ROS ORTAMINDA OTONOM ARAÇ LOKALİZASYONU İÇİN GERÇEK ZAMANLI GPS VE IMU VERİSİ İŞLEME

Cem Tolga MÜNYAS
Tofaş Türk Otomobil Fab. A.Ş, R&D İzmir Branch, İzmir / Türkiye
ORCID: 0009-0002-9878-9566

ÖZET

Doğru lokalizasyon, otonom sürüşün kritik bir bileşeni olmasıyla beraber araçların dinamik olarak değişen ortamlarda güvenli ve verimli bir şekilde seyretmesini sağlar. Küresel Konumlandırma Sistemi (GPS) verileri ve Atalet Ölçüm Ünitesi (IMU) verileri, sensör füzyon teknikleriyle birleştirildiğinde, otonom sürüş uygulamaları için araç konumunun, yönünün ve yörünge planlamasının belirlenmesinde temel bir rol oynar. Bu proje, Robot İşletim Sistemi (ROS2) çerçevesinde gerçek zamanlı bir GPS veri simülasyonu ve sensörlerden gelen veriyi işleyerek farklı mesaj tiplerine dönüştürmeyi hedeflemektedir. Proje uygulamasında GPS ve IMU sensör verilerini araç üzerinden gerçek zamanlı olarak topladıktan sonra gerekli koordinat dönüşümlerini yapar ve işlenen veri akışlarını GPS, IMU, araç pozu, hız ve GNSS-INS oryantasyonu dahil olmak üzere ilgili ROS konularına yayınlar. Bunlara ek olarak, otonom araca monte edilmiş sensörlerin arasındaki uzamsal ilişkileri korumak için dönüşümler yapar ve aynı şekilde bunu ilgili ROS konusuna yayınlar. Veri akışı dönüşümleri, GPS verilerinin coğrafi koordinatlardan yerel Kartezyen çerçevelere dönüştürülmesini, sensör çıktılarının aracın referans çerçevesiyle hizalanmasınu ve lokalizasyondaki gelişmiş doğruluk için zaman senkronizasyonunun korunmasını içerir. Proje kapsamında geliştirilen ROS2 paketi, hem canlı sensör verisinden hem de önceden kaydedilmiş veri kümelerinden veri alımını destekleyecek şekilde dizayn edildiğinden simülasyon ortamında geliştirme ve test etme esnekliği sağlar. Sistem, tanımlı ROS mesajlarını kullanarak ve sensör kalibrasyonu uygulayarak, GPS konumlandırma ve IMU bozulmasındaki hataları düzeltir; sensör füzyonu, otonom navigasyon ve yörünge izleme gibi otonom sürüşü güçlendiren uygulamaları destekleyen yerel lokalizasyonun doğruluğunu artırır. Özetlemek gerekirse, projenin çıktıları otonom araçlarda lokalizasyon ve sensör füzyonu uygulamaları için önemli ve uyarlanabilir işlenmiş veri akışı sağlamaktadır.

Anahtar Kelimeler: Lokalizasyon, Otonom Sürüş, Robot İşletim Sistemi, GPS Veri Paylaşımı, Otonom Araçlar

REAL-TIME GPS AND IMU DATA PROCESSING FOR AUTONOMOUS VEHICLE LOCALIZATION IN ROS FRAMEWORK

ABSTRACT

Accurate localization is a critical component of autonomous driving, enabling vehicles to navigate safely and efficiently in dynamically changing environments. Global Positioning System (GPS) data, and IMU (Inertial Measurement Unit) data combined with sensor fusion techniques, play a fundamental role in determining vehicle position, orientation, and trajectory planning for autonomous driving applications. This project provides a robust GPS data simulation and publishing tool within the Robot Operating System (ROS2) framework. The project facilitates the real-time collection of GPS and IMU data, applies necessary coordinate transformations, and publishes the processed information (localization-related data streams) to the relevant ROS topics including GPS, IMU, vehicle pose, velocity and GNSS-INS orientation. Additionally, it broadcasts transformations to maintain spatial relationships between different frames of the autonomous vehicle. The transformation process includes converting GPS data from geographic coordinates to local Cartesian frames, aligning sensor outputs with the vehicle's reference frame, and maintaining time synchronization for improved accuracy. The developed package supports data retrieval from live sensors and pre-recorded datasets, ensuring flexibility in development and testing in the simulation environment. By leveraging well-defined ROS messages and applying sensor calibration, the system enhances localization accuracy, correcting errors in GPS positioning and IMU drift, which supports applications in the context of autonomous driving such as sensor fusion, autonomous navigation, and trajectory tracking. All in all, the project provides robust and adaptable processed data streams that are crucial for localization and sensor fusion applications in autonomous vehicles in real-world scenarios**.**

Keywords: Localization, Autonomous Driving, Robot Operating System, GPS Data Publisher, Autonomous Vehicles2

1. INTRODUCTION

In autonomous driving applications, the localization of the robot is an important concept. The robot which is the autonomous vehicle here relies on the localization stack within the autonomous driving system. Accurate and real-time localization is a fundamental requirement for autonomous driving systems. Autonomous vehicles must constantly determine their position, orientation, and movement velocity to safely navigate and interact with their surroundings. These localization processes rely on multiple sensors including GNSS (GPS), IMUs, LiDARs, and cameras. In this study, we emphasize the real-time processing of GPS and IMU data, which are often used together due to their complementary characteristics. GPS provides global positioning while the IMU offers high-rate motion tracking.

However, despite the critical nature of this sensor data, raw outputs from GPS and IMU sensors are often not directly usable in robotics systems like those built with the Robot Operating System (ROS). They may use incompatible coordinate systems, different data formats, and unsynchronized timestamps. Furthermore, sensor outputs need to be transformed into a coherent local coordinate frame, and they must be synchronized to ensure consistent time-aligned data fusion.

This project addresses these issues by designing and implementing a robust ROS2 package that integrates GPS and IMU data to generate reliable localization information. The developed node processes sensor streams from pre-recorded datasets (and soon live sensors), transforms coordinate frames, aligns timestamps, and publishes standardized ROS messages, which are directly compatible with Autoware, a widely-used open-source autonomous driving software stack.

2. MATERIALS AND METHODS

2.1 Tools and Autonomous Driving Software Stack

The entire software architecture relies on ROS2, a flexible middleware platform that facilitates communication between modular software components known as "nodes". This modular design allows seamless integration with existing autonomous systems.

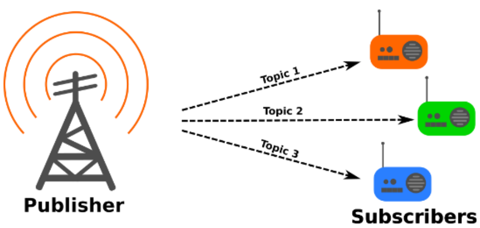

Figure 1. ROS2 Publisher-Subscriber working mechanism [7]

For the autonomous driving software stack side, the project utilizes Autoware, an open-source autonomous driving framework, serves as the foundational stack. It provides capabilities such as mapping, localization, perception, planning and control as can be seen on its high-level architecture.

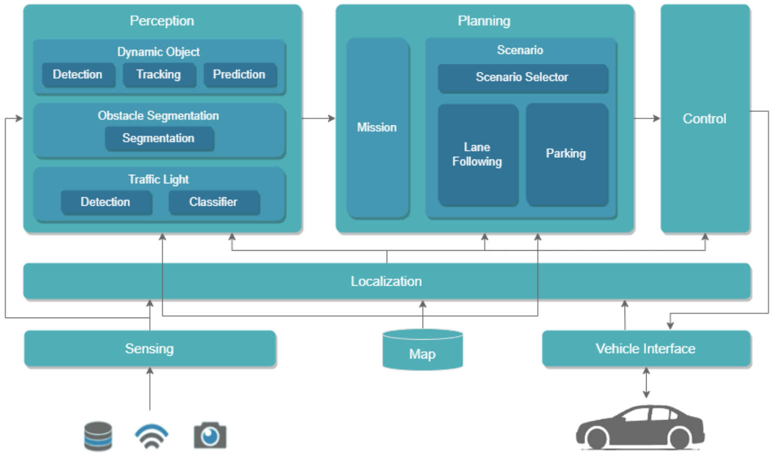

Figure 2. High-level architecture of Autoware [1]

The GPS and IMU datasets used in this work are collected from an experimental platform equipped with GNSS receivers and an Inertial Measurement Unit. These datasets were recorded in CSV format and represent realistic scenarios encountered in vehicle environments.

For development, testing and visualization processes throughout the project, wu utilized Autoware for autonomous software stack support, ROS2 Humble for middleware and node orchestration, Rviz and RQT Graph for real-time visualization of sensor behaviors and node-topic interactions, GeographicLib for coordinate transformation and also custom calibration files for sensor alignment.

2.2 ROS2 Node & Key Functionalities

The core contribution of this project is a ROS2 package featuring three main functionalities. The first key component is the multi-format input support of the package. The node can read GPS and IMU data from CSV files (currently supported), with live sensor streaming support under development. It parses key parameters such as timestamp, position (latitude/longitude), orientation angles, velocity, and acceleration. Second key component is the coordinate transformation. The collected raw GPS and IMU data is transformed from the Military Grid Reference System (MGRS) to the Universal Transverse Mercator (UTM) format. This is critical because MGRS, while human-readable, is not suitable for metric-based robotic computations. UTM, with its Cartesian characteristics, simplifies path planning and sensor fusion tasks.

Additionally, extrinsic calibration is applied to align sensor outputs with the vehicle's local frame. The last key component of this package is that we publish new messages into seven different ROS topics that is newly created and can be mainly used for localization stack enhancements. After synchronization and transformation, the data is published to seven different standardized ROS2 topics. These topics are utilized by other modules for real-time localization, navigation, and mapping purposes.

3. RESULTS AND DISCUSSIONS

3.1 Data Flow Pipeline

The data flow pipeline in this project represents the step-by-step process of converting raw GPS and IMU sensor data that is collected with a vehicle in traffic into meaningful, synchronized, and transformed localization messages that can be utilized by autonomous driving systems. This pipeline consists of four main stages.

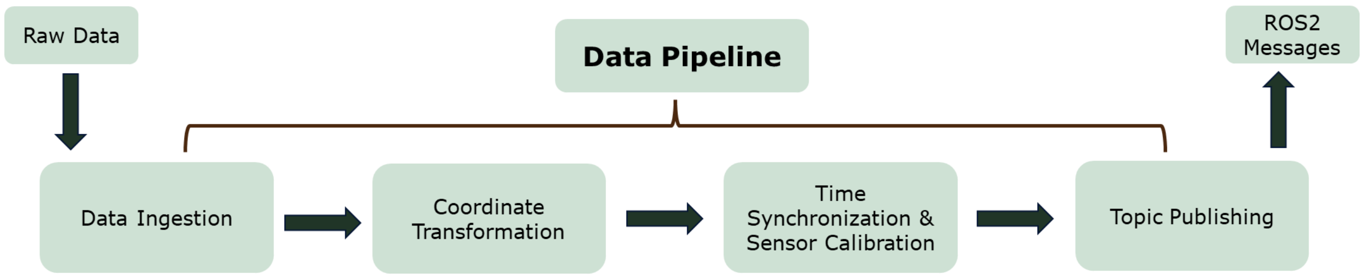

Figure 3. Data flow pipeline

3.1.1 Data Ingestion

The first stage is "Data Ingestion". The aim here is to read and parse the raw GPS and IMU data stored in a CSV file. The subprocesses for the first stage can be divided into three different categories: CSV parsing, parameter extraction, and timestamp indexing. The node begins by ingesting raw data from a structured CSV file. Each row in the CSV corresponds to a single timestamped observation containing information from either GPS or IMU sensors. Then, in the paremeter extraction, fields such as timestamp, latitude, longitude, orientation (roll, pitch, yaw), linear velocity, and acceleration are extracted. These parameters are critical for downstream processing. Lastly, each entry is assigned its precise timestamp to facilitate later synchronization. GPS data typically comes at a lower rate (e.g., 1–5 Hz), while IMU data is available at higher frequencies (e.g., 50–100 Hz), which makes accurate timestamp management crucial.

3.1.2 Coordinate Transformation

The second step of the data pipeline is “Coordinate Transformation”. The aim here is to convert GPS-based geodetic coordinates into a local Cartesian coordinate system (UTM), and align them with the vehicle's reference frame so that robot could clarify the exact location on the global map with this information. We can deeply analyze the coordinate transformation in four subtopics. The first topic is the “Geodetic to Cartesian Conversion”. The node uses GeographicLib, an open-source geographic computation library, to convert Military Grid Reference System (MGRS) or Latitude/Longitude values into Universal Transverse Mercator (UTM) coordinates [3]. This is necessary due to different capabilities of two coordinate reference systems that can be seen on the comparison table. UTM coordinates allows easier metric computations such as distance and angle measurements for our case.

MGRS (Military Grid Reference System)	Cartesian UTM (Universal Transverse Mercator)
Used by NATO militaries and field operations	Used for general mapping, engineering & robotics applications
Human-readable (easier to read/communicate)	Detailed but longer readability
Derived from UTM coordinates	Converts directly from latitude / longitude

Table 1. Comparison of MGRS and Cartesian UTM coordinate systems

The second subtopic is ENU (East-North-Up) transformation. In cases where local ENU (East-North-Up) frame is required, additional transformations are applied to orient the coordinates relative to a known reference origin [3]. Then, another subtopic is the static frame transform application. These transformations are applied to map sensor data from sensor-specific frames (e.g., gnss_link, lidar_link) to the main vehicle frame (base_link). This ensures that all sensor data is spatially aligned. On the other hand, there is also dynamic transform publishing application. If the vehicle is in motion, dynamic transform broadcasters are used to publish transformations in real-time, linking coordinate frames for continuous navigation.

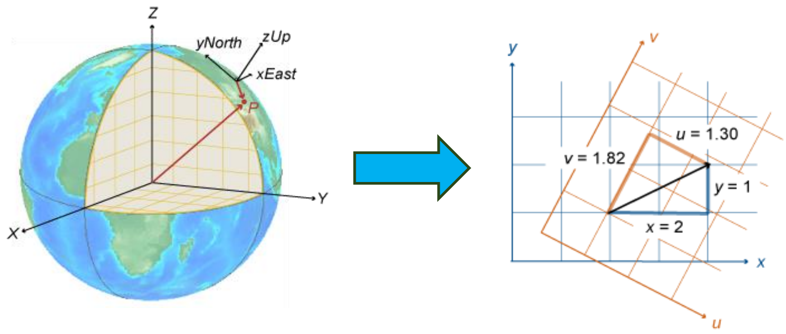

Figure 4. Coordinate transformation from the global ECEF system to a local ENU system [8]

3.1.3 Time Synchronization and Sensor Calibration

The third step of the data pipeline is time synchronization and sensor calibration. The goal is to align sensor streams from GPS and IMU and apply calibration parameters to relate their physical positions and orientations. To ensure reliable and precise localization, it is critical to synchronize the data streams from the GPS and IMU sensors while applying accurate sensor calibration parameters. In this project, GPS data—typically updated at a lower frequency—is aligned with high-rate IMU data using ROS2's built-in timing utilities. This alignment allows both sensors to contribute meaningfully to the same time-stamped localization frame. Without synchronization, discrepancies between timestamps can lead to inconsistent or drifting position estimates, especially during dynamic vehicle motion. Once temporal alignment is established, the next step involves calibrating the physical arrangement of sensors on the vehicle. This is done using a configuration file containing extrinsic calibration parameters, which define the relative position (translation) and orientation (rotation) between each sensor (e.g., GNSS, IMU, LiDAR) and the vehicle's base reference frame. Applying these transformations ensures that all sensor data is mapped into a unified spatial frame, allowing for accurate fusion and interpretation across the system. This calibration step is particularly crucial for tasks like pointcloud alignment, motion estimation, and trajectory tracking. Proper synchronization and calibration thus form the backbone of any effective localization system, ensuring that the input data accurately reflects the real-world vehicle dynamics in both time and space.

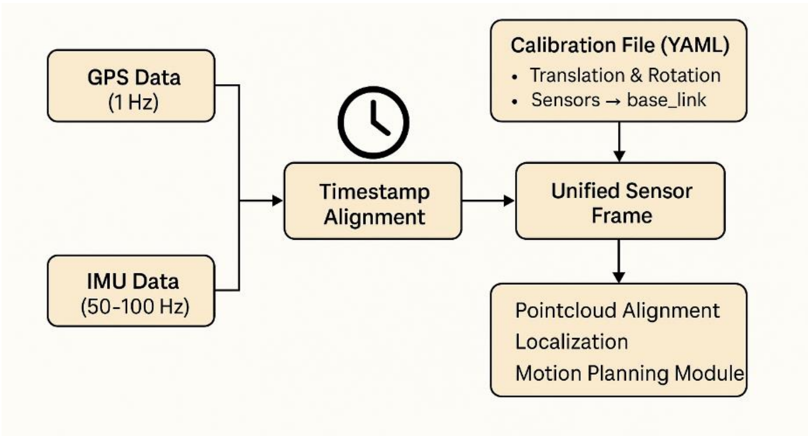

Figure 5. Sensor time synchronization and calibration flow

3.1.4 Topic Publishing

The final process of the work is publishing new localization related messages into standardized ROS topics. After the GPS and IMU data are synchronized and calibrated, the final stage involves publishing this processed information as ROS2 topics for use by other nodes in the autonomous vehicle system. This step acts as a bridge between sensor-level data processing and high-level modules like localization, mapping, and path planning. In the developed pipeline, GPS data—typically in the form of latitude, longitude, and altitude—is first converted into UTM (Universal Transverse Mercator) coordinates to allow for more accurate metric-based positioning in a Cartesian frame. This information, along with the high-frequency orientation and acceleration data from the IMU, is encapsulated in ROS2-compatible message types such as sensor_msgs/NavSatFix, geometry_msgs/PoseStamped, or sensor_msgs/Imu. These messages are then published to predefined topics at consistent rates. Maintaining a consistent and correct message structure ensures seamless integration with downstream nodes like localization algorithms, visualization tools (e.g., RViz), and control systems. Additionally, quality-of-service (QoS) profiles are configured to guarantee reliable communication, particularly for time-critical data streams. Through this architecture, the vehicle gains a continuous and accurate stream of spatial awareness, enabling it to respond intelligently to its environment in real time.

Figure 6. Visualization of the autonomous vehicle in RViz with the localization enhancements

4. CONCLUSIONS

This project presents a complete and modular solution for integrating GPS and IMU data in real time to enhance localization in autonomous vehicles using the ROS2 framework. The developed system, which processes pre-recorded or real-time sensor data streams, delivers significant improvements in spatial accuracy, data fusion reliability, and platform compatibility. This project delivers a robust ROS2-based solution for real-time GPS and IMU data processing to enhance localization in autonomous vehicles. By aligning sensor timestamps, converting geodetic coordinates to a Cartesian system, and applying precise calibration, the system achieves high-accuracy localization. The improved sensor fusion capability ensures that GPS and IMU data are consistently aligned in space and time, contributing to more reliable vehicle positioning. The package is fully compatible with both ROS2 and Autoware, allowing seamless integration into existing autonomous stacks. Simulation and real-world dataset testing confirmed the system's effectiveness, showing clear improvements in localization accuracy and pointcloud alignment. This work establishes a strong foundation for future live sensor integration and advanced fusion approaches.

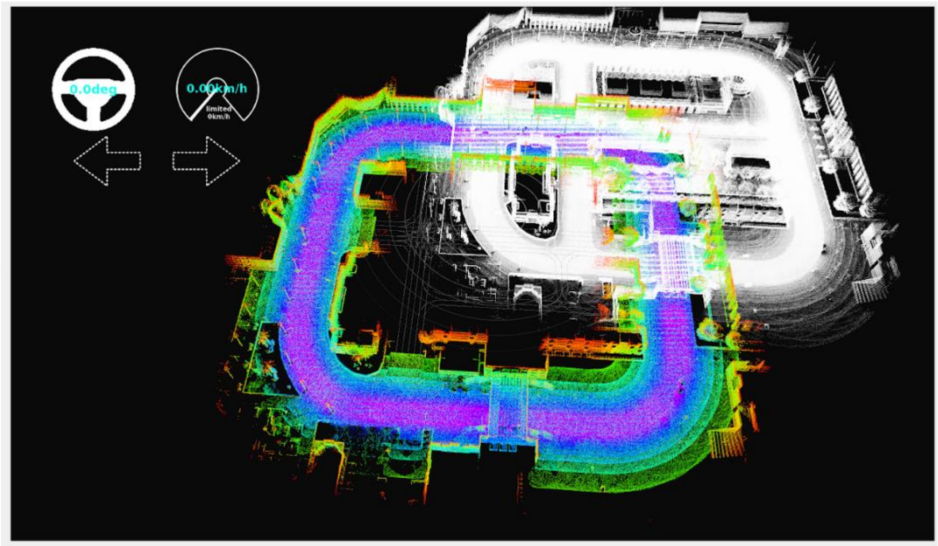

Figure 7. Localization drift & point cloud alignment before the implementation

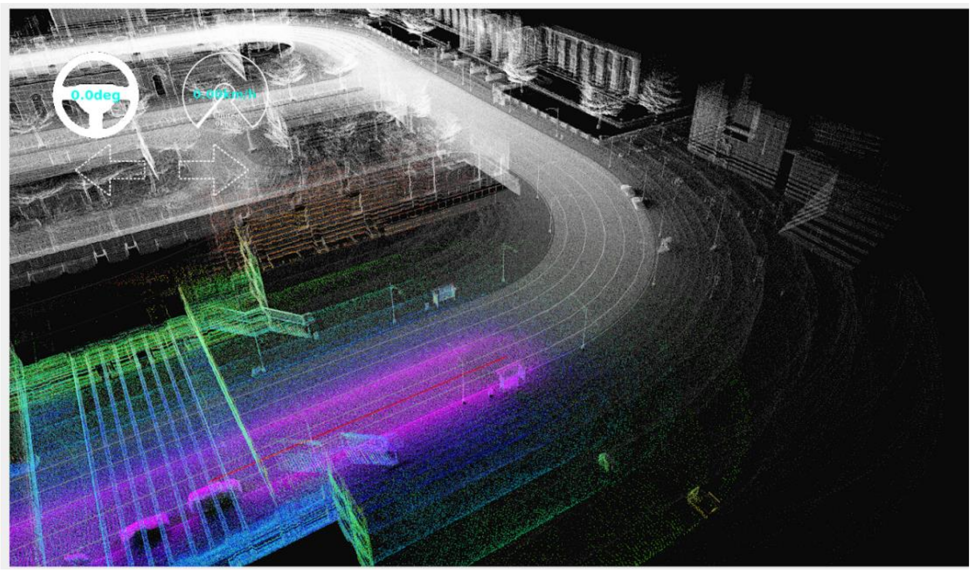

Figure 8. Localization drift & point cloud alignment after the implementation

5. REFERENCES

[1] Autoware Foundation. (2024). *Autoware documentation*. Retrieved from https://www.autoware.org

[2] Fox, D., Burgard, W., & Thrun, S. (1999). Markov localization for mobile robots in dynamic environments. *Journal of Artificial Intelligence Research*, *11*, 391–427. https://doi.org/10.1613/jair.616

[3] GeographicLib. (2025). *GeographicLib: A library for geographic projections and geodesy*. Retrieved from https://geographiclib.sourceforge.io

[4] Hofmann-Wellenhof, B., Lichtenegger, H., & Collins, J. (2001). *GPS: Theory and practice* (5th ed.). Springer-Verlag.

[5] Ko, J., & Fox, D. (2008). GP-BayesFilters: Bayesian filtering using Gaussian process prediction and observation models. *Autonomous Robots, 27*(1), 75–90. https://doi.org/10.1007/s10514-009-9119-5

[6] Quigley, M., Gerkey, B., Conley, K., Faust, J., Foote, T., Leibs, J., Wheeler, R., & Ng, A. Y. (2009). ROS: An open-source Robot Operating System. In *Proceedings of the ICRA Workshop on Open Source Software* (Vol. 3, No. 3.2).

[7] ROS2 Documentation. (2025). *Robot Operating System 2 (ROS2) documentation*. Retrieved from https://docs.ros.org/en/humble

[8] MathWorks. (n.d.). *Vectors in 3-D coordinate systems*. MathWorks. Retrieved April 20, 2025, from https://www.mathworks.com/help/map/vectors-in-3-d-coordinate-systems.html