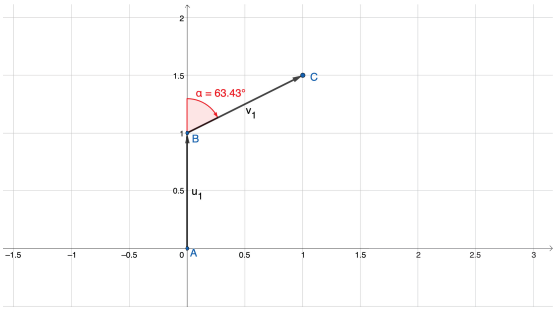

Fig. 3. Rotation angle between vectors

$$q_w = \cos\left(\frac{angle}{2}\right),$$
$$q_x = v_x \cdot \sin\left(\frac{angle}{2}\right),$$
$$q_y = v_y \cdot \sin\left(\frac{angle}{2}\right),$$
$$q_z = v_z \cdot \sin\left(\frac{angle}{2}\right).$$

Then the Euler-Krylov angles are calculated:

$$bank = \arctan\left(\frac{2\left(q_w \cdot q_x + q_y \cdot q_z\right)}{1 - 2\left(q_z^2 + q_x^2\right)}\right),$$
$$altitude = \arcsin\left(2\left(q_w \cdot q_y - q_z \cdot q_x\right)\right),$$
$$heading = \arctan\left(\frac{2\left(q_w \cdot q_z + q_x \cdot q_y\right)}{1 - 2\left(q_y^2 + q_z^2\right)}\right).$$

Thus, the angles in radians of rotation around three axes are obtained when moving from point 1 to point 2. Then, similar actions should be repeated for the following vectors. However, it should be taken into account that after the first rotation, the local coordinate system has changed relative to the initial one. Therefore, it is necessary to rotate the vectors back to work in the initial coordinate system. To do this, the resulting quaternion is constructed, which represents the total rotation at the current time. To get the resulting rotation back the following formula is used:

$$quat\_rotated_w = q_w,$$
$$quat\_rotated_x = -q_x,$$
$$quat\_rotated_y = -q_y,$$
$$quat\_rotated_z = -q_z.$$

In order to apply the inverse quaternion for the vectors, it is necessary to obtain product of two quaternions q_1 and q_2:

$$q_w = q_1.w \cdot q_2.w - q_1.x \cdot q_2.x - q_1.y \cdot q_2.y - q_1.z \cdot q_2.z,$$
$$q_x = q_1.w \cdot q_2.x + q_1.x \cdot q_2.w + q_1.y \cdot q_2.z - q_1.z \cdot q_2.y,$$
$$q_y = q_1.w \cdot q_2.y - q_1.x \cdot q_2.z + q_1.y \cdot q_2.w + q_1.z \cdot q_2.x,$$
$$q_z = q_1.w \cdot q_2.z + q_1.x \cdot q_2.y - q_1.y \cdot q_2.x + q_1.z \cdot q_2.w.$$

Next, apply the inverse quaternion to the vector v. Create a quaternion vq from the vector v:

$$t = q \otimes vq,$$

$$r = t \otimes quat\_rotated.$$

These actions are applied sequentially for all pairs of vectors. Thus, an array of rotation angles along three axes is formed.

To show an example we illustrated the algorithm with a synthetic example on a two-dimensional plane (for the case when a driver makes a turn). There is a set of points in the figure: $A(0,0,0)$, $B(1,0,0)$, $C(1,1.5,0)$, $D(1.5,2.5,0)$, $E(2.5,3.5,0)$, $F(4,2.5,0)$, $G(5,1.5,0)$, $H(6,0,0)$. From this set of points, we obtain a set of displacement vectors for each moment in time, where each component of the vector represents the distance traveled at that moment. We obtain the set of vectors: $a = (0,1,0)$, $b = (1,0.5,0)$, $c = (0.5,1,0)$, $d = (1,1,0)$, $e = (1.5,-1,0)$, $f = (1,-1,0)$, $g = (1,-1.5,0)$. Next, we need to create quaternions for the first two vectors, assuming the z coordinate is zero. The angle between the vectors is found through the scalar product. The following quaternion is obtained:

$$(w,x,y,z) = (0.8507, 0.0, 0.0, -0.5257)$$

Next, the Krylov-Euler angles are found using the formula mentioned above: $(0.0, 0.0, -63.4349)$. After this, it is necessary to perform the inverse rotation of vectors b and c to return to working in the initial coordinate system. We calculate the inverse quaternion of the total rotation at the current moment (in this case, $(w : 0.8507,\ x : 0.0,\ y : 0.0,\ z : -0.5257)$). The inverse quaternion is:

$$(w,x,y,z) = (0.8507, 0.0, 0.0, 0.5257)$$

Now we multiply vectors b and c by the inverse quaternion to return to the initial coordinate system. Vector $\mathbf{b}$ now has the following coordinates: $(x : 0,\ y : 1.0,\ z : 0.0)$. Vector $\mathbf{c}$ has the coordinates: $(x : -0.6708,\ y : 0.8944,\ z : 0.0)$. The quaternion for vectors b and c is $(w : 0.9486, x : 0.0, y : 0.0, z : 0.3162)$. Therefore, the resulting rotation quaternion is $(w : 0.9732, x : 0.0, y : 0.0, z : -0.2298)$.

Further, similar steps are performed iteratively. As a result, the angles of rotation for all initial vectors are obtained as shown in Fig. 5

C. Accelerometer data calculation

To find the accelerometer data, it is necessary to determine the speed based on the coordinates. Acceleration is calculated as the change in speed over unit time. This requires the use of coordinates obtained by converting GPS data into the local coordinate system. The following formulas are used:

$$V_x = \frac{\Delta x}{\Delta t}; \quad V_y = \frac{\Delta y}{\Delta t}; \quad V_z = \frac{\Delta z}{\Delta t}$$

$$a_x = \frac{\Delta V_x}{\Delta t}; \quad a_y = \frac{\Delta V_y}{\Delta t}; \quad a_z = \frac{\Delta V_z}{\Delta t}$$

D. Visualization of the obtained values

To visualize the values, it is necessary to plot the obtained trajectory using the gyroscope and acceleration data. We propose the following algorithm for plotting the trajectory:

Set the initial synthetic vector to $(1,0,0)$. Since angular velocity is measured in degrees per second, multiply the given values by 0.1 seconds to obtain the rotation angles in degrees. Rotate the current normalized vector by these angles to get the new vector. Note that for validation, instead of using the synthetic vector $(1,0,0)$, a vector constructed from the first two GPS points is used. This is necessary to set the correct direction for future turns; otherwise, the trajectory would be rotated relative to the actual path.

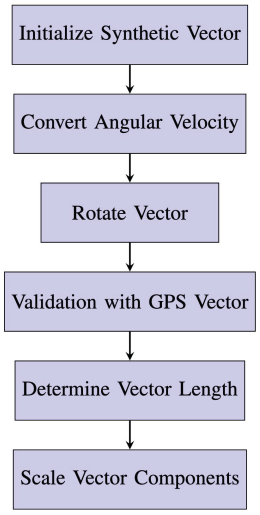

Fig. 4. The Pipeline Visualization

Next, determine the length of the new vector. This length represents the linear velocity. Multiply each component of the vector by the velocities and divide by the L_2 norm of the vector.

E. Validation of the obtained values

To evaluate the accuracy of the proposed method for generating accelerometer and gyroscope data based on GPS data, it is necessary to compare the trajectories constructed from the original GPS data with those obtained from the calculated accelerometer and gyroscope values. The Fréchet distance metric [18] will be used for this purpose.

The Fréchet distance between two curves P and Q is defined as:

$$\delta_F(P,Q) = \inf_{\alpha,\beta} \max_{t\in[0,1]} \|P(\alpha(t)) - Q(\beta(t))\|,$$

where:

- P and Q are the parameterizations of the two trajectories.
- $\alpha(t)$ and $\beta(t)$ are continuous, non-decreasing functions mapping the interval $[0,1]$ to the domain of the trajectories.

- $\|\cdot\|$ represents the Euclidean distance between the points $P(\alpha(t))$ and $Q(\beta(t))$ on the two trajectories.

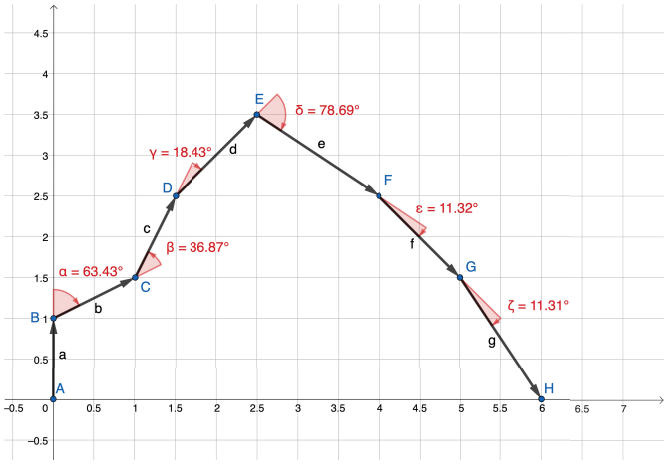

Fig. 5. The angles of rotation for all vectors

IV. Results

The method described above has been implemented using Python 3.10. For testing we used the following [19] dataset, which contains information about drivers' trips. The text files contain GPS sensor output with a frequency of 3-6 measurements per second. Each file describes the movement over a 20-second period during which a road maneuver was performed.For the test, maneuvers performed by two drivers were used, with a total of 5700 trips of 20 seconds each considered.

Performing the mentioned method, firstly, all GPS coordinates were converted to coordinates in a local system with units in meters. Then, we calculated the accelerometer data based on the available GPS data. As a result, speed and acceleration were obtained for each moment in time with a granularity of meters per second.

Next, to compute the gyroscope data, the obtained coordinates were converted to vectors and then transformed into quaternions. Based on these quaternions, Euler angles were derived. Using these rotation angles, angular velocities at each moment in time, i.e., gyroscope data, can be computed.

Then, for each trip, two trajectories were constructed: one based on the computed speed and gyroscope values from the implemented method, and the other based on latitude and longitude from the GPS data. To simplify the validation of the two trajectories, the starting vector for trajectory construction is taken from the vector constructed from the first two points of the actual trajectory. Otherwise, the computed trajectory might be rotated relative to the actual trajectory (see, Fig. 6 - Fig. 11 for different maneuvers).

The maximum length of a single route over the 20-second period is 791 meters, while the minimum length is 2 meters. The average route length is 182 meters. To determine how similar the trajectories of the drivers are, we calculated the Fréchet distance for each pair of trajectories. The average Fréchet distance across all drivers and trips was **5.4** meters. This means that, on average, the trajectories of two drivers are within 5.4 meters of each other. Additionally, for each 20-second segment, a relative metric was calculated as the ratio of the Fréchet distance to the traveled distance. The average value of this metric was 3% across all examined segments and drivers.

For visual interpretation of the results, the actual trajectory was plotted on maps (see, Figs. 7, 9, and 11). The map shows the actual trajectory based on the GPS data. The graphs on the left depict the trajectory calculated using the algorithm, while the graphs on the right show the actual trajectory in meters. As can be seen, the obtained trajectories match in both considered cases, which, together with the small value of the Fréchet distance, clearly demonstrates the correctness and effectiveness of the algorithm.

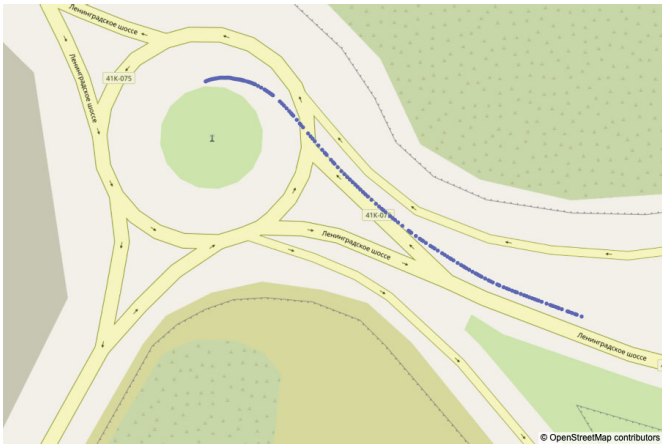

Fig. 6. Real Trajectory on Map: Trip 1

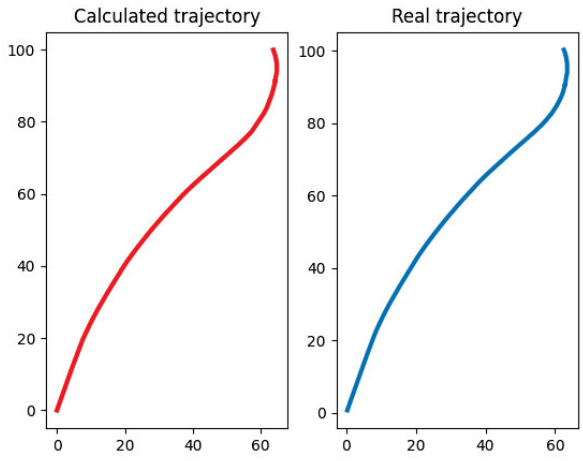

Fig. 7. Calculated and real trajectories in meters: Trip 1

Next, we applied a maneuver type classifier from our previous work [5]. As a result, each maneuver was classified into one of six types: aggressive deceleration, aggressive accelera-

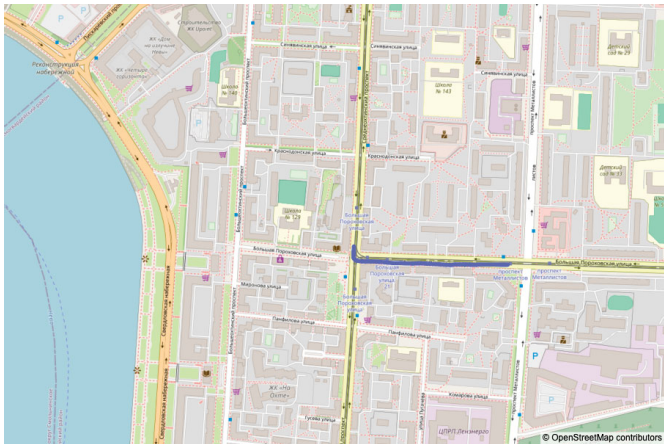

Fig. 8. Real Trajectory on Map: Trip 2

Fig. 10. Real Trajectory on Map: Trip 3

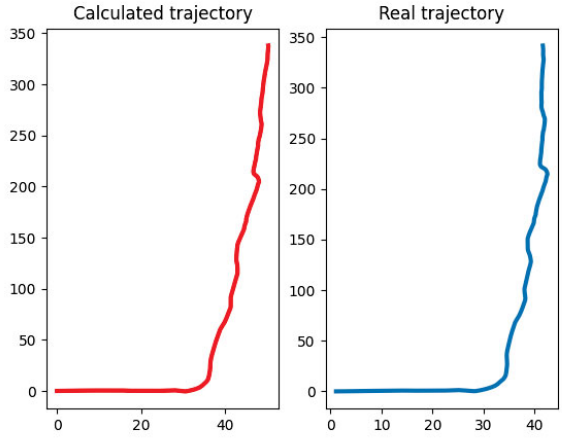

Fig. 9. Calculated and real trajectories in meters: Trip 2

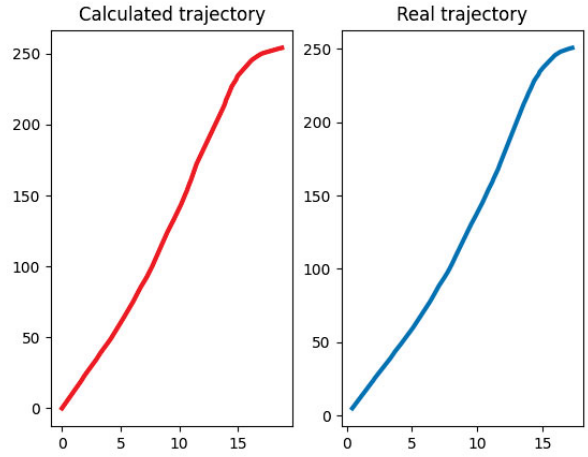

Fig. 11. Calculated and real trajectories in meters: Trip 3

tion, non-aggressive event, aggressive turn left, aggressive turn right, and no maneuver. Table I presents the Fréchet distance values by maneuver type. The metric values for different types of maneuvers differ by no more than 1.5 meters. From experiments we conclude that there is no correlation between the maneuver type and the proposed method accuracy. The algorithm returns a similar error across all maneuver types.

TABLE I
FRÉCHET DISTANCE FOR DIFFERENT MANEUVER TYPES

Maneuver type	Fréchet distance
aggressive deceleration	4.12
aggressive accelerations	5.29
non-aggressive event	5.74
aggressive turn left	6.03
aggressive turn right	5.47
no maneuver	6.35

The implemented method demonstrated high accuracy in calculating accelerometer and gyroscope data based on GPS data. The comparison of the constructed trajectories with the actual GPS data showed significant alignment across various types of road maneuvers. This is confirmed by the low Fréchet distance between the trajectories, indicating a high degree of precision in the proposed method. Therefore, the algorithm successfully addresses the task of motion reconstruction based on limited data, demonstrating its potential applicability in various tasks related to the analysis and monitoring of road maneuvers.

V. CONCLUSION

In this work, a method was developed and implemented synthetic accelerometer and gyroscope data based on GPS sensors. The method includes constructing quaternions to represent rotations, which are then converted into Krylov-Euler angles. The approach has been validated and visualized, and the results are accessible as open-source.

Experiments showed that the Fréchet distance is 5.4 meters across all the data used for testing, which is on average 3% of the trip length. It was also established that the proposed

algorithm maintains its accuracy depending on the type of maneuver performed.

In the future, this opens opportunities for further research in driver behavior analysis using datasets that do not contain the necessary information.

ACKNOWLEDGMENT

This work was supported by the Russian State Research FFZF-2022-0005.

REFERENCES

[1] G. A. M. Meiring and H. C. Myburgh, "A review of intelligent driving style analysis systems and related artificial intelligence algorithms," *Sensors*, vol. 15, no. 12, pp. 30 653–30 682, 2015. [Online]. Available: https://www.mdpi.com/1424-8220/15/12/29822

[2] S. Ansari, F. Naghdy, H. Du, and Y. Pahnwar, "Driver mental fatigue detection based on head posture using new modified relu-bilstm deep neural network," *IEEE Transactions on Intelligent Transportation Systems*, vol. PP, pp. 1–13, 08 2021.

[3] G. Sikander and S. Anwar, "Driver fatigue detection systems: A review," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 6, pp. 2339–2352, 2018.

[4] G. A. Kashevnik A., Lashkov I., "Methodology and mobile application for driver behavior analysis and accident prevention," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 6, pp. 2427–2436, 2019.

[5] V. Shushkova, A. Kashevnik, Y. Rzhonsnitskaya, and A. Blazhenov, "Analysis of the vehicle maneuver and driver emotion: Methodology and results discussion," in *2024 35th Conference of Open Innovations Association (FRUCT)*. IEEE, 2024, pp. 699–706.

[6] W. Othman, A. Kashevnik, A. Ali, and N. Shilov, "Drivermvt: In-cabin dataset for driver monitoring including video and vehicle telemetry information," *Data*, vol. 7, no. 5, 2022. [Online]. Available: https://www.mdpi.com/2306-5729/7/5/62

[7] M. G. Jasinski and F. Baldo, "A method to identify aggressive driver behaviour based on enriched gps data analysis," *Proceedings of the GEOProcessing*, 2017.

[8] V. Vaitkus, P. Lengvenis, and G. Žylius, "Driving style classification using long-term accelerometer information," in *2014 19th international conference on methods and models in automation and robotics (MMAR)*. IEEE, 2014, pp. 641–644.

[9] J. Engelbrecht, M. J. Booysen, and G.-J. Van Rooyen, "Recognition of driving manoeuvres using smartphone-based inertial and gps measurement," 2014.

[10] S. H. Sánchez, R. F. Pozo, and L. A. H. Gómez, "Driver identification and verification from smartphone accelerometers using deep neural networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 1, pp. 97–109, 2020.

[11] J. Ferreira, E. Carvalho, B. V. Ferreira, C. de Souza, Y. Suhara, A. Pentland, and G. Pessin, "Driver behavior profiling: An investigation with different smartphone sensors and machine learning," *PLoS one*, vol. 12, no. 4, p. e0174959, 2017.

[12] A. Allouch, A. Koubâa, T. Abbes, and A. Ammar, "Roadsense: Smartphone application to estimate road conditions using accelerometer and gyroscope," *IEEE Sensors Journal*, vol. 17, no. 13, pp. 4231–4238, 2017.

[13] A. Jain and V. Kanhangad, "Human activity classification in smartphones using accelerometer and gyroscope sensors," *IEEE Sensors Journal*, vol. 18, no. 3, pp. 1169–1177, 2017.

[14] X. Heng, Z. Wang, and J. Wang, "Human activity recognition based on transformed accelerometer data from a mobile phone," *International Journal of Communication Systems*, vol. 29, no. 13, pp. 1981–1991, 2016.

[15] M. Moshkin, "Distance measurement using accelerometer and gyroscope sensors," https://habr.com/ru/articles/255329/, 2015, accessed: 2024-08-05. [Online]. Available: https://habr.com/ru/articles/255329/

[16] M. Faizullin and G. Ferrer, "Best axes composition extended: Multiple gyroscopes and accelerometers data fusion to reduce systematic error," *Robotics and Autonomous Systems*, vol. 160, p. 104316, 2023.

[17] R. Ahmadian, M. Ghatee, and J. Wahlström, "Discrete wavelet transform for generative adversarial network to identify drivers using gyroscope and accelerometer sensors," *IEEE Sensors Journal*, vol. 22, no. 7, pp. 6879–6886, 2022.

[18] T. Eiter and H. Mannila, "Computing discrete fréchet distance," 1994.

[19] W. Othman, A. Kashevnik, B. Hamoud, and N. Shilov, "Driversvt: Smartphone-measured vehicle telemetry data for driver state identification," *Data*, vol. 7, no. 12, p. 181, 2022.