DOI: https://doi.org/10.15276/aait.08.2025.24
UDC 004.8:621.391:629.021

Online anomaly detection in correlated data streams using robust Kalman filtering

Taras I. Zavaliy[1)]
ORCID: https://orcid.org/0009-0002-7544-782X; taras.i.zavalii@lpnu.ua. Scopus Author ID: 36104709800
Nataliya B. Shakhovska[1)]
ORCID: https://orcid.org/0000-0002-6875-8534; nataliya.b.shakhovska@lpnu.ua. Scopus Author ID: 42962320400
[1)] Lviv Polytechnic National University, 12, Stepan Bandera Str. Lviv, 79013, Ukraine

ABSTRACT

The demand for online data analysis opens new challenges and research opportunities. The growing wave of IoT devices, low-cost sensors, and robotic systems generates vast amounts of high-frequency streaming data. Efficient online analysis of such data requires algorithms that operate under memory and latency constraints, often within a sliding-window framework. However, the reliability of these data streams critically affects the accuracy of the inference results.This study considers one of the tasks in streaming data analysis – anomaly detection in the smartphone sensors data streams. Our goal was to improve the quality of the geolocation by filtering out anomalies in the signal and then measure the accuracy of trajectory estimation for pedestrian navigation. Pedestrian navigation in urban environment is non-trivial because of global navigation sattelite system signal distortions. These distortions can be caused by various factors such as multipath effects, signal blockage from tall buildings, and interference, which are common in dense urban areas. The full data pipeline requires robust techniques for smartphone sensor data processing wich include low-pass or high-pass filtering of acceleration signal, synchronizing several streams by the timestamps, converting measurements from the device frame of reference to the global coordinate system, feature enrichments etc. When multiple data streams from device sensors are available, their fusion can be used to mitigate the limitations of individual sources. One of the adopted methods for this is the so-called robust Kalman filter. We compared this method with an ensemble anomaly detection method (iForest) applied to the geolocation data stream in the pedestrian navigation set up. We used orthogonal distance metric to compare predicted trajectories with groud truth coordinates and showed that robust Kalman filter achieves superior performance in the streaming setting. A mean deviation from the ground truth trajectory of one metre and eighty-three centimetres was achieved on the test dataset, with the total route length measuring one hundred eighty-four metres.

Keywords: Streaming data; inertial navigation; anomaly detection; Kalman filter; iForest

***For citation*: Zavaliy T. I., Shakhovska N. B. "Online anomaly detection in correlated data streams using robust Kalman filtering". *Applied Aspects of Information Technology*. 2025; Vol.8 No.4: 377–385. DOI: https://doi.org/10.15276/aait.08.2025.24**

INTRODUCTION

In recent years, there has been a surge of interest in developing alternative and hybrid positioning approaches that combine satellite navigation data with information from other sources, such as inertial sensors, cameras, Wi-Fi access points, Bluetooth beacons, and ultrasonic sensors. This is especially important for indoor navigation tasks, where GNSS signals are unavailable or unreliable, as well as for ensuring navigation of unmanned aerial vehicles and mobile robotic systems operating in complex or unknown environments. The advancement of such hybrid positioning methods brings new challenges, with a key issue being the real-time processing of streaming data. Data from various sources of different modalities arrive continuously and at high frequency, generating large volumes of streaming information.

These multimodal streams are heterogeneous in terms of frame of reference, format, rate, volume, frequency, and accuracy. A significant challenge is the presence of noise and anomalies in the data, which is a common problem for both satellite signals (multipath effects, atmospheric phenomena, interference) and local sources (vibration, interference, limited resolution, or sensor failures).
The presence of anomalies in data streams not only complicates the task of determining position but also increases the risk of incorrect decisions, which can lead to critical consequences in real-world applications. This motivates the development of robust processing algorithms capable of extracting useful signals from large volumes of noisy information in real time. Furthermore, fusion algorithms are required to be adaptive to the dynamic and unpredictable nature of sensor failures and environmental changes, ensuring continuous and reliable operation even when individual data streams become corrupted or unavailable.

This study reviews multiple examples of online anomaly detection algorithms and investigates the application of the robust Kalman filter (RAKF) and iForest to account for anomalies in GNSS signal in the pedestrian navigation task in urban environment.

RELATED WORKS

Let us consider the mainstream methods and algorithms for anomaly detection in data streams, focusing on their online implementation. Typically, these methods operate based on statistical analysis, distance metrics, density estimation, neural networks, or isolation techniques [1]. Some of these algorithms are suitable for detecting anomalies in temporally dependent geospatial data streams (such as pedestrian, bike, car, unmanned vehicle trajectory coordinates). In such cases, the data contain at least several dimensions, and anomalies may include individual outlier points, sudden "jumps", as well as atypical deviations or even loops in the trajectory.

Statistical (threshold-based) methods. Simple calculations of variance, moving averages, Z-score, CUSUM/MA (exponentially weighted moving average), and similar techniques are used to detect deviations beyond the expected variance, highlighting large spikes or gradual trends [2]. For example, in sensor failures monitoring systems, sliding thresholds for error or residuals are often employed to filter out anomalous values. The Z-score is a streaming algorithm that measures how much a value deviates from the mean within a window. The MAD (moving average deviation) operates in very similar way but uses the median and absolute deviation, making it more robust to noise.

Density- and distance-based methods. Algorithms such as k-Nearest Neighbors (k-NN) and Local Outlier Factor (LOF) detect anomalies by computing the "density" of points in the neighborhood of a given observation. For instance, LOF assesses the local density around a point and flags it as anomalous if its density is significantly lower than that of its neighbors. In [3], the authors compare cluster-based LOF with a Gaussian classifier for filtering data from wireless sensor networks. For streaming data, some modified versions like memory efficient incremental LOF summarize (cluster) older data and merge these summaries, or introduce Z-score pruning to reduce the amount of calculations [4], [5]. Similarly to LOF, in the k-NN approach, a point is considered an outlier if it has few neighbors within a specified radius. Another streaming algorithm, NETS (NET-effect-based Stream outlier detection), aggregates points into multidimensional cells within a sliding window and monitors changes in their density. This allows the algorithm to identify the most anomalous cells and then evaluate the anomaly score of each point in the region based on the distance to its neighbors or local density. This approach significantly reduces computational costs compared to classical neighbor-search-based methods and is suitable for real-time streaming data analysis. The authors of the NETS algorithm report a fivefold improvement in processing speed compared to other distance- and window-based algorithms [6].

Tree-based (ensemble) methods. These are particularly effective for streaming data analysis due to their logarithmic execution time. For example, Random Cut Forests (RCF) algorithm partitions the dataset and constructs a tree for each partition. This method is widely adopted for anomaly detection, as seen in analytical platforms like OpenSearch, which supports real-time stream processing. RCF computes a "sketch" of the incoming data stream, assigning each input value an anomaly score and a confidence score [7]. An evolution of this approach is the Isolation Forest (iForest) [8] – an ensemble regression technique that randomly divides the data into a binary tree. Points that can be easily isolated – requiring small number of splits – are classified as anomalies. Selection of hyperparameters such as n (number of trees in the forest), sample size, and contamination (expected proportion of anomalies) is typically performed using n-estimators and coordinate search with cross-validation, although the algorithm is already robust for $n>20$.

Online Isolation Forest constructs a histogram of the data and utilizes a sliding window to forget outdated values [9], thus adapting to concept drift, or changes in the underlying data distribution. Another streaming implementation, iForestASD, also employs a sliding window within which the standard iForest is applied to build random partitions (trees) [10]. If the number of detected anomalies within the window exceeds a certain threshold u, the current model is discarded and a new one is constructed for the window, accommodating concept drift. In study [11], the authors compare iForestASD with Half-Space Trees across three datasets, demonstrating iForestASD's superiority in terms of the F1-score, though the algorithm's processing speed decreases as the window size increases.

Deep learning methods. Neural networks generally demonstrate high performance in real-time applications [12], [13]. Authors in [14] achieved millisecond-level anomaly detection latency by employing a multimodal asynchronous hybrid network that combines streams from event-based cameras and images from RGB cameras (GNN+CNN). In case of time series and high-dimensional data streams, methods such as AutoRegressive Integrated Moving Average

ISSN 2617-4316 (Print)
ISSN 2663-7723 (Online)

(ARIMA), Exponentially Weighted Moving Average (EWMA), or recurrent networks (LSTM, RNN, autoencoders) are utilized. For example, the LSTM network can predict the next coordinate or velocity, and significant deviation between predictions and actual values serve as indicator of anomaly. Another approach, Language Model-based Trajectory Anomaly Detection (LM-TAD), models movement trajectories as token sequences and applies "perplexity" and "unexpectedness" metrics to identify outliers and anomalous trajectory segments [13]. This technique is well-suited to streaming processing because of its use of key-value caching in the attention mechanism.

Each of these methods offers distinct advantages and limitations: some respond rapidly to sudden spikes (LOBF, CUSUM), others effectively detect drifts (EWMA, M-estimators), while some scale efficiently to large data streams (iForest, RS-Hash). Window-based methods (LOF, RNN) can incorporate movement history (over the last few seconds), whereas clustering and most neural networks require a training set of trajectories. Point anomalies (abrupt deviations) are typically detected by statistical or distance-based methods, while atypical trajectory segments are best identified by clustering or sequence models (LM-TAD, LSTM, transformers) [13], [15]. The choice of the method in any specific scenario depends on the trade-off between response speed, model complexity, and the feasibility of model training. Equally important is the algorithm's adaptability to concept drift (changes in data distribution) and its efficiency under memory constraints [1].

In the following section, we focus on a statistical method for fusing multi-sensor data streams, where anomaly detection is achieved by analyzing the difference between predicted and measured values.

METHODS

We consider the task of anomaly detection in correlated IMU/GNSS data streams from smartphone sensors, as well as methods for evaluating the accuracy of the algorithm in pedestrian navigation task. To estimate the accuracy, it is necessary to compute the root mean square deviation (RMS) of the entire calculated trajectory from a groud thruth path. In this scenario, the following constraints are assumed: the pedestrian's initial position is known, the initial velocity is zero, the smartphone's sensors are calibrated using built-in software tools, detailed technical characteristics of the GNSS signal are unavailable, only it's accuracy estimates provided by the Android API is known.

The Kalman filter is a standard tool for data stream fusion; however, by default, it is not sufficiently sensitive to deviations in the characteristics of the input data. An anomaly or degradation in one of the data streams does not immediately influence the fusion result. Thus, adaptive reconfiguration of the filter is required as sson as we detect anomalies or changes in signal characteristics. To address this, robust estimators (such as *m*-estimators) are employed, and neural networks can be utilized for computing and updating covariance matrices [16]. In geolocation, anomaly detection can also be performed using GNSS signal quality assessment methods (e.g., RAIM/FDE). Ultimately, the Robust Adaptive Kalman Filter (RAKF) [17], [18], [19] applies Mahalanobis distance or Huber's criteria to test innovation/residual vectors at each iteration, subsequently modifying the process or measurement covariance matrices (Q, R). The threshold is usually selected from the χ^2-distribution. In some cases, the filter may fully switch to an alternative model (i.e., a different process matrix) depending on the source of the error – whether it comes from the sensor or the actuator [17].

Therefore, the anomaly detection method for GNSS coordinate stream should incorporate IMU acceleration data to identify significant discrepancies between predicted inertial displacement and the displacement measured by GNSS. The sensor fusion process enables dynamic adjustment of the "trust" assigned to each data source. In our pedestrian motion case, at each iteration k of the Kalman filtering, we compute the new state and its uncertainty as:

$$x_k = Fx_{k-1} + Gu_{k-1}$$
$$P_{k|k-1} = FP_{k-1}F^T + Q^{\cdot}$$

The state transition matrix F and state vector x models the motion of the system in one dimension (position p, velocity v and acceleration a) in the absence of external forces, assuming constant acceleration:

$$Fx = \begin{bmatrix} 1 & \Delta t & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p \\ v \\ a \end{bmatrix}.$$

The force applied to the system is represented by the control matrix G, which models acceleration changes caused by the pedestrian movement. Given that acceleration changes are measured directly by the IMU sensor at high frequency, we assume that the system's nonlinear acceleration is accurately approximated within each discrete time interval. The control matrix G implements the trapezoidal integration method for processing the control input (acceleration) data to update the overall state x_k.

It is multiplied by the control vector u, which contains both the current and previous accelerations a_k, a_{k-1}:

$$Gu = \begin{bmatrix} \frac{\Delta t^2}{4} & \frac{\Delta t^2}{4} \\ \frac{\Delta t}{2} & \frac{\Delta t}{2} \\ 0.5 & 0.5 \end{bmatrix} \begin{bmatrix} a_k \\ a_{k-1} \end{bmatrix}.$$

Our noise covariance matrices Q and R are derived from sensor's white noise standard deviation, R^2 score and GNSS accuracy estimate [20].

At each prediction step, the drift in the integrated velocity is corrected by replacing it with the velocity predicted by our pre-trained LSTM network:

$$v = v_{LSTM}.$$

The LSTM network was trained on the dataset with 133000 total acceleration samples and 2500 target speed measurements. It achieved mean average error of 0.087 m/s and R^2 score of 0.83 on the test dataset [20].

Next, we perform the correction at each iteration k, at the GNSS sampling frequency of 1 second, as follows:

$$y_k = z_k - Hx_{k-1},$$
$$P_k = (I - KH)P_{k-1},$$
$$K = PH^T(HPH^T + R)^{-1}.$$

Here, y_k is the innovation (residual) vector, z_k is the measurement vector, I is the diagonal identity matrix, and $H = [1 \quad 0 \quad 0]$ is the observation matrix, projecting the state vector x onto GNSS coordinates while ignoring velocity and acceleration.

The Kalman gain K is computed at each iteration, as is the Mahalanobis distance for the innovation vector, according to:

$$d_M^2 = y_k^T(HPH^T + R)^{-1}y_k$$
$$d_M^2 \sim \chi^2(m).$$

The Mahalanobis distance should follow the χ^2-distribution with m degrees of freedom. This distribution determines the threshold value (in our case – one standard deviation), which strictly constrains the GNSS measurement's conformity to the model.

To assess the algorithm's performance, we compare the computed trajectory to the ground truth path. Calculating the root mean square deviation (RMS) for the entire route is nontrivial because: 1) the data points represent geographic coordinates, subject to Earth's curvature; 2) coordinates are received as a stream and require a window function for distance calculation; 3) the ground truth route is a static array of coordinates without timestamps. Table 1 briefly compares possible metrics for distance calculation between two trajectories.

We employed the orthogonal distance to the segment method, as it is a fast and accurate approach that does not require timestamps. Similarly to the average minimum distance, it involves $O(n*m)$ operations and is suitable for streaming algorithms. Other metrics require simplifications for streaming application or synchronization of two streams using timestamps.

RESULTS

The data structure of inertial (IMU) and geolocation (GNSS) sensor streams is mostly flat. Each message in the acceleration sensor stream contains the following fields:

– *timestamp*, absolute time of the individual measurement;

– *seconds elapsed*, relative time since the start of the measurement session;

– *x*, acceleration along the X axis in m/s²;

– *y*, acceleration along the Y axis in m/s²;

– *z*, acceleration along the Z axis in m/s².

And the data stream from the geolocation sensor contains messages with the following structure:

– *timestamp*, absolute time of the individual measurement;

– *seconds elapsed*, relative time since the start of the measurement session;

– *altitude*, height in meters relative to the WGS84 ellipsoid,

– *bearing*, deviation in degrees relative to geographic north, {0; 360};

– *speed*, speed in m/s;

– *latitude*, latitude in degrees, {-90; 90};

– *longitude*, longitude in degrees, {-180; 180}.

A quick exploratory analysis provided an initial assessment of data quality. One of the main issues is that GNSS sensor readings can be highly inconsistent. In urban environments, signal loss for several seconds may be observed instead of a stable stream at a frequency of 1 Hz. Acquiring GNSS sattelites lock at the beginning of recording requires time, so a waiting period was added at the start of each route. In dense urban living areas, negative

ISSN 2617-4316 (Print)
ISSN 2663-7723 (Online)

Table 1. **Metrics for estimating distance between two trajectories**

Metric	Description	Algorithmic complexity	Streaming implementation	Timestamps required
Orthogonal Distance to the Segment	Each point of the predicted trajectory is projected onto the nearest segment of the ground truth trajectory.	O(n * m) (*n, m* – number of points in each trajectory)	Yes	No
Hausdorff Distance	Estimates the greatest minimal distance between two sets of points.	O(n * m)	Partial (buffer)	No
Fréchet Distance	Takes the ordering of the data points into account. Determines the minimum length of the "leash" between two routes.	O(n * m) with dynamic programming	No (global)	No
Dynamic Time Warping (DTW)	Aligns two sequences with different frequencies.	O(n * m) with dynamic programming	Partial	Yes
RMSE by time	Root mean square error between points at corresponding time moments.	O(n) (when frequency is equal)	Partial	Yes
Average Minimum Distance (AMHD)	For each predicted point, the minimum distance to closest ground truth point is determined.	O(n * m)	Yes	No

***Source*: compiled by the authors**

effects such as signal reflection and satellite blockage were observed, causing the GNSS trajectory to gradually deviate from the true trajectory.

Thankfully, in addition to geodetic coordinates, the Android API provides accuracy estimates, which are summarized for one of the routes in Table 2. These estimates are used to initialize the measurement covariance matrix R in our Kalman filter. Heading accuracy is provided in degrees. Horizontal and vertical accuracy are estimated in meters as the radius of uncertainty. Speed accuracy is provided in meters per second.

Table 2. **Accuracy estimates for GNSS-measurements on one of the routes**

Metric	*count*	*mean*	*std*	*min*	*max*
Heading accuracy	178	28.22	15.73	0	103.9
Speed accuracy	178	0.7	0.31	0.08	1.5
Vertical accuracy	178	2.67	0.79	2.5	10.49
Horizontal accuracy	178	3.88	0.8	3.79	12.45

***Source*: compiled by the authors**

Advanced data fusion from multiple sensors is integrated into Android operating system; thus, orientation measurements are stable and contain minimal drift. Only local magnetic field distortions can significantly affect the estimation of movement direction. We need to know the exact device orientation not only for transforming accelerations into the global ENU (East-North-Up) coordinate system during preprocessing, but also for converting scalar speed value into a vector quantity.

To demonstrate the effectiveness of the proposed Kalman-LSTM-Robust algorithm, GNSS coordinates were deliberately distorted along the CD segment of the trajectory, and a 10-second signal loss was simulated at one of the trajectory's corners (D). Fig. 1 represents the results of trajectory recovery using the predicted inertial velocity and displacement (shown in red).

Under normal conditions, the calculated inertial trajectory closely follows the GNSS trajectory, as the Kalman filter corrects the prediction based on measurements every second. However, in the presence of an anomalous deviation at point F and signal loss at point D, the LSTM-Kalman-Robust algorithm discards GNSS readings, completely bypassing the correction step and relying solely on IMU data. As shown in Fig. 2, the streaming iForestASD algorithm accurately detects anomalous outliers. However, since it operates within a sliding window, it returns to the "normal" mode with some delay, which results in greater impact of drift in the Kalman filter. The number of trees was set to n=100, the window size was 300 seconds, and the threshold value was 0.25.

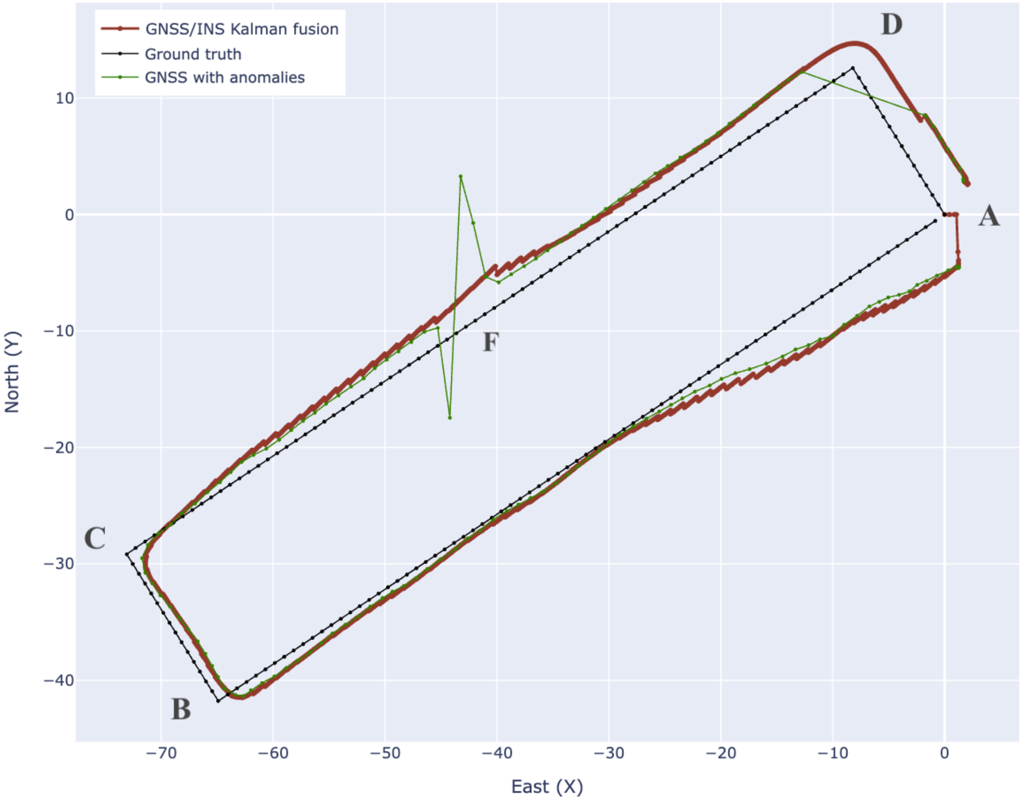

Fig. 1. **The results of the Kalman-LSTM-Robust algorithm under simulated 10-second GNSS signal loss at point *D* and intentional distortion at point *F***
***Source*: compiled by the authors**

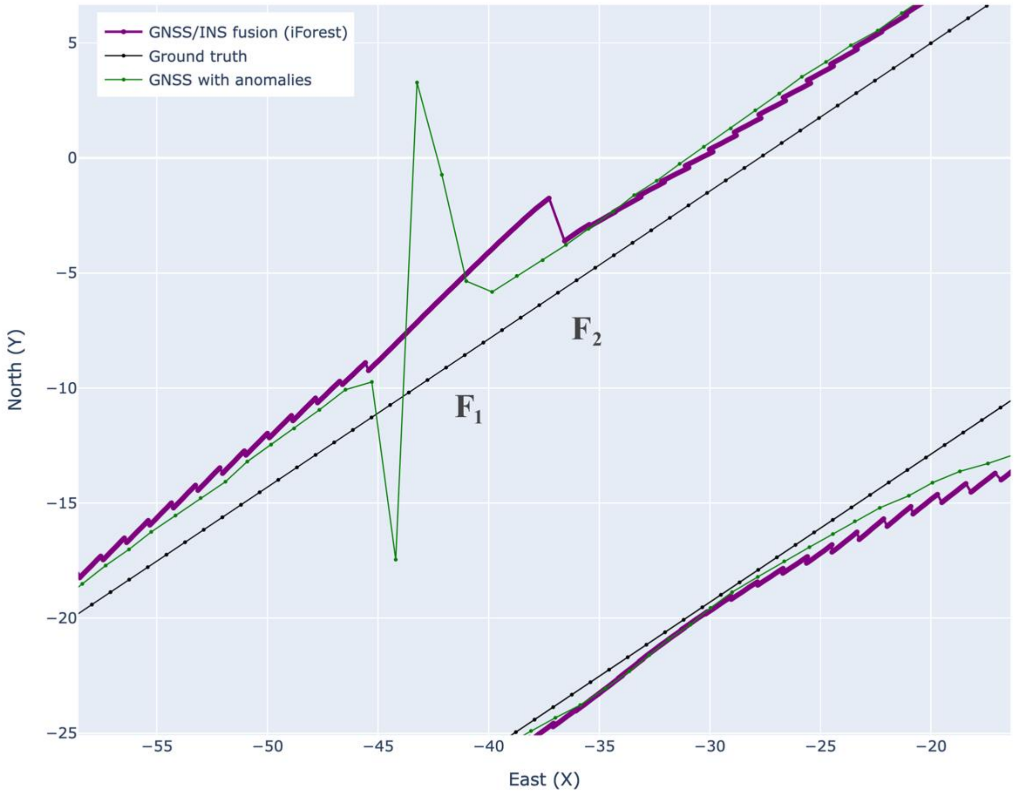

Fig. 2. **The iForest-LSTM-Kalman algorithm introduces a delay at point F2, which negatively affects the mean deviation from the ground truth trajectory**
***Source*: compiled by the authors**

ISSN 2617-4316 (Print)
ISSN 2663-7723 (Online)

Table 3. **Impact of robust Kalman filter and iForest in trajectory prediction relative to GNSS ground truth trajectory**

№	Method	Mean deviation from the ground truth, meters	Maximum deviation from the ground truth, meters
1	LSTM	2.33	6.51
2	LSTM-Kalman	1.91	5.81
3	LSTM-Kalman-Robust	1.83	4.51
4	iForest-LSTM-Kalman	1.86	4.52
5	GNSS measurement with distortions	1.81	11.13

Source: **compiled by the authors**

For each point X in the computed trajectory, to estimate the deviation from the ground truth, the orthogonal distance to the segment was used to find the nearest point Y on the ground truth trajectory segment. Then we followed by calculating the standard Euclidean distance (L2 norm) between the ENU coordinates of these points. For the predicted trajectory ABCDE, the mean and maximum deviations were 2.3 and 6.5 meters, respectively (see Table 3). After adding anomaly detection to the Kalman filter, the mean and maximum deviation went down to 1.83 and 4.51 meters, respectively.

CONCLUSIONS

The article investigates methods for anomaly detection in data streams within the pedestrian navigation task. Anomaly detection is considered as one of the steps in the full sensor data processing pipeline: noise filtering, coordinate transformation, feature transformation, LSTM inference, outlier filtering, IMU and GNSS data fusion, final error estimation.

It was demonstrated that fusing several data streams from smartphone sensors using a conventional Kalman filter does not provide sufficient robustness to anomalies; a method for eliminating GNSS coordinate jumps is required.

Using only inertial data for LSTM-based speed prediction resulted in an average deviation of 2.33 meters on a 184-meter route. Incorporating readings from additional sensors in a robust Kalman filter, with filtering of anomalous GNSS sensor readings, reduced the mean deviation to 1.83 meters and the maximum deviation from 6.51 to 4.51 meters.

The advantages of this approach to anomaly detection include its high speed, versatility for various movement models (pedestrians, vehicles, drones), and the absence of a need for training.

REFERENCES

1. Cao, Y., Ma, Y., Zhu, Y. & Ting, K. M. “Revisiting streaming anomaly detection: benchmark and evaluation”. *Artificial Intelligence Review*. 2024; 58 (1): 8, https://scopus.com/pages/publications/85208694032. DOI: https://doi.org/10.1007/s10462-024-10995-w.

2. Skakovskyi, V. O., Savitskyi, R. S. & Fant, M. O. “Methods and algorithms for anomaly detection”. *Computer Science and Applied Mathematics*. 2025; 1: 80–88. DOI: https://doi.org/10.26661/2786-6254-2025-1-10.

3. Giannoni, F., Mancini, M. & Marinelli, F. “Anomaly detection models for IoT time series data”. *arXiv*. 2018. DOI: https://doi.org/10.48550/arXiv.1812.00890.

4. Salehi, M., Leckie, C., Bezdek, J. C., Vaithianathan, T. & Zhang, X. “Fast memory efficient local outlier detection in data streams”. *IEEE Transactions on Knowledge and Data Engineering*. 2016; 28: 3246–3260. DOI: https://doi.org/10.1109/TKDE.2016.2597833.

5. Yang, X., Zhou, W., Shu, N. & Zhang, H. “A fast and efficient local outlier detection in data streams”. In *Proceedings of the International Conference on Image, Video and Signal Processing*. New York, NY, USA. 2019. p. 111–116, https://scopus.com/pages/publications/85065799808. DOI: https://doi.org/10.1145/3317640.3317653.

6. Yoon, S., Lee, J.-G. & Lee, B. S. "NETS: extremely fast outlier detection from a data stream via set-based processing". *Proc. of VLDB Endowment*. 2019; 12 (11): 1303–1315, https://scopus.com/pages/publications/85077815763. DOI: https://doi.org/10.14778/3342263.3342269.

7. Guha, S., Mishra, N., Roy, G. & Schrijvers, O. "Robust random cut forest based anomaly detection on streams". In *Proceedings of the 33rd International Conference on International Conference on Machine Learning*. New York, NY, USA. 2016. p. 2712–2721, https://scopus.com/pages/publications/84998672568 DOI: https://dl.acm.org/doi/10.5555/3045390.3045676.

8. Liu, F. T., Ting, K. M. & Zhou, Z.-H.. "Isolation Forest". In *2008 Eighth IEEE International Conference on Data Mining*. 2008. p. 413–422, https://scopus.com/pages/publications/67049142378. DOI: https://doi.org/10.1109/ICDM.2008.17.

9. Leveni, F., Cassales, G. W., Pfahringer, B., Bifet, A. & Boracchi, G. "Online Isolation Forest". In *Proceedings of the 41st International Conference on Machine Learning*. 2024. p. 27288–27298. – Available from: https://proceedings.mlr.press/v235/leveni24a.html. – [Accessed: 01 Nov 2024].

10. Ding, Z. & Fei, M. "An anomaly detection approach based on Isolation Forest Algorithm for streaming data using sliding window". *IFAC Proceedings Volumes*. 2013; 46 (20): 12–17, https://scopus.com/pages/publications/84896376427. DOI: https://doi.org/10.3182/20130902-3-CN-3020.00044.

11. Togbe, M. U. et al. "Anomaly detection for data streams based on isolation forest using scikit-multiflow". In *Computational Science and Its Applications – ICCSA 2020*. 2020; 12252: 15–30, https://scopus.com/pages/publications/85092254171. DOI: https://doi.org/10.1007/978-3-030-58811-3_2.

12. Xia, Y. et al. "Anomaly detection for urban vehicle GNSS observation with a hybrid machine learning system". *Remote Sensing*. 2020; 12 (6): 971, https://scopus.com/pages/publications/85082308320. DOI: https://doi.org/10.3390/rs12060971.

13. Mbuya, J., Pfoser, D. & Anastasopoulos, A. "Trajectory anomaly detection with language models". *arXiv*. 2024. DOI: https://doi.org/10.48550/arXiv.2409.15366.

14. Xiao, D. et al. "When every millisecond counts: real-time anomaly detection via the multimodal asynchronous hybrid network". *arXiv*. 2025. DOI: https://doi.org/10.48550/arXiv.2506.17457.

15. Lan, D. T. & Yoon, S. "Trajectory clustering-based anomaly detection in indoor human movement". *Sensors*. 2023; 23 (6):3318, https://scopus.com/pages/publications/85151195976. DOI: https://doi.org/10.3390/s23063318.

16. Mortada, H., Falcon, C., Kahil, Y., Clavaud, M. & Michel, J.-P. "Recursive KalmanNet: Deep Learning-Augmented Kalman Filtering for state estimation with consistent uncertainty quantification". *arXiv*. 2025. DOI: https://doi.org/10.48550/arXiv.2506.11639.

17. Soken, H. & Hajiyev, C. "Robust Adaptive Kalman Filter for Estimation of UAV dynamics in the presence of Sensor/Actuator faults". *Aerospace Science and Technology*. 2013; 28: 376–383, https://scopus.com/pages/publications/84878845132. DOI: https://doi.org/10.1016/j.ast.2012.12.003.

18. Yang, L., Lin, X., Hou, Y., Ren, J. & Wang, M. "Application of an Improved Adaptive Unscented Kalman Filter in vehicle driving state parameter estimation". *International Journal of Adaptive Control and Signal Processing*. 2025; 39 (5): 1021–1035, https://scopus.com/pages/publications/85218675693 DOI: https://doi.org/10.1002/acs.3989.

19. Yu, Z., Zhang, Q., Yu, K. & Zheng, N., "A State-Domain Robust Chi-Square test method for GNSS/INS integrated navigation", *Journal of Sensors*, vol. 2021, pp. 1–8, Oct. 2021. https://scopus.com/pages/publications/85117411521. DOI: https://doi.org/10.1155/2021/1745383.

20. Zavaliy, T. "Fusion of GNSS and IMU sensor streams for improved pedestrian navigation". In *Developments in Information and Knowledge Management Systems for Business Applications*. 2025; 9: 215–234, https://scopus.com/pages/publications/105018004844. DOI: https://doi.org/10.1007/978-3-031-95955-4_12.

Conflicts of Interest: The authors declare that they have no conflict of interest regarding this study, including financial, personal, authorship or other, which could influence the research and its results presented in this article

Received 20.10.2025
Received after revision 27.11.2025
Accepted 04.12.2025

DOI: https://doi.org/10.15276/aait.08.2025.24
УДК 004.8:621.391:629.021

Онлайн-виявлення аномалій в залежних потоках даних з використанням стійкого фільтру Калмана

Завалій Тарас Ігорович[1)]
ORCID: https://orcid.org/0009-0002-7544-782X; taras.i.zavalii@lpnu.ua. Scopus Author ID: 36104709800
Шаховська Наталія Богданівна[1)]
ORCID: https://orcid.org/0000-0002-6875-8534; nataliya.b.shakhovska@lpnu.ua. Scopus Author ID: 42962320400
[1)] Національний університет “Львівська політехніка”, вул. Степана Бандери, 12, Львів, 79013, Україна

АНОТАЦІЯ

Попит на онлайн-аналіз даних породжує нові виклики та задачі для досліджень. Ми спостерігаємо поширення IoT-пристроїв, недорогих сенсорів та роботизованих систем, які генерують значні обсяги високочастотних потокових даних. Ефективний онлайн-аналіз таких даних потребує алгоритмів, що працюють за умов обмежених обчислювальних ресурсів та пам'яті, часто в рамках ковзаючого вікна. Що не менш важливо, надійність цих потоків даних суттєво впливає на точність результатів моделювання. У цьому дослідженні розглядається одна з актуальних задач аналізу потокових даних – виявлення аномалій у потоках даних із сенсорів смартфону. Основною метою роботи було підвищення якості геолокації шляхом фільтрації аномалій у потоці гео-координат та подальше вимірювання точності обчислення траєкторії в задачі навігації пішохода. Навігація пішохода в міському середовищі є нетривіальною задачею через спотворення сигналу глобальної навігаційної супутникової системи. Такі спотворення часто спричинені різними факторами, зокрема ефектом багатопроменевого поширення, блокуванням сигналу високими будинками та інтерференцією, що є типовим для густонаселених міських районів. Для повного циклу опрацювання потоків даних необхідно застосовувати робастні методи обробки сенсорних даних смартфону, що включають фільтрацію сигналу прискорення (низько- або високочастотну), синхронізацію кількох потоків за часовими мітками, перетворення вимірів із системи координат пристрою в глобальну систему координат, інженерію ознак тощо. У випадку, коли наявні потоки даних з різних сенсорів, їхнє злиття може компенсувати недоліки окремого сенсора. Одним із поширених методів для цього є так званий робастний фільтр Калмана. Ми порівняли цей метод із ансамблевим методом виявлення аномалій (iForest), застосованим до потоку геолокаційних даних під час моделювання руху пішохода. Ми використали метрику ортогональної відстані для порівняння прогнозованої траєкторії з координатами еталонного маршруту, та показали, що в потоковому режимі робастний фільтр Калмана демонструє кращі результати. На тестовому наборі даних було досягнуто середнього відхилення від еталону в один метр вісімдесят три сантиметри при загальній довжині маршруту сто вісімдесят чотири метри.

Ключові слова: потокові дані; інерційна навігація; виявлення аномалій; фільтр Калмана; iForest

ABOUT THE AUTHORS

Taras I. Zavaliy - PhD student, Department of Artificial Intelligence. Lviv Polytechnic National University, 12, Stepan Bandera Str. Lviv, 79005, Ukraine
ORCID: https://orcid.org/0009-0002-7544-782X; taras.i.zavalii@lpnu.ua. Scopus Author ID: 36104709800
Research field: Sensor fusion, data streaming, machine learning

Завалій Тарас Ігорович - аспірант кафедри Штучного інтелекту. Національний університет «Львівська політехніка», вул. Степана Бандери, 12. Львів, 79005, Україна

Nataliya B. Shakhovska - Doctor of Engineering Science, professor at the Department of Artificial Intelligence. Lviv Polytechnic National University, 12, Stepan Bandera Str. Lviv, 79005, Ukraine
ORCID: https://orcid.org/0000-0002-6875-8534; nataliya.b.shakhovska@lpnu.ua. Scopus Author ID: 42962320400
Research field: Big data, information systems, machine learning

Шаховська Наталія Богданівна - доктор технічних наук, професор кафедри Штучного інтелекту. Національний університет «Львівська політехніка», вул. Степана Бандери, 12. Львів, 79005, Україна