

Low-Cost AIS Vessel Tracking System for Real-Time Web Visualization Using Software Defined Radio and Leaflet.js

Inka Trisna Dewi¹, Tri Budi Santoso², Rini Satiti³, Fabria Alieftya Yusvianti⁴,
Nafilah Aqilatuz Zuhroh⁵

^{1,2,3,4,5}Department of Electrical Engineering, Politeknik Elektronika Negeri Surabaya, Jl. Raya ITS, Surabaya, 60111

ARTICLE INFO

Article historys:

Received : 25/06/2026

Revised : 28/07/2026

Accepted : 03/08/2026

Keywords:

Automatic Identification System;
Leaflet.js; RTL-SDR; Vessel
Tracking; Web-Based Monitoring

ABSTRACT

A low-cost edge-based Automatic Identification System (AIS) monitoring platform was developed for real-time vessel surveillance, integrating AIS signal acquisition, decoding, data management, and geospatial visualization within a unified web-based architecture. The proposed platform was implemented using commercially available hardware components with a total hardware cost of approximately USD 189, providing a cost-effective alternative for real-time AIS monitoring. The system employs RTL-SDR for signal reception and a lightweight C-based processing pipeline incorporating GMSK demodulation, NRZI decoding, and CRC16 validation. Experimental deployment at Tanjung Perak Port under dense maritime traffic conditions demonstrated continuous end-to-end operation, achieving an average throughput of 3.71 messages per second and an end-to-end latency of 0–2 seconds. Decoded AIS records were stored in MariaDB and visualized through a Leaflet.js-based interface supporting real-time vessel tracking, historical trajectory reconstruction, and traffic analytics. Existing AIS studies commonly address signal reception, decoding, and visualization as separate processing stages. In contrast, the proposed platform consolidates the complete processing chain from radio-frequency AIS acquisition to web-based maritime information delivery within a single edge-computing environment. The results demonstrate the feasibility of employing resource-constrained edge devices for continuous AIS processing and provide an integrated framework for real-time maritime monitoring without reliance on high-performance computing infrastructure.



This work is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/)

Corresponding Author:

Inka Trisna Dewi

Department of Electrical Engineering, Politeknik Elektronika Negeri Surabaya, 60111, Surabaya

Email: inkatrisnad@pens.ac.id

1. INTRODUCTION

The increasing density and operational complexity of maritime traffic have made reliable real-time vessel monitoring an essential component of maritime safety and surveillance systems [1, 2]. This requirement is particularly critical in Indonesia, an archipelagic nation with extensive territorial waters and strategic positioning along major international shipping routes, resulting in continuous high-volume vessel traffic [3]. To ensure safety of life at sea, navigational efficiency, and environmental protection, the International Maritime Organization (IMO) mandates the use of the Automatic Identification System

(AIS) under Chapter V of the SOLAS Convention [4, 5]. AIS is required for vessels above specific tonnage thresholds and operates over VHF maritime channels at 161.975 MHz and 162.025 MHz [6], continuously broadcasting vessel identity, position, speed, heading, and navigational status for collision avoidance and maritime domain awareness [7, 9].

Despite its established role, conventional AIS monitoring infrastructures remain limited by deployment cost and computational requirements, which restrict the scalability of distributed surveillance systems in resource-constrained environments. Recent adoption of Software Defined Radio (SDR), particularly RTL-SDR, has reduced acquisition cost while enabling reception across a wide frequency range including AIS channels [10,11]. However, existing studies typically address AIS subsystems in isolation: signal reception and tracking improvements using RTL-SDR-based setups [12], GIS-based visualization and traffic analysis from preprocessed AIS data [13], and standalone demodulation and decoding pipelines implemented using GNU Radio frameworks [14]. These approaches do not provide a continuous transformation pathway from raw AIS RF signals to operational maritime information, particularly when deployed on resource-constrained edge platforms.

The primary contribution of this work is a complete AIS edge-computing architecture that unifies RF signal acquisition, GMSK demodulation, NRZI decoding, CRC16 validation, database structuring, trajectory reconstruction, maritime metric computation, and web-based geospatial visualization into a single continuous processing pipeline on low-cost hardware. The system is implemented using RTL-SDR for AIS capture, Raspberry Pi for edge processing, MariaDB for data storage, and Leaflet.js for spatial visualization. The processing chain enables continuous transformation of raw AIS signals into decoded message streams through sequential processing stages. Experimental results under dense maritime traffic conditions demonstrate sustained message throughput and bounded end-to-end latency from RF reception to database insertion and map rendering.

2. RESEARCH METHOD

The proposed AIS monitoring platform was implemented to perform signal acquisition, message decoding, data management, and web-based vessel visualization within a unified processing framework. The overall system design and operational workflow are illustrated in Figure 1 and Figure 2.

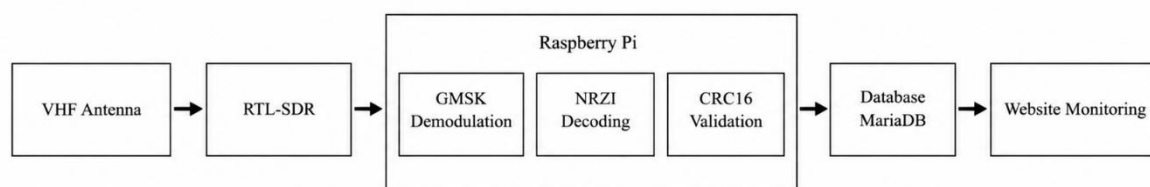


Figure 1. AIS monitoring system block diagram

Figure 1 outlines the principal functional components of the proposed platform and their interactions within the end-to-end AIS processing chain. AIS signals received through the VHF antenna are captured by the RTL-SDR receiver and forwarded to the Raspberry Pi, where message decoding and validation are performed before the resulting vessel information is stored and delivered to the web-based monitoring interface.

2.1. System Design

The system was configured to receive AIS transmissions over the maritime VHF channels at 161.975 MHz and 162.025 MHz. As shown in Figure 1, the architecture comprises a VHF antenna, an RTL-SDR receiver, a Raspberry Pi processing unit, a MariaDB database, and a Leaflet.js-based visualization interface.

AIS signals captured by the VHF antenna are digitized by the RTL-SDR and subsequently processed by the Raspberry Pi for message decoding, data management, and visualization. The hardware and software specifications employed in the implementation are summarized in Table 1.

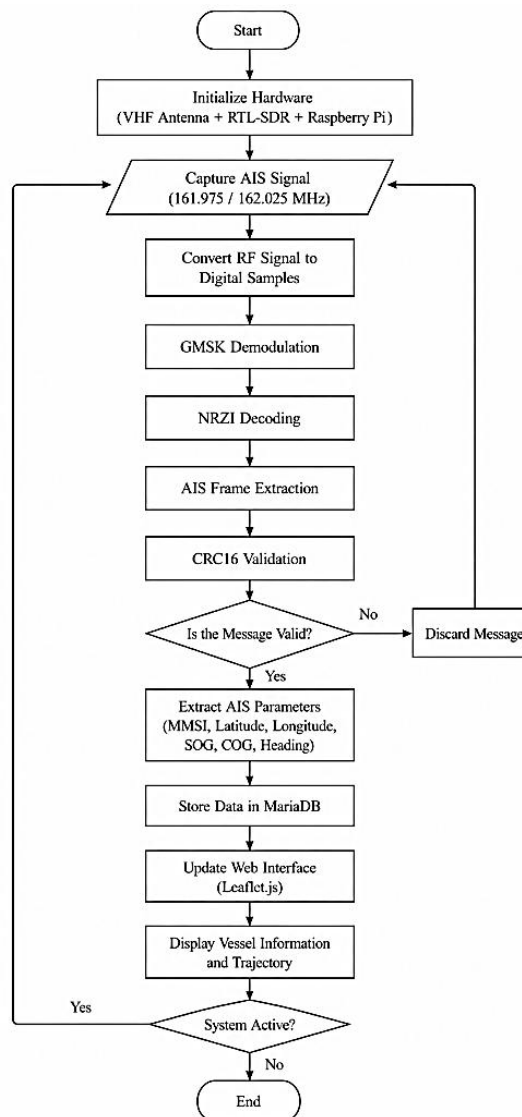


Figure 2. System workflow

2.2. Procedure and AIS Data Acquisition

As illustrated in Figure 2, the system workflow begins with AIS signal reception through the VHF antenna and RTL-SDR, followed by the conversion of radio signals into digital data for processing on Raspberry Pi using an AIS decoding algorithm developed in the C programming language.

The decoding process includes GMSK demodulation, NRZI decoding, and data verification using CRC16 to ensure the validity of the received information. Subsequently, the validated AIS data are extracted to obtain vessel parameters and stored in the database.

Table 1. Hardware and software configuration

Component	Specification
Processing Unit	Raspberry Pi 3 Model B+
Operating System	Raspberry Pi OS (32-bit)
SDR Receiver	RTL-SDR V4
AIS Frequencies	161.975 MHz, 162.025 MHz
Sampling Rate	300 kHz
Receiver Gain	80 dB
Database	MariaDB
Visualization Library	Leaflet.js

To substantiate the low-cost positioning of the proposed platform, the itemized hardware cost of the implemented system is presented in Table 2.

Table 2. Bill of materials and total hardware cost

Component	Unit Cost (USD)
RTL-SDR V4	50.28
Raspberry Pi 3 Model B+	55.08
VHF Antenna	67.04
MicroSD Card 32 GB	8.38
USB Cable	1.40
Coaxial Cable RG58	6.70
Total	188.88

The total hardware cost of the proposed system is approximately USD 189, substantially lower than commercial AIS receiver and base station systems, which typically range from USD 500 to over USD 10,000 depending on channel capacity and connectivity features [15]. This cost differential substantiates the low-cost positioning of the proposed platform relative to existing commercial alternatives.

2.3. Database Integration and Web-Based Visualization

Decoded AIS records are automatically stored in MariaDB, which serves as a centralized repository for both real-time and historical vessel information. The database is integrated with the web application to enable continuous access to vessel data and support subsequent analytical functions.

Geospatial visualization is implemented using Leaflet.js through an interactive map interface. Leaflet.js was selected for its lightweight architecture and efficient handling of dynamic geospatial data [16]. Vessel locations are rendered as dynamically updated markers, each containing key navigational attributes, including MMSI, geographical coordinates, speed, heading, and navigational status. This integration enables continuous spatial representation of vessel activity while facilitating intuitive observation of maritime traffic conditions through a web-based environment.

2.4. Experimental Setup and Evaluation Metrics

Field observations were conducted in the vicinity of Tanjung Perak Port, Surabaya, a maritime corridor characterized by intensive vessel activity and persistent AIS transmissions. AIS data were collected continuously over a 60-minute observation period. During the experiment, AIS signals received by the RTL-SDR receiver were processed by the Raspberry Pi, stored in the MariaDB database, and subsequently delivered to the web-based visualization platform.

System performance was evaluated using three operational metrics, message throughput, end-to-end latency, and processing stability. Message throughput refers to the number of valid AIS messages successfully decoded and processed within each observation interval. The average message throughput was calculated using equation (1):

$$\text{Throughput} = \frac{N_{msg}}{T} \quad (1)$$

where:

N_{msg} denotes the total number of valid AIS messages processed during the observation period (messages), and T represents the corresponding observation time (s).

End-to-end latency represents the elapsed time between AIS signal reception and the availability of vessel information on the web interface. Processing stability was assessed through the continuity of AIS decoding, database synchronization, and visualization updates throughout the observation period.

Collectively, these metrics were employed to quantify the performance of the complete processing chain and to assess the capability of the proposed architecture to sustain continuous AIS acquisition, decoding, data management, and visualization within a unified operational environment.

3. RESULTS AND DISCUSSION

The implemented AIS monitoring platform establishes an integrated processing environment that combines AIS signal acquisition, message decoding, data management, and vessel visualization. The hardware implementation of the AIS receiver system is presented in Figure 3.



Figure 3. Hardware implementation of the AIS receiver system

The decoded output displayed on the Raspberry Pi terminal, as shown in Figure 4, confirms consistent extraction of vessel-related information. The results indicate that signal reception stability is influenced by the combined configuration of the RTL-SDR and VHF antenna optimized for AIS frequencies. In addition, the use of a C-based decoding algorithm reduces computational overhead, allowing the system to remain responsive on resource-constrained hardware. The CRC16 mechanism also plays a crucial role in improving data reliability by filtering invalid AIS packets.

3.1. Web-Based Vessel Data Visualization

Decoded AIS records were successfully stored in the MariaDB database in real time, as illustrated in Figure 5. The stored information includes MMSI, geographical coordinates, vessel speed, heading, and navigational status, providing the data foundation required for real-time visualization and historical trajectory analysis.

A comparison between the timestamps observed in Figure 4 and Figure 5 indicates that decoded AIS records became available in the database within approximately 0–2 seconds after successful decoding, demonstrating timely propagation of information across the processing chain.

```
[REKAP] 1 menit ke-1 = 213 data diterima
Ter kirim: MMSI 525000011 | Waktu 2026-05-25 20:54:03
Ter kirim: MMSI 525023138 | Waktu 2026-05-25 20:54:04
Ter kirim: MMSI 525114133 | Waktu 2026-05-25 20:54:04
Ter kirim: MMSI 525201479 | Waktu 2026-05-25 20:54:04
Ter kirim: MMSI 525016550 | Waktu 2026-05-25 20:54:05
Ter kirim: MMSI 525801170 | Waktu 2026-05-25 20:54:05
Ter kirim: MMSI 525901286 | Waktu 2026-05-25 20:54:06
Ter kirim: MMSI 525007390 | Waktu 2026-05-25 20:54:06
Ter kirim: MMSI 525020261 | Waktu 2026-05-25 20:54:07
```

Figure 4. AIS decoded output on Raspberry Pi terminal

id	mmsi	latitude	longitude	speed	course	waktu
214	525000011	-7.217932	112.721722	0	111	2026-05-25 20:54:03
215	525023138	-7.187277	112.725833	0	0	2026-05-25 20:54:04
216	525114133	-7.194522	112.74243	0.1	360	2026-05-25 20:54:04
217	525201479	-7.188257	112.724585	0	34.3	2026-05-25 20:54:04
218	525016550	-7.204395	112.724708	0.1	273	2026-05-25 20:54:05
219	525801170	-7.20305	112.722633	0.1	148	2026-05-25 20:54:05
220	525901286	-7.17372	112.685027	0	311.4	2026-05-25 20:54:06
221	525007390	-7.20584	112.725213	0	266.1	2026-05-25 20:54:06
222	525020261	-7.18046	112.676385	0	0	2026-05-25 20:54:07

Figure 5. AIS data storage in MariaDB database

The stored AIS records were subsequently visualized through a Leaflet.js-based web interface, as shown in Figure 6. Vessel locations are represented as dynamic markers positioned according to their

geographical coordinates. Marker interaction provides access to vessel-specific information, including MMSI, speed, heading, country of origin, and navigational status, as illustrated in Figure 7.

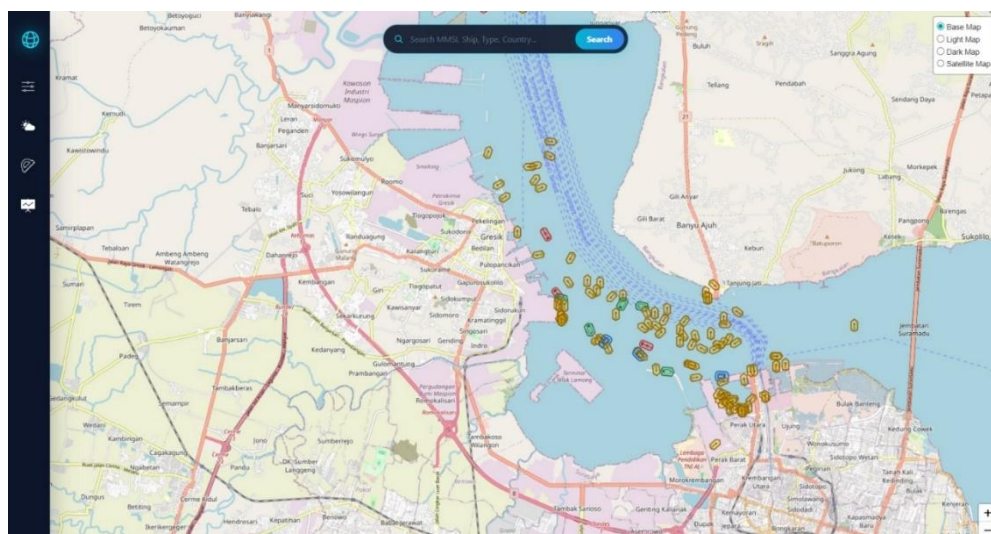


Figure 6. Web-based AIS vessel monitoring interface

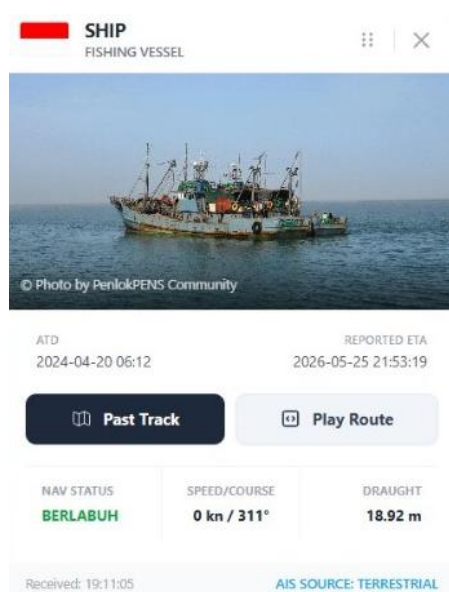


Figure 7. Vessel detailed information view

From an interpretative perspective, spatial visualization significantly enhances data interpretability compared to text-based representations, as it enables users to directly observe vessel distribution and movement patterns within a geographic context without additional interpretation steps. The system also integrates a vessel trajectory tracking feature that enables historical movement analysis. This feature provides temporal analysis of vessel trajectories over a specified time interval, as shown in Figure 8. Such capability can be utilized for maritime surveillance, traffic density analysis, and maritime incident investigation.

To enhance situational awareness, the system also provides maritime traffic statistics through an interactive dashboard, as shown in Figure 9. The dashboard aggregates AIS records stored in the database and presents vessel distributions according to vessel type, navigational status, and country of origin. During the observation period, a total of 114 vessels were identified, consisting of 98 anchored vessels and 16 sailing vessels. These results demonstrate the capability of the platform to consolidate AIS data into operationally interpretable maritime information within a single interface.



Figure 8. Vessel trajectory tracking feature

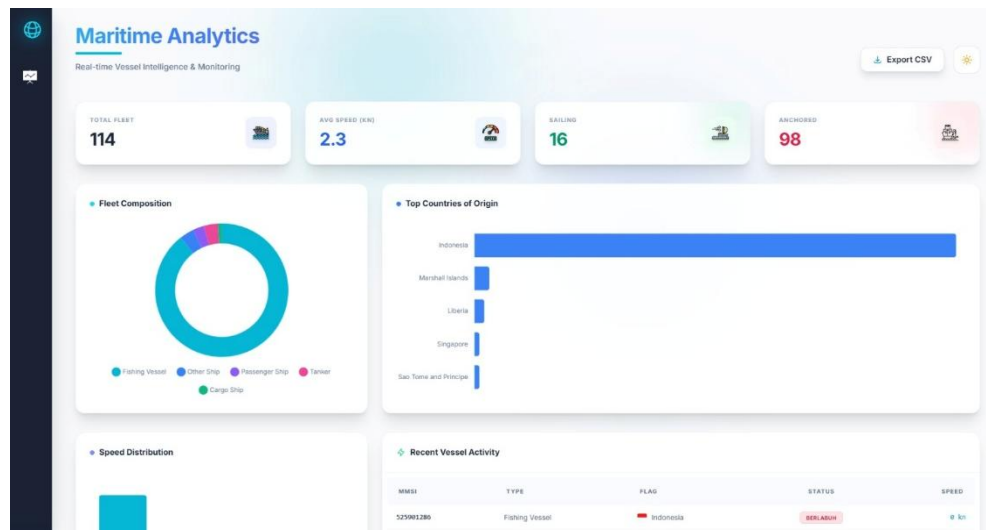


Figure 9. System dashboard and statistical visualization

3.2. System Performance Evaluation

System performance was evaluated over a 60-minute observation period using message throughput, end-to-end latency, and processing stability as evaluation metrics. Message throughput statistics are presented in Table 3.

As shown in Table 3, message counts ranged from 996 to 1,295 across the twelve 5-minute intervals, with an average of 1,113 valid messages processed per interval and a cumulative total of 13,357 messages. Based on (1), the corresponding average throughput was approximately 3.71 messages s^{-1} . To quantify the dispersion of message counts across observation intervals, the standard deviation (SD) and coefficient of variation (CV) were calculated using equations (2) and (3):

$$SD = \sqrt{\frac{\sum (x_i - \bar{x})^2}{n - 1}} \quad (2)$$

$$CV = \frac{SD}{\bar{x}} \times 100\% \quad (3)$$

where:

x_i denotes the message count in interval i , $\bar{x}$ denotes the mean message count (messages), and n is the total number of observation intervals ($n = 12$).

The calculated SD and CV were 96.12 messages and 8.64%, respectively, indicating limited temporal variability and demonstrating that the proposed AIS receiver maintained consistent message acquisition and stable throughput throughout the 60-minute observation period despite natural fluctuations in maritime traffic density.

Table 3. AIS data received at tanjung perak

Interval Time	Number of Data Received
20:53 – 20:58	1,295
20:59 – 21:03	1,045
21:04 – 21:08	1,029
21:09 – 21:13	1,055
21:14 – 21:18	1,036
21:19 – 21:23	1,040
21:24 – 21:28	1,117
21:29 – 21:33	1,159
21:34 – 21:38	1,238
21:39 – 21:43	1,215
21:44 – 21:48	1,132
21:49 – 21:53	996
Total	13,357
Average	1,113

End-to-end latency was estimated from the timestamp differences between decoded AIS messages generated at the Raspberry Pi processing stage and the corresponding database insertion records. A comparison of the timestamps shown in Figures 4 and 5, together with uninterrupted updates on the web interface, confirms continuous temporal propagation throughout the acquisition, decoding, storage, and visualization pipeline. The observed end-to-end latency ranged from 0 to 2 seconds, demonstrating the capability of the proposed edge-based architecture to support near real-time maritime monitoring.

3.3. Discussion

The results indicate that the integration of RTL-SDR-based AIS reception with web-based visualization enables a cohesive maritime monitoring workflow within a single edge-computing environment. Beyond AIS data reception, the implemented platform combines signal acquisition, message decoding, data management, trajectory reconstruction, and geospatial visualization to support near real-time observation of vessel movements.

Table 4 summarizes the comparison between the proposed system and previous AIS monitoring studies. As shown, the main contribution of this work is not merely the implementation of AIS reception, but the integration of signal acquisition, decoding, database management, and web-based visualization into a unified edge-computing framework. In addition, the proposed platform extends conventional AIS monitoring by providing historical trajectory analysis and statistical dashboard visualization within a single system.

The integration of Leaflet.js further enhances the practicality of the proposed platform by providing a lightweight and interactive visualization interface suitable for resource-constrained edge devices such as the Raspberry Pi. Beyond real-time vessel positioning, the platform supports historical trajectory analysis and statistical dashboard visualization, extending the functionality of conventional AIS monitoring systems. These capabilities enable the proposed platform to provide not only real-time vessel tracking but also comprehensive maritime situational awareness within a unified web-based environment.

Table 4. Comparison with previous studies

Previous Studies	Object	Method	Throughput (msg/s)	Latency	Hardware Cost
[9]	AIS signal reception, decoding, and web-based visualization	VHF antenna + AIS receiver + LabVIEW + Leaflet.js	NR	NR	NR
[12]	AIS signal reception and monitoring	Yagi antenna + RTL-SDR + Low Noise Amplifier + SDR-Sharp + AISMon + OpenCPN	NR	NR	NR
[13]	Ship movement tracking using AIS data	VHF antenna + AIS receiver + Quantum GIS	NR	NR	NR
Proposed System	AIS signal reception and monitoring	VHF antenna + RTL-SDR + Raspberry Pi + C-based AIS decoding + MariaDB + Leaflet.js	3.71	0–2 s	USD 189

4. CONCLUSION

This study presented an AIS-based vessel monitoring platform that integrates signal acquisition, message decoding, data management, and web-based geospatial visualization within a unified edge-computing architecture. Field observations conducted under dense maritime traffic conditions showed that the system processed a total of 13,357 valid AIS messages during a 60-minute observation period, corresponding to an average throughput of 3.71 messages per second. The measured end-to-end latency of approximately 0–2 seconds indicates that vessel information can be delivered to the monitoring interface with minimal delay. The integration of RTL-SDR-based reception, Raspberry Pi-based processing, MariaDB data management, and Leaflet.js visualization establishes a continuous processing chain from AIS radio transmissions to operational vessel information. In addition to real-time vessel tracking, the platform supports historical trajectory reconstruction and maritime traffic analytics within a single operational environment. With a total hardware cost of approximately USD 189, substantially lower than commercial AIS receiver and base station systems, the proposed platform demonstrates that reliable real-time maritime monitoring can be achieved using low-cost edge-computing hardware, highlighting its feasibility for large-scale deployment in resource-constrained environments.

A limitation of the proposed system is that situational awareness is constrained to vessels transmitting valid AIS messages, as the monitoring process relies exclusively on AIS-derived information. Consequently, non-AIS targets, stationary maritime objects, coastal infrastructure, and vessels with inactive AIS transponders cannot be detected or visualized by the platform. Future work may therefore explore the integration of complementary sensing technologies, such as marine radar, camera-based monitoring, or multi-source data fusion, to provide a more comprehensive representation of the maritime environment.

Acknowledgments

This research was supported by the Internal Research Grant of Politeknik Elektronika Negeri Surabaya (PENS) 2026 under contract number No. 012/SP2H/PENS/PPK/2026.

REFERENCES

- [1] N. R. Kadafuk and U. Mahendradatta, "Implementation of Vessel Traffic Service (VTS) and Automatic Identification System (AIS) for Shipping Safety: a Case Study of Type a Class II Benoa Navigation District," vol. 3, 2026.
- [2] V. J. Senjaya, "Pemanfaatan automatic identification system berbasis open source untuk pemantauan kapal dalam rangka menjaga keamanan perairan indonesia".

- [3] F. R and A. Burhanuddin, "Implementasi kebijakan sekuritisasi maritim presiden jokowi dalam menghadapi aktivitas ilegal di perairan indonesia" *Indonesian J.Maritime*, vol. 4, no. 2, pp. 85–98, Mar. 2024, doi: 10.17509/ijom.v4i2.64135.
- [4] "International Convention for the Safety of Life at Sea (SOLAS), 1974." Accessed: May 31, 2026. [Online]. Available: [https://www.imo.org/en/about/conventions/pages/international-convention-for-the-safety-of-life-at-sea-\(solas\),-1974.aspx](https://www.imo.org/en/about/conventions/pages/international-convention-for-the-safety-of-life-at-sea-(solas),-1974.aspx)
- [5] M. B. Zaman, "Study on Safety of Navigation using Automatic Identification System for Marine Traffic Area Case Study: Malacca Straits," *IJMEIR*, vol. 1, no. 1, Dec. 2016, doi: 10.12962/j25481479.v1i1.1462.
- [6] Y. W. Matrutty, Y. Saragih, P. Waluyo, and S. Suroyo, "Analisis keberhasilan automatic identification system (AIS) pada kapal tug boat leo power 2206," *Power Elektronik : Jurnal Orang Elektro*, vol. 11, no. 2, p. 266, Sep. 2022, doi: 10.30591/polektro.v12i1.3649.
- [7] T. D. Rahmawati, A. A. I. S. Wahyuni, and T. Septia, "Analisis peran Sistem Identifikasi Otomatis (Ice) terhadap keselamatan navigasi pada posisi umum di kapal MV. Guhi Mas," *ejmi*, Jul. 2025, doi: 10.62391/ejmi.v7i1.122.
- [8] J. Koto and N. A. Amran, "Development of Automatic Identification System in Strait of Batam-Singapore," 2016.
- [9] H. Saputra and J. Koto, "Development of Automatic Identification System (AIS) for Vessels Traffic Monitoring in the Strait of Singapore and Batam Waterways," 2018.
- [10] V. S. Tomar and V. Bhatia, "Low Cost and Power Software Defined Radio Using Raspberry Pi for Disaster Effected Regions," *Procedia Computer Science*, vol. 58, pp. 401–407, 2015, doi: 10.1016/j.procs.2015.08.047.
- [11] D. Ball, N. Naik, and P. Jenkins, "Lightweight and Cost-Effective Spectrum Analyser Based on Software Defined Radio and Raspberry Pi," in *2017 European Modelling Symposium (EMS)*, Manchester: IEEE, Nov. 2017, pp. 260–266. doi: 10.1109/EMS.2017.51.
- [12] R. P. Sari, L. Lindawati, and S. Soim, "Monitoring Kapal Menggunakan Automatic Identification System(AIS) Dengan RTL-SDR dan Low Noise Amplifier (LNA)," *protk*, vol. 9, no. 2, p. 119, Sep. 2022, doi: 10.33387/protk.v9i2.4691.
- [13] H. Saputra, "Penggunaan Data Automatic Identification System (AIS) untuk Mengetahui Pergerakan Kapal (Studi Kasus pada Lalu Lintas Kapal di Selat Singapura dan Perairan Batam)".
- [14] B. Oumimoun, L. Nahiri, H. Idmouida, A. Addaim, Z. Guennoun, and K. Minaoui, "Software Defined AIS Receiver Implementation Based on RTL-SDR and GNU Radio," in *2022 IEEE Asia Pacific Conference on Wireless and Mobile (APWiMob)*, Bandung, Indonesia: IEEE, Dec. 2022, pp. 1–5. doi: 10.1109/APWiMob56856.2022.10014409.
- [15] M. Gucma, "Low Cost Ais System for Safe Navigation," *Journal of Konbin*, vol. 6, no. 3, pp. 235–246, Jan. 2022, doi: 10.2478/v10040-008-0070-2.
- [16] I. A. Marleni and A. Gunaryati, "Presensi Karyawan Berbasis Web dengan Fitur Lokasi Leaflet JS menggunakan Laravel," *jtik*, vol. 7, no. 3, pp. 479–485, Jul. 2023, doi: 10.35870/jtik.v7i3.947.