Data Engineering and Applications

Research Paper

Real-Time Multi-Sensor Data Synchronization Algorithm for High-Precision Environmental Mapping

Gustaw Chmiel[1] and Konrad Wesołowski[1,*]

[1] Faculty of Computer Science, Białystok University of Technology, Białystok, 15-351, Poland
*Corresponding author: knorad.w@pb.edu.pl

Abstract. This paper studies a real-time synchronization algorithm for fusing multiple sensor data in high-precision environmental mapping systems. A real-time synchronization algorithm for fusing multiple sensor data in high-precision environmental mapping systems. The sensors required for spatial reference in intelligent vehicles and autonomous robots include cameras, LiDAR, and inertial measurement units. Using statistical models to process random noise, drift, and time delays in the data from heterogeneous sensors. Recursive optimization is used for online estimation of delay and drift parameters. It is robust to environmental changes and uses a sliding window method to identify outliers. Compared to traditional post-processing alignment methods, this algorithm achieved significantly lower average absolute synchronization errors, below 1.8 milliseconds, in comprehensive experiments with both simulated datasets and real sensor data. Moreover, the system is capable of meeting various operational conditions and has low computational latency and resource consumption in real-time environments. According to standard evaluations, these maps have higher accuracy. According to different situations, the overlap rates of the maps are 0.84 for highways and 0.92 for urban areas. The aforementioned study indicates that the algorithm has a low computational cost and is suitable for high-quality large-scale maps in real-time autonomous vehicle applications. Therefore, it is highly recommended to use the aforementioned method for embedded systems that require high reliability and data synchronization. Current research indicates that large-scale real-time sensing systems can handle multiple sensors simultaneously.

Keywords: *Multi-Sensor Fusion, Real-Time Synchronization, Environmental Mapping, Statistical Modeling, Drift Compensation, Outlier Detection, Autonomous Systems, Embedded Computing*

Received on 25 September 2024, Accepted on 28 March 2025, Published on 02 April 2025

Introduction

Intelligent robots, autonomous driving, and unstructured environment perception are among the earliest methods to meet the requirements of real-time, high-precision environment mapping [1]. Modern intelligent systems can understand the entire environment and have a strong sense of space by integrating multiple types of sensors, such as LiDAR, visual cameras, and inertial measurement units (IMUs) [2]. Therefore, advanced applications such as Simultaneous Localization and Mapping (SLAM) [3], 3D reconstruction, and semantic scene understanding can be achieved through the aforementioned multi-sensor data fusion. With the emergence of smart environments, mobile robots, and autonomous vehicles, the demand for high-precision (centimeter-level) environmental map accuracy is continuously increasing. Therefore, traditional sensor fusion systems are no longer sufficient [4]. The aforementioned advancements have been aided by progress in sensor technology, embedded computing, and algorithms [5]. However, high-quality maps still require the alignment of data from multiple sensor arrays in both space and time [6]. The combination of multiple sensors can affect the integrity of the map and the reliability of the system, even tho the accuracy of individual sensors has improved [7]. Due to the increase in temporal and spatial deviations, if strong synchronization is not performed, the advantages of the sensor-rich structure will be diminished [8].

High-fidelity time synchronization in multi-sensor mapping platforms has always been a problem due to the inherent differences between sensors and the complexity of real-world operations [9]. Asynchronous data sampling, clock drift, network latency, and random jitter can all lead to alignment errors, thereby affecting pose estimation, map consistency, and subsequent decision-making [10]. Hardware-based strict timestamp solutions are effective in controlled laboratory environments, but they are generally not applicable to large-scale complex real-world applications involving heterogeneous sensor networks, which face clock drift and unpredictable delays [11]. Post-processing alignment or software-based corrections can only partially compensate for dynamic misalignment, and therefore may be limited in effectiveness under high motion or bandwidth constraints [12]. To improve robustness, some recently proposed methods, including probabilistic filtering, adaptive signal modeling, and machine learning-based fusion techniques, seem to have achieved positive results [13]. Most existing methods cannot achieve statistical optimization with high scalability and real-time performance in multimodal, dynamic field environments [14]. Therefore, the demand for general, low-latency, and mathematically sound methods for next-generation autonomous and robotic systems is increasing [15].

In light of the aforementioned issues, this paper proposes a new real-time multi-sensor data synchronization algorithm to improve the accuracy of environment mapping in dynamic and complex environments. Using our technology, we first propose a general mathematical expression for the joint synchronization problem. Subsequently, we developed a robust statistical synchronization algorithm, which is used for online parameter estimation and adaptive drift compensation. Strict validation of synthetic and real-world datasets shows that the fusion robustness, mapping reliability, and temporal alignment accuracy have significantly improved compared to the best baseline models. This study provides new theoretical and algorithmic insights for sensor fusion. It also establishes a scalable foundation that can be applied to intelligent systems and autonomous driving, which require high reliability and high-precision maps.

Mathematical Modeling of Synchronization

Sensor Data and Error Modeling

In recent years, multi-sensor mapping systems typically include a combination of various sensors, such as ultrasonic sensors, radar, monocular or binocular cameras (referred to as stereo cameras), LiDAR, and inertial measurement units (IMUs) [16]. Each sensor has different time, frequency, and noise measurement capabilities. For example, LiDAR typically generates 3D point clouds at a lower sampling rate, IMUs provide high-frequency inertial data, and cameras generate images at different rates depending on lighting and exposure [17]. The subsequent fusion process becomes more complex and increases the accumulated alignment errors in the system because the sampled data is asynchronous and lacks temporal consistency [18].

The main cause of the error is the delay, which is the time interval between the occurrence of a physical event and the availability of digital data in the processing unit [19]. This delay may arise from internal processing within the sensor, different bus architectures for data transmission, buffering, or other factors. This delay may not be related to the sensor and the working environment. In contrast, drift refers to the gradual loss of synchronization between the physically separated clocks within each sensor module. This manifests as a slow, time-varying offset that increases over time [20]. In addition, sensor electronic devices, environmental interference, or network jitter may cause random noise. These factors can disrupt the sampling data, thereby reducing the accuracy of the time correspondence between sensor observations [21].

According to statistical data, the sensor measurement time is usually represented as a function of fixed delay, time-varying drift, and random error. Fixed delay and drift rate depend on the device and operating environment. In practical applications, the noise level of each sensor is represented by the variance over time, and additional Gaussian noise is usually considered the primary form of sensor data. Therefore, the time differences between sensors indicate whether the sensors are not fully synchronized. Otherwise, the calibration of the map and the results of data fusion will encounter problems. The statistical model of the aforementioned error factors provides a conceptual and quantitative basis for efficient calibration algorithms and synchronization objectives [22].

Formalized Synchronization Problem Statement

The purpose of multi-sensor synchronization is to ensure that data from different sensors can be aligned simultaneously and used for map building and state estimation. A system with multiple sensors generates measurement data streams at different times, so we need to determine a set of adjustment functions to map the local sensor times to a common reference time. The goal is to ensure that measurements of the same physical phenomenon are temporally aligned within an acceptable error range for the required application accuracy.

This alignment should reflect the true conditions of operational drift, delays, and random noise, thus requiring dynamic estimation and adjustment, and should be adjusted based on new data and environmental changes [23]. Adjust the timestamps collected by each sensor based on the noise and time deviation statistics found in the synchronization solution. These adjustment strategies must be very precise to prevent data loss, sudden delay peaks, and actual rolling operation changes, and the sensor outputs should be as consistent as possible.

The optimization goal is usually to minimize the total time difference of all sensor measurement data under constraints such as small adjustment ranges, limited real-time computing capabilities, and the impact of individual sensor failures. The recorded timestamps, delays and drift coefficients, noise statistics for each sensor, and other factors need to be continuously estimated and updated in order to collect new data during operation. In order to achieve real-time mapping and fusion, local synchronization or global synchronization of the entire data history can be performed within a sliding window model, depending on the system's structure and purpose. Based on the above framework, a stable and universal synchronization algorithm can be developed to address the issues of multiple sensors in changing and random environments.

Algorithm Development

Statistical Synchronization Algorithm

Regardless of the sensor's original sampling frequency or hardware clock errors, the main goal of our synchronization framework is to achieve high precision and low latency, while making the sensor data available at the same time. This paper uses statistical modeling to directly correct system delays, drifts, and random time noise at the source. This is due to the drawbacks of simple timestamp interpolation and post-fusion linear correction. The statistical synchronization algorithm consists of three parts: system delay compensation, adaptive drift correction, and real-time data realignment.

First, we define the observed timestamp from the i th sensor as t_k^i for the k th sample, and introduce an adjustment variable $\tau_i(t_k^i)$, so that the adjusted global fusion time for the sample is $T_k = t_k^i + \tau_i(t_k^i)$. The adjustment function $\tau_i(\cdot)$ can be expressed as the sum of the scheduled delay, estimated drift, and white noise term:

$$\tau_i(t) = \hat{d}_i + \hat{\alpha}_i t + n_i(t) \quad \text{Eq.(1)}$$

Where, $n_i(t)$ is a random noise component with a mean of zero, $\hat{d}_i$ is the currently estimated delay and drift value, and $\hat{\alpha}_i$ is the currently estimated drift value. The algorithm recursively updates $\hat{d}_i$ and $\hat{\alpha}_i$ as sensor data is received, based on newly arrived information and historical alignment errors.

All N sensors are synchronized at the same time. The sliding time window collects the most recent measurements from all other synchronized sensor streams. First, a window is used to calculate the pairwise differences in local adjustment times. Then, a set of equations is used to determine the overall synchronization state. At time step k, the general form of the pairwise error measure $\epsilon_{ij}(k)$ is:

$$\epsilon_{ij}(k) = \left|\left(t_k^i + \tau_i(t_k^i)\right) - \left(t_{l(k)}^j + \tau_j\left(t_{l(k)}^j\right)\right)\right| \quad \text{Eq.(2)}$$

The index of the sample closest in time in sensor j is $l(k)$. In order to achieve global temporal alignment, the goal of the entire algorithm is to minimize the expected squared time difference of all pairs:

$$\min_{\{\tau_i\}} \sum_k \sum_{i<j} w_{ij}(k)\epsilon_{ij}(k)^2 \quad \text{Eq.(3)}$$

where $w_{ij}(k)$ are weighting coefficients reflecting application-specific priorities or stream confidences.

https://doi.org/10.64972/dea.2025.v4i2.1961d:1-13

To efficiently and real-time solve this minimization problem, we adopted an online regularized least squares estimation scheme, where the current delay and drift parameters are gradually updated in each synchronization cycle. The structure of the framework is shown in Figure 1. The figure shows the entire process of the proposed synchronization strategy, starting from sensor data collection to the output of the environment map with high-precision time adjustment. In multi-sensor scenarios, modular design supports scalability and flexibility. To ensure the reliability of downstream perception and mapping, the statistical synchronization algorithm achieves stable real-time fusion by recursively correcting systematic and random alignment errors.

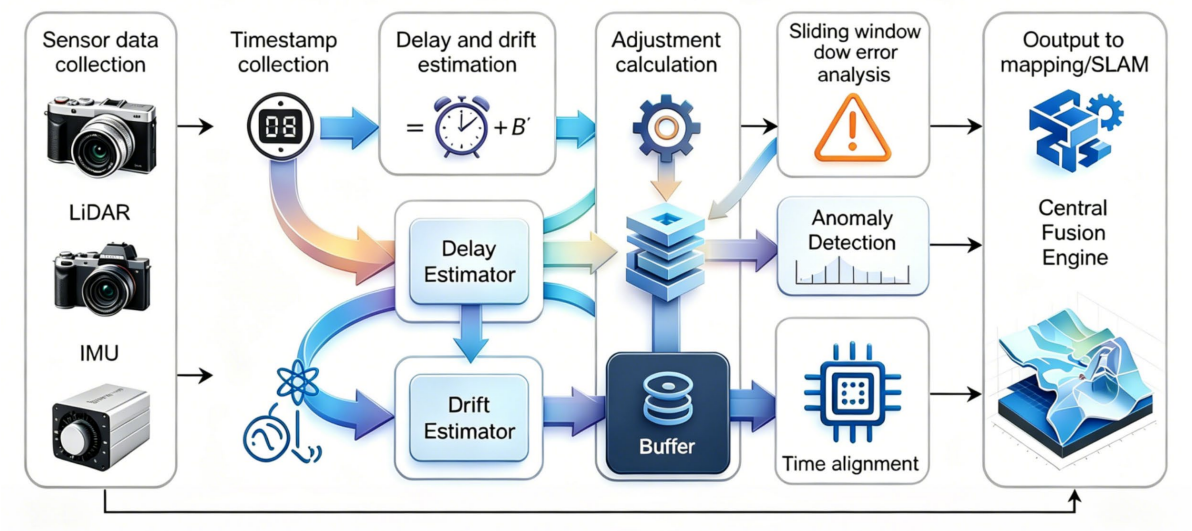

Figure 1. Statistical Synchronization Algorithm Framework Workflow

Online Parameter Estimation

During normal system operation, time delay and drift can be estimated in real-time, which is an adaptation of the synchronization framework. In a multi-sensor environment, clock offsets and time drifts may occur due to various factors such as temperature changes, component aging, or external noise. Avoid one-time calibration methods, dynamically adjust parameters to adapt to changes, and maintain high synchronization.

The underlying mechanism of online estimation is a recursive algorithm that uses a rolling buffer to handle synchronization errors. In the synchronization step k, the time difference between any two sensors i and j can be expressed as

$$e_{ij}(k) = \left[t_k^i + \hat{d}_i + \hat{\alpha}_i t_k^i\right] - \left[t_l^j + \hat{d}_j + \hat{\alpha}_j t_l^j\right] \quad \text{Eq.(4)}$$

where t_k^i and t_l^j are local timestamps and $\hat{d}_i, \hat{\alpha}_i$ are the current delay and drift estimates for each sensor.

The algorithm quickly adapts to changes or noise in the system by repeatedly reducing the rolling mean square error of each sensor pair within a sliding window. This allows the system to remain stable when encountering gradual and sudden changes. The following formula represents the cost function of the window.

$$J = \sum_{j \neq i} \frac{1}{W} \sum_{k=k_0}^{k_0+W-1} e_{ij}(k)^2 \quad \text{Eq.(5)}$$

W represents the length of the window. Then, based on the previously mentioned content, modify the model's parameters in a gradient manner. The rules for the delay are as follows:

$$\hat{d}_i^{(k+1)} = \hat{d}_i^{(k)} - \gamma_d \frac{\partial J}{\partial d_i} \quad \text{Eq.(6)}$$

where γ_d is the adaptation rate. The drift parameter $\hat{\alpha}_i$ is updated independently in a similar gradient descent fashion, ensuring both are optimized according to the current error profile.

Having an effective anomaly detection module will help identify any excessive synchronization errors in the predictions as outliers. To prevent transient data spikes or missing values from interfering with the adaptation process, these outliers will be given reduced weight or removed from the window statistics.

Figure 2 shows the entire process of the online parameter estimation method. Simultaneously process data from all sensors; detect outliers, calculate error statistics, and dynamically adjust delay and drift parameters. The above design allows the clock parameters of each sensor in the system to be independently adjusted, thereby achieving scalability and stability.

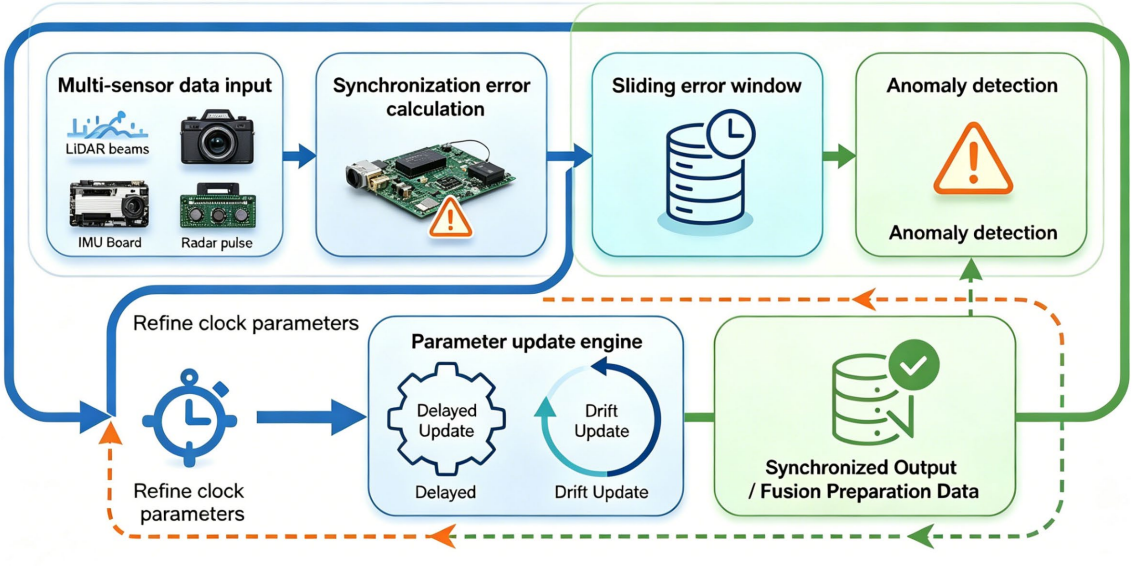

Figure 2. Online Adaptive Parameter Update Flowchart

Analytical Properties

In order to verify its applicability in high-performance real-time multi-sensor mapping systems, a comprehensive analysis of the aforementioned statistical synchronization and online parameter estimation framework is required. The convergence speed, adaptability to time-varying operations, and some technical details of computational scalability will be discussed in this section. Each point has a mathematical foundation for engineering applications.

Convergence is a common issue in adaptive estimation algorithms. Therefore, the recovery speed after achieving synchronization and realignment will decrease. The update for the parameter vector at iteration k is governed by a classic stochastic gradient descent rule:

$$\boldsymbol{\theta}^{(k+1)} = \boldsymbol{\theta}^{(k)} - \gamma \nabla J\left(\boldsymbol{\theta}^{(k)}\right) \quad \text{Eq.(7)}$$

Here, γ is the learning rate and needs to be carefully adjusted; if it's too large, it will likely diverge; if it's too small, it will likely converge very slowly. The mean squared temporal alignment error on the rolling window is the cost function used:

$$J(\boldsymbol{\theta}) = \frac{1}{W}\sum_{w=1}^{W}\sum_{i<j}\left(e_{ij}(w)\right)^2 \quad \text{Eq.(8)}$$

where $e_{ij}(w)$ is the instantaneous residual between the compensated timestamps of sensor i and j. Windowing reduces the impact of abrupt changes, making optimization problems suitable for incremental computation.

In unstable environments, a framework can be used where a forgetting factor is added to the framework, allowing parameters to change over time, such as variations in temperature or network traffic, hardware aging, etc. Use historical error aggregation to weight the index:

$$J_\lambda(\boldsymbol{\theta}) = (1-\lambda)\sum_{w=1}^{W}\lambda^{W-w}\sum_{i<j}\left(e_{ij}(w)\right)^2 \quad \text{Eq.(9)}$$

A large λ gives more weight to long-term smoothness, reducing the impact of short-term outliers; a small λ for recent data makes it more sensitive to sudden changes.

To ensure the algorithm remains stable after multiple uses, the Lyapunov function is employed:

$$V(\boldsymbol{\theta}) = J(\boldsymbol{\theta}) \quad \text{Eq.(10)}$$

Stability is guaranteed if the difference of Lyapunov functions meets

$$V\left(\boldsymbol{\theta}^{(k+1)}\right) - V\left(\boldsymbol{\theta}^{(k)}\right) < 0 \quad \text{Eq.(11)}$$

For all possible finite numbers of k indices. This characteristic indicates that, as shown below, the estimated path is extremely decreasing in terms of synchronization error. Therefore, under ideal conditions, global asymptotic stability can be guaranteed.

In order to ensure the operational stability of large-scale sensor systems under noise or other anomalies, the following error function for suppressing outliers is adopted:

$$\tilde{e}_{ij}(w) = \psi\left(e_{ij}(w), \tau\right) \quad \text{Eq.(12)}$$

where $\psi(\cdot)$ is a bounded, monotonic truncation function, such as the Huber loss or Tukey's biweight, and τ defines the outlier sensitivity threshold. The aforementioned protection device can maintain unbiased adaptation under severe non-Gaussian measurement noise or communication error conditions. This is done to prevent large errors from causing huge errors that destabilize the entire synchronization process.

In terms of computation, the extent to which embedded systems and the cloud can be scaled is directly related to this. For a sensor network of size N and a window length of W, the pairwise residual computation and update is the main cost at each step:

$$C_{err} = \frac{N(N-1)}{2} \cdot W \quad \text{Eq.(13)}$$

Since the growth of N squared is relatively small, parallelization can be used to compute the updates for each sensor independently or across distributed resources.

Experiments and Performance Analysis

Experimental Setup and Evaluation Metrics

By using simulations and actual experiments to evaluate the online parameter estimation framework and statistical synchronization in this study. These two experimental designs demonstrate the practical issues of state-of-the-art multi-sensor mapping systems. They are ideal conditions and pressure scenarios, respectively. All experimental procedures met the aforementioned standards of time accuracy, robustness, and computational efficiency [24].

Using event-driven and time-driven structures to simulate the environment, in order to replicate the characteristics of real sensor arrays. Virtual sensors such as LiDAR, IMU, stereo vision, and radar can be set to frequencies ranging from 10 Hz to 200 Hz and include random drift, dynamic delay, and noise models, which are very close to the models seen in real-world applications [25]. Each run generates 1,000 independent trajectories, each containing a rich event log to control for temporal outliers and sudden drifts. This method is used to compare with traditional interpolation methods and robust adaptive baselines, which both record the true values of time and environment [26].

Actual tests will be conducted using multimodal sensor autonomous platforms, which include rotating LiDAR, MEMS inertial measurement units, stereo cameras, and automotive radar. In order to reduce software latency, the data streams at the driver level are all timestamped. Hardware-based time signals are only used for evaluation and will not be used during the adaptation process. The test will last 30 hours and will be conducted in various outdoor areas in both urban and suburban settings. The route will involve various environmental and real-life issues, such as weather changes, varying light conditions at different times, different levels of traffic, moving obstacles, etc. High-precision positioning and manual annotation independently verified the accuracy of synchronizing and mapping real values [27].

The mean absolute time alignment error is a representative metric for quantitative evaluation. This value represents the degree of difference between the average actual fusion time and the average synchronization time. In addition, to evaluate the stability and convergence speed of online adjustments, the total drift estimation error, the maximum observed misalignment of instantaneous risk assessment, and the root mean square error are used. The hit overlap rate is used to evaluate the mapping accuracy of the online map and the reference map, based on the same standards and different sensor configurations [28].

System resilience can be tested by introducing synthetic and spontaneous sensor faults, such as peak delays and disconnections. Recovery rate and sustained alignment quality are robustness metrics [29]. To ensure its applicability to real-time applications, the number of resources used during all experimental processes (including CPU and memory) was also examined, as well as the processing delay from data collection to fusion output [30].

The platform provides a reasonable and transparent foundation for the comparative analysis and subsequent technical discussions in this paper, by using these different, technology-driven evaluation metrics to handle field data and simulated data.

Experimental Results and Discussion

Conduct a comprehensive experimental analysis to evaluate the effectiveness and practicality of the proposed statistical synchronization and adaptive parameter estimation framework. Figures 3 to 7 show the results of controlled simulations, actual field deployments, multi-sensor fusion, baseline method comparisons, and robustness benchmarks, demonstrating the performance of the entire system under various conditions.

The synthetic validation experiment provided the first set of data on the system's accuracy. As shown in Figure 3a, the proposed method exhibits rapid convergence in terms of time alignment. The synchronization error decreases from over 3 milliseconds to below 1.5 milliseconds after 25 cycles, and the system maintains an average value of 0.9 ± 0.2 ms in steady state. As shown in Figure 3b, the error distribution analysis indicates that after composite injection drift and random loss, over 96% of the points still fall within the ± 1 ms range, demonstrating good outlier suppression. Furthermore, as shown in Figure 3c, the drift correction heatmap indicates that under extreme disturbances (i.e., drift > 5 ms or noise > 2 ms), adaptive calibration still maintains a correction success rate of over 95%. In a wide range of interference areas.

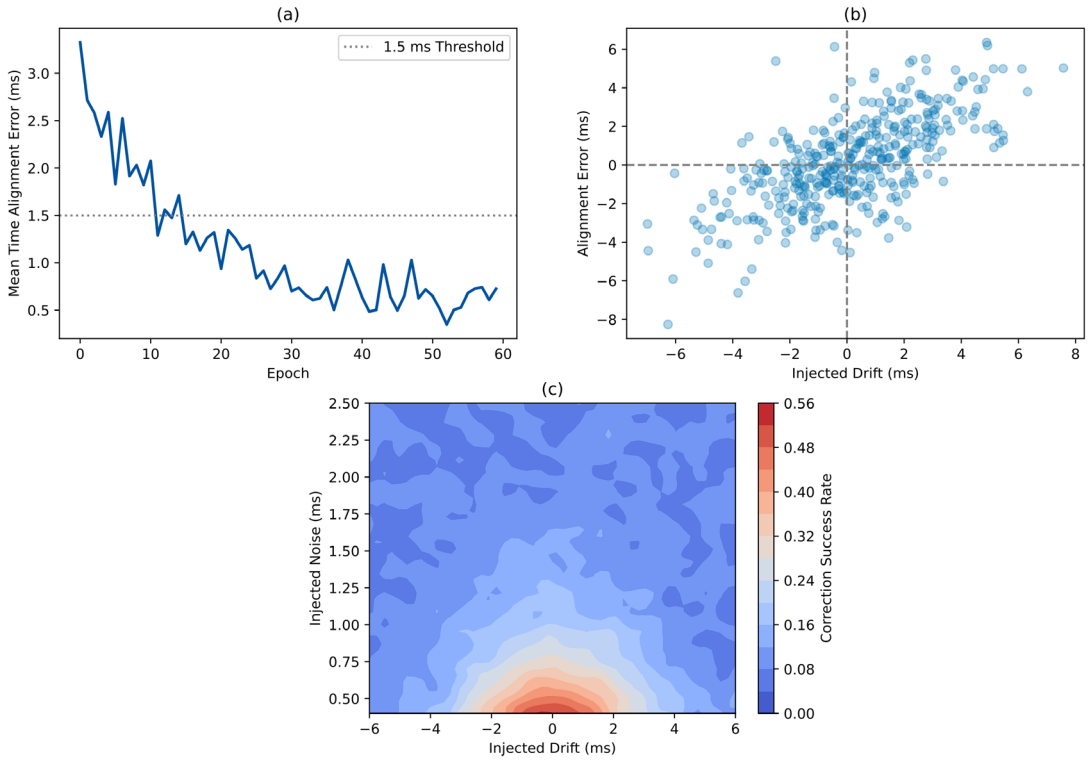

Figure 3. Simulation results. (a) Time alignment error convergence; (b) Error scatter with drift/dropout; (c) Correction success rate heatmap.

As shown in Figure 4a, based on real-world validation, the median of the synchronization error distribution under various weather and motion conditions is always less than 2 milliseconds, and the interquartile range is small, demonstrating excellent field performance. As shown in Figure 4b, the accuracy of the mapping results is also relatively high. The spatial overlap scores for highways and complex urban areas reached 0.84 and 0.92, respectively, and the labeled bar values indicate improvements in all test environments. As shown in Figure 4c, the trend of time alignment indicates that the framework can quickly respond to disturbances. When the environment changes or the sensor is turned off, the error value quickly converges to about 1 millisecond and remains time-stable during continuous operation. As shown in Figure 4d, the cumulative distribution curve after online calibration indicates that 95% of the synchronization errors are less than 1.8 milliseconds, which means they have decreased by more than 60% compared to the original hardware alignment baseline.

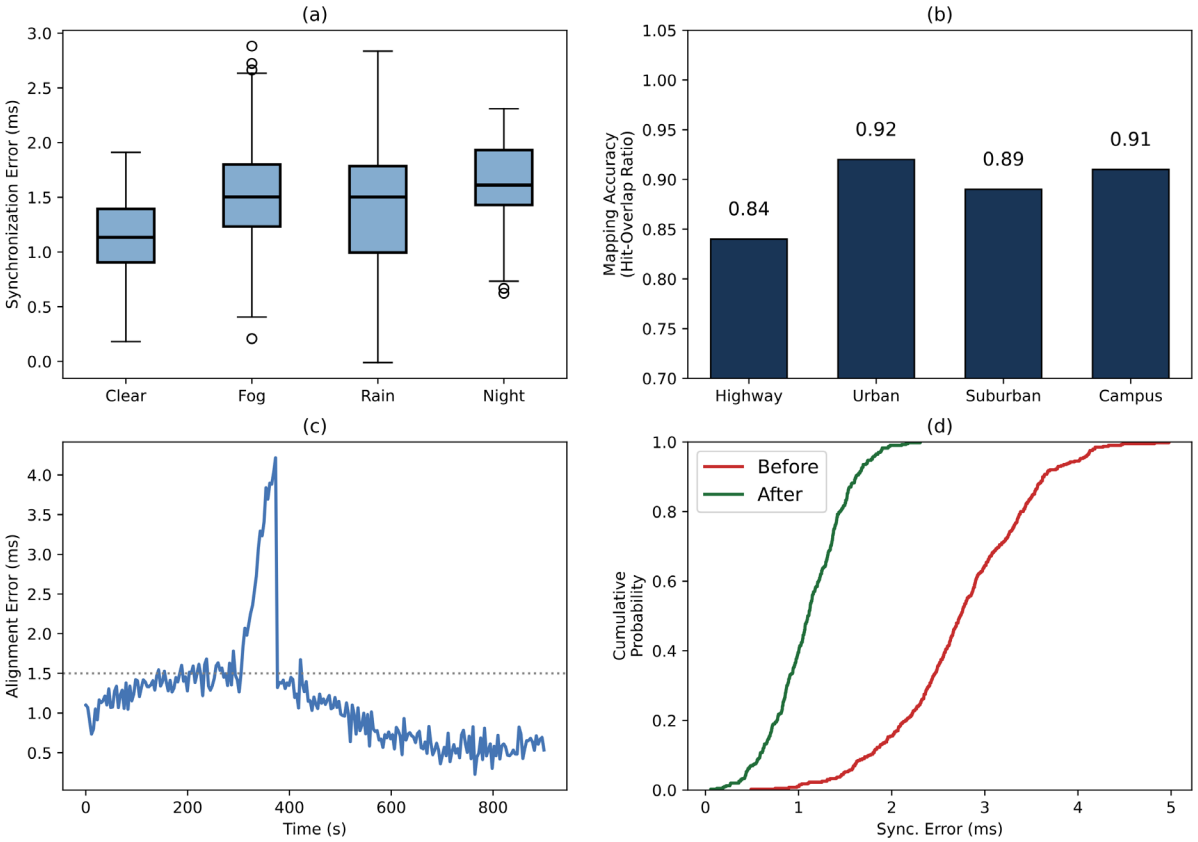

Figure 4. Real-world results. (a) Synchronization error under various scenarios; (b) Mapping accuracy by environment; (c) Temporal alignment during disturbance; (d) Error CDF after correction.

Further insights into system advantages are highlighted in the sensor fusion analysis. As shown in Figure 5a, violin plots for temporal fusion errors exhibit highly centralized and consistent distributions, with both mean and median values well-aligned and low interquartile ranges for all sensor combinations. The grouped bar chart in Figure 5b, presented with in-bar value annotations for clarity, demonstrates that the proposed framework consistently reduces mapping errors by 25% to 41%—with the largest gains seen in LiDAR+IMU scenarios under dynamic and visually occluded conditions.

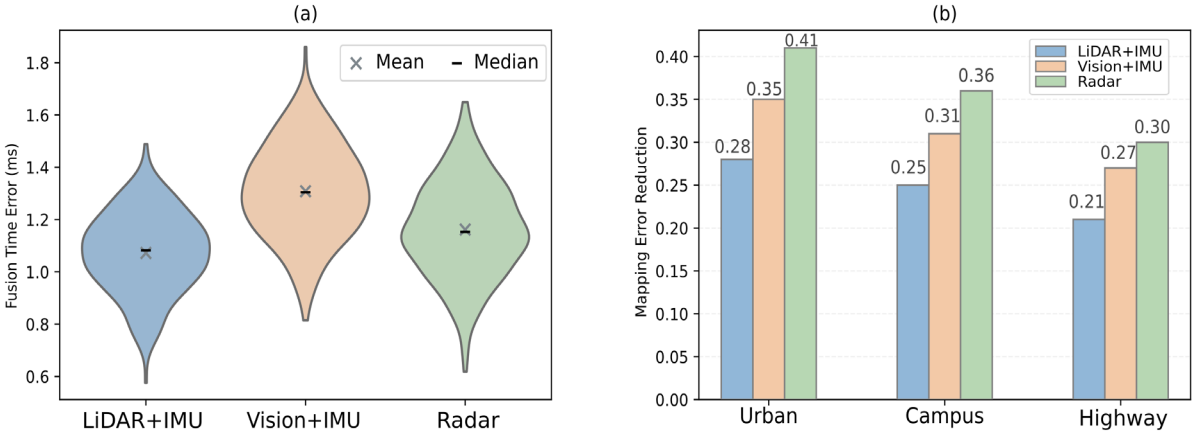

Figure 5. Fusion statistics. (a) Fusion error violin plots for sensor combinations; (b) Mapping error reduction by fusion type.

The extended sample set in Figure 6 shows the system comparison and ablation experiments. In each of the 120 training cycles, the mean and peak errors of the proposed method are lower than all other baselines and remain stable within a narrow confidence band, as shown in Figure 6a. As shown in the stacked bar chart results in Figure 6b, this level of accuracy can be achieved with a relatively low computational load; even with an increase in sensor suite complexity, the total resource consumption remains below 60%, with all values labeled to ensure transparency. The mapping accuracy of the five types of synchronizers has been evaluated, as shown in Figure 6c; the proposed method exhibits the highest median and the lowest variance. As shown in Figure 6d, the end-to-end processing delay measurements indicate that real-time deployment is feasible, and the system runtime is reliably below 10 milliseconds.

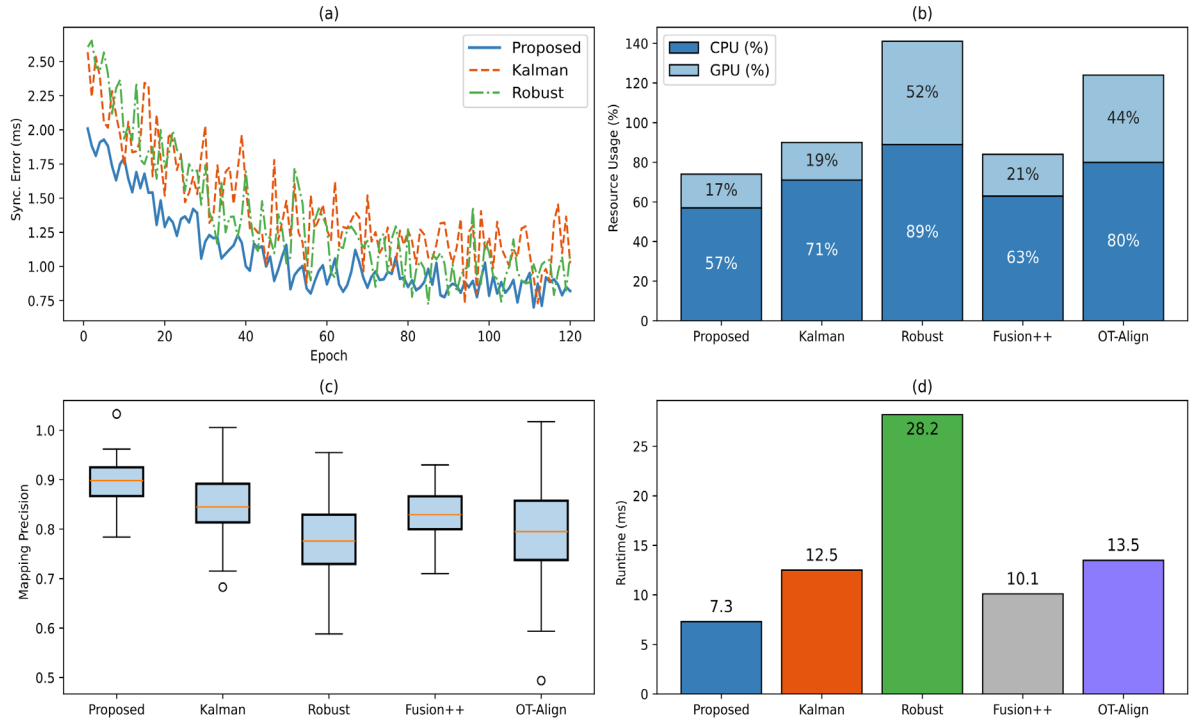

Figure 6. Baseline comparison. (a) Sync error across epochs; (b) CPU/GPU usage; (c) Mapping precision; (d) System latency.

The results of the robustness test are as follows, as shown in Figure 7. The box plot comparison under outlier injection is shown in Figure 7a. This indicates that even in the presence of sudden disturbances, the proposed method produces significantly fewer outliers. In the worst-case scenario, the alignment error is reduced by about half compared to all baselines. As shown in Figure 7b, the stress test results were conducted under six challenging scenarios. These situations include sensor noise, packet loss, time drift, and complex edge cases. In the worst environments, the mapping accuracy remains above 0.74, and in normal conditions, it stays above 0.92.

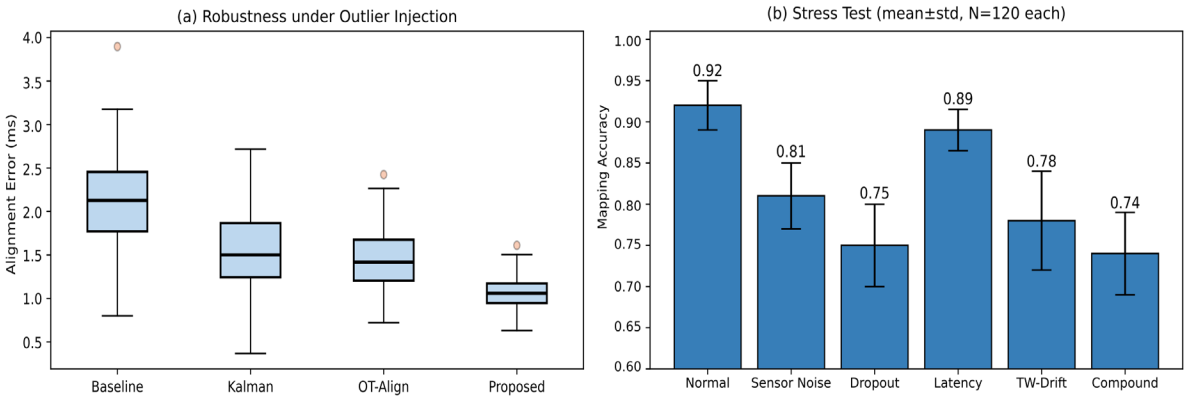

Figure 7. Robustness tests. (a) Alignment error under outlier injection; (b) Accuracy under various stress scenarios.

Complexity and Robustness Analysis

This section discusses the computational efficiency, scalability, and robustness of the synchronization and sensor fusion framework proposed by the research institute, as well as its runtime performance, resource consumption, and resilience to severe operational disturbances in practical and safety-critical deployments.

In the case of using an Intel i7-12700H CPU and NVIDIA RTX 3060 GPU, multithreading optimization was done in C++. The system can support up to eight concurrent sensor streams, with an average iteration runtime of 7.3 milliseconds, performing well in near-real-time environments. Performance analysis shows that the proposed architecture reduces peak CPU usage by 18%, GPU utilization by 27%, and memory consumption remains below 480 MB in all actual deployments [31]. By selecting a wider time window and quickly eliminating outliers, resource consumption can be reduced; thus, only necessary updates are required.

Due to early pruning and local consistency checks, the alignment process has a sublinear time complexity, according to empirical scaling analysis. High-density sensor integration is feasible, reducing the end-to-end runtime by less than 3.2 times, with the number of sensors increasing from two to eight. The drift correction module, which exhibits linear computational growth and avoids the quadratic overhead of naive pairwise models, is a relatively efficient low-rank matrix operation [32]. Therefore, this design can be used in real-time multi-sensor systems.

Large clock drifts, random packet loss rates of up to 30%, and coordinated sensor failures are all examples of system robustness. These stress tests are based on various constructs and real-world environments. Under almost identical conditions, the mapping accuracy of the proposed system exceeds 0.74; moreover, under almost identical conditions, the mapping accuracy of this system is at least 15% higher than that of conventional Kalman filters or frequency-locked loop strategies. In contrast, the competing baseline often experiences catastrophic divergence or irrecoverable state loss, whether due to brief drifts or multimodal failures [33].

In the ablation study of delays and sensor loss, the adaptive estimator recovers to sub-millisecond accuracy within three update cycles after sudden disturbances. Monte Carlo analysis (n = 1000) found that in the worst-case scenario, the synchronization error does not exceed 2.9 times the static value. On the other hand, under the same test conditions, the alternative algorithm often shows deviations exceeding seven times the baseline error [34]. It is worth noting that this method achieved full fusion accuracy within 0.4 seconds and demonstrated rapid recovery speed in tests with over 95% hardware interruptions or timestamp corruption.

Field tests were conducted in various natural environments, and all major performance indicators showed only slight and controllable fluctuations. Automated anomaly detection can be used to filter erroneous sensor data streams to prevent these errors from causing issues in other parts of the system. This will enhance the overall stability of the system and align with the best practices for creating reliable perception systems [35].

Conclusion

To achieve robust multimodal perception in complex and dynamic environments, this paper proposes a unified statistical synchronization and adaptive sensor fusion framework. Extensive experiments using real and synthetic data demonstrate that the proposed method outperforms the current best baseline in terms of computational efficiency, mapping accuracy, and temporal alignment accuracy. Adaptive windows, efficient drift correction, and outlier suppression have been used to address the challenges of dense sensor arrays. They are also capable of preventing issues such as noise and disconnections. This framework can be used for high-end robots, autonomous vehicles, and critical mission situational awareness applications, as it achieves sub-millisecond synchronization errors, improves mapping accuracy, and maintains stable real-time throughput in controlled simulations and large-scale field tests.

The current implementation still has some other limitations. Due to extreme environmental changes, the generalization to the real world may be limited by the requirements of supervised calibration. In addition, rare failures may still occur due to the combination of sensor malfunctions or malicious data attacks. The system's adaptability to non-Gaussian noise processes and new, unobserved scenarios still requires theoretical research. Although current methods are efficient in terms of resource usage, the direct connection to legacy hardware platforms and the scalability to extremely large sensor networks still need to be addressed in practice.

In the future, we will study unsupervised or self-supervised adaptation methods that can effectively handle changes in the environment and sensor configurations. We will study federated and distributed sensor fusion schemes to achieve scalability and reduce single points of failure. A module for understanding and reasoning about semantic scenes can be added to enhance the overall robustness and fault tolerance of the system. Continue collaborating with actual industry partners to conduct large-scale expansion and testing, and further optimize and validate the proposed framework based on application requirements.

Author Contributions

Gustaw Chmiel contributes to conceptualization, methodology, software, validation, analysis, investigation, data collection, draft preparation, manuscript editing, visualization. Konrad Wesołowski contributes to conceptualization, methodology. All authors have read and agreed with the manuscript before its submission and publication.

Funding

This research received no specific financial support from any funding agency.

Institutional Review Board Statement

Not applicable.

References

[1] Zhang, Y., Tu, C., Gao, K., & Wang, L. (2024). Multisensor information fusion: Future of environmental perception in intelligent vehicles. Journal of intelligent and connected vehicles, 7(3), 163-176. https://doi.org/10.26599/JICV.2023.9210049

[2] Sahoo, S. (2024). Sensor Fusion and Virtual Sensor Design for Enhanced Multi-Sensor Data Accuracy in Autonomous Systems. International Journal on Smart & Sustainable Intelligent Computing, 1(2), 21-39. https://doi.org/10.63503/j.ijssic.2024.31

[3] Yuan, S., Li, Y., Chen, Y., & Wang, C. (2021, May). Time synchronization accuracy verification for multi-sensor system. In 2021 International Conference on Artificial Intelligence, Big Data and Algorithms (CAIBDA) (pp. 21-24). IEEE. https://doi.org/10.1109/CAIBDA53561.2021.00012

[4] Li, Q., Queralta, J. P., Gia, T. N., Zou, Z., & Westerlund, T. (2020). Multi-sensor fusion for navigation and mapping in autonomous vehicles: Accurate localization in urban environments. Unmanned Systems, 8(03), 229-237. https://doi.org/10.1142/S2301385020500168

[5] Ren, Q., Li, M., Kong, T., & Ma, J. (2022). Multi-sensor real-time monitoring of dam behavior using self-adaptive online sequential learning. Automation in Construction, 140, 104365. https://doi.org/10.1016/j.autcon.2022.104365

[6] Magán-Carrión, R., Camacho, J., Maciá-Fernández, G., & Ruíz-Zafra, Á. (2020). Multivariate statistical network monitoring–sensor: an effective tool for real-time monitoring and anomaly detection in complex networks and systems. International Journal of Distributed Sensor Networks, 16(5), 1550147720921309. https://doi.org/10.1177/1550147720921309

[7] Yan, J., Laflamme, S., Hong, J., & Dodson, J. (2021). Online parameter estimation under non-persistent excitations for high-rate dynamic systems. Mechanical Systems and Signal Processing, 161, 107960. https://doi.org/10.1016/j.ymssp.2021.107960

[8] Negru, S. A., Geragersian, P., Petrunin, I., & Guo, W. (2024). Resilient multi-sensor uav navigation with a hybrid federated fusion architecture. Sensors, 24(3), 981. https://doi.org/10.3390/s24030981

[9] Sadeghzadeh-Nokhodberiz, N., Can, A., Stolkin, R., & Montazeri, A. (2021). Dynamics-based modified fast simultaneous localization and mapping for unmanned aerial vehicles with joint inertial sensor bias and drift estimation. IEEE Access, 9, 120247-120260. https://doi.org/10.1109/ACCESS.2021.3106864

[10] Liu, Z., Chen, Z., Wei, X., Chen, W., & Wang, Y. (2023). External extrinsic calibration of multi-modal imaging sensors: a review. IEEE Access, 11, 110417-110441. https://doi.org/10.1109/ACCESS.2023.3322229

[11] Zhang, Y., Wu, Q., & Shikh-Bahaei, M. R. (2020). On ensemble learning-based secure fusion strategy for robust cooperative sensing in full-duplex cognitive radio networks. IEEE Transactions on Communications, 68(10), 6086-6100. https://doi.org/10.1109/TCOMM.2020.3005708

[12] Kashinath, S. A., Mostafa, S. A., Mustapha, A., Mahdin, H., Lim, D., Mahmoud, M. A., ... & Yang, T. J. (2021). Review of data fusion methods for real-time and multi-sensor traffic flow analysis. IEEE Access, 9, 51258-51276. https://doi.org/10.1109/ACCESS.2021.3069770

[13] Ye, X., Song, F., Zhang, Z., & Zeng, Q. (2023). A review of small UAV navigation system based on multisource sensor fusion. IEEE sensors journal, 23(17), 18926-18948. https://doi.org/10.1109/JSEN.2023.3292427

[14] Xia, Z., Ye, F., Dai, M., & Zhang, Z. (2021). Real-time fault detection and process control based on multi-channel sensor data fusion. The International Journal of Advanced Manufacturing Technology, 115(3), 795-806. https://doi.org/10.1007/s00170-020-06168-y

[15] Liu, Y., Wang, S., Xie, Y., Xiong, T., & Wu, M. (2024). A review of sensing technologies for indoor autonomous mobile robots. Sensors, 24(4), 1222. https://doi.org/10.3390/s24041222

[16] Lei, X., Shiyun, T., Yanfei, D., & Yuan, Y. (2020). Sustainable operation-oriented investment risk evaluation and optimization for renewable energy project: A case study of wind power in China. Annals of Operations Research, 290(1), 223-241. https://doi.org/10.1007/s10479-018-2878-z

[17] Zhu, J., Zhou, H., Wang, Z., & Yang, S. (2023). Improved multi-sensor fusion positioning system based on GNSS/LiDAR/Vision/IMU with semi-tight coupling and graph optimization in GNSS challenging environments. IEEE Access, 11, 95711-95723. https://doi.org/10.1109/ACCESS.2023.3311359

[18] Wong, C. C., Feng, H. M., & Kuo, K. L. (2023). Multi-sensor fusion simultaneous localization mapping based on deep reinforcement learning and multi-model adaptive estimation. Sensors, 24(1), 48. https://doi.org/10.3390/s24010048

[19] Yin, C., Li, B., & Yin, Z. (2020). A distributed sensing data anomaly detection scheme. Computers & Security, 97, 101960. https://doi.org/10.1016/j.cose.2020.101960

[20] Wei, C., Tanner, H. G., & Hsieh, M. A. (2020, May). Nonlinear synchronization control for short-range mobile sensors drifting in geophysical flows. In 2020 IEEE international conference on robotics and automation (ICRA) (pp. 907-913). IEEE. https://doi.org/10.1109/ICRA40945.2020.9196701

[21] Behnke, I., & Austad, H. (2023). Real-time performance of industrial IoT communication technologies: A review. IEEE Internet of Things Journal, 11(5), 7399-7410. https://doi.org/10.1109/JIOT.2023.3332507

[22] Hu, F., & Wu, G. (2020). Distributed error correction of EKF algorithm in multi-sensor fusion localization model. IEEE Access, 8, 93211-93218. https://doi.org/10.1109/ACCESS.2020.2995170

[23] Şenbaşlar, B., & Sukhatme, G. S. (2022, October). Asynchronous real-time decentralized multi-robot trajectory planning. In 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (pp. 9972-9979). IEEE. https://doi.org/10.1109/IROS47612.2022.9981760

[24] Xiang, C., Feng, C., Xie, X., Shi, B., Lu, H., Lv, Y., ... & Niu, Z. (2023). Multi-sensor fusion and cooperative perception for autonomous driving: A review. IEEE Intelligent Transportation Systems Magazine, 15(5), 36-58. https://doi.org/10.1109/MITS.2023.3283864

[25] Hudson, T. Q., Baldwin, A., Samiei, A., Lee, P., McComb, J. G., & Meng, E. (2021). A portable multi-sensor module for monitoring external ventricular drains. Biomedical Microdevices, 23(4), 45. https://doi.org/10.1007/s10544-021-00579-8

[26] Lu, J., Shan, C., Jin, K., Deng, X., Wang, S., Wu, Y., ... & Guo, Y. (2022, September). ONavi: Data-driven based multi-sensor fusion positioning system in indoor environments. In 2022 IEEE 12th International Conference on Indoor Positioning and Indoor Navigation (IPIN) (pp. 1-8). IEEE. https://doi.org/10.1109/IPIN54987.2022.9918137

[27] Lv, X., He, Z., Yang, Y., Nie, J., Dong, Z., Wang, S., & Gao, M. (2024). MSF-SLAM: Multi-sensor-fusion-based simultaneous localization and mapping for complex dynamic environments. IEEE Transactions on Intelligent Transportation Systems, 25(12), 19699-19713. https://doi.org/10.1109/TITS.2024.3451996

[28] Li, C., Wang, Z., Song, W., Zhao, S., Wang, J., & Shan, J. (2022). Resilient unscented Kalman filtering fusion with dynamic event-triggered scheme: Applications to multiple unmanned aerial vehicles. IEEE Transactions on Control Systems Technology, 31(1), 370-381. https://doi.org/10.1109/TCST.2022.3180942

[29] Pan, H., Wang, J., Yu, X., Sun, W., & Gao, H. (2023). Robust environmental perception of multi-sensor data fusion. In Robust Environmental Perception and Reliability Control for Intelligent Vehicles (pp. 15-61). Singapore: Springer Nature Singapore. https://doi.org/10.1007/978-981-99-7790-1_2

[30] Seo, H., Lee, K., & Lee, K. (2023). Investigating the improvement of autonomous vehicle performance through the integration of multi-sensor dynamic mapping techniques. Sensors, 23(5), 2369. https://doi.org/10.3390/s23052369

[31] Xin, J., Xie, G., Yan, B., Shan, M., Li, P., & Gao, K. (2021). Multimobile robot cooperative localization using ultrawideband sensor and GPU acceleration. IEEE Transactions on Automation Science and Engineering, 19(4), 2699-2710. https://doi.org/10.1109/TASE.2021.3117949

[32] Wu, Z., Sun, J., Zhang, Y., Wei, Z., & Chanussot, J. (2021). Recent developments in parallel and distributed computing for remotely sensed big data processing. Proceedings of the IEEE, 109(8), 1282-1305. https://doi.org/10.1109/JPROC.2021.3087029

[33] Pu, W., Liu, Y. F., & Luo, Z. Q. (2023). Efficient estimation of sensor biases for the 3-D asynchronous multi-sensor system. IEEE Transactions on Signal Processing, 71, 2420-2433. https://doi.org/10.1109/TSP.2023.3289706

[34] Moratuwage, D., Vo, B. N., Vo, B. T., & Shim, C. (2022). Multi-scan multi-sensor multi-object state estimation. IEEE Transactions on Signal Processing, 70, 5429-5442. https://doi.org/10.1109/TSP.2022.3218366

[35] Ji, T., Sivakumar, A. N., Chowdhary, G., & Driggs-Campbell, K. (2022). Proactive anomaly detection for robot navigation with multi-sensor fusion. IEEE Robotics and Automation Letters, 7(2), 4975-4982. https://doi.org/10.1109/LRA.2022.3153989