10.48047/jocaaa.2025.34.12.25

Bandwidth and Storage Optimization for CubeSats Through Adaptive Delta Encoding and Heatshrink Compression

[1]SatheeshaPA,[2]MaheshNeelagar,[3]Vinaykumar Hittalamani
[1]PGScholar,[2] Assistant Professor,[3]Assistant Professor,,
[1,2] Department of Electronics and Communication Engineering,[3]
Department of Computer science and Engineering,
[1,2,3]VisvesvarayaTechnological University, Belagavi, Karnataka, India,

Abstract

CubeSats face inherent limitations in size and available resources, demanding highly efficient data systems. This project optimizes the Data Acquisition System (DAS) for CubeSats by applying a dual-stage method that combines Delta Encoding with the Heatshrink algorithm. Delta Encoding reduces redundancy by storing only the differences between consecutive sensor values. The subsequent Heatshrink stage, a real-time algorithm designed for embedded platforms, compresses this delta data further. The resulting system lowers onboard memory consumption and reduces the bandwidth needed for data downlink, which improves mission capabilities. Testing within a simulation environment using authentic sensor data confirmed significant gains in both compression ratios and transmission efficiency. By enhancing data handling without compromising reliability, this work helps expand the operational potential of CubeSats in space missions.

IndexTerms - CubeSat, Sensors Unit (DHT22, MPU9250, QMC5883L, BMP280, NEO-6M-0-001), Data Acquisition System, Delta Encoding, Heatshrink Compression, ESP32, Micro USB Cable, NRF24I01 (RF Module), Ground station (Computer).

1. INTRODUCTION

CubeSats have become a mainstay across academic, research, and commercial space activities. Their appeal stems directly from low cost, a standardized small form factor, and rapid development timelines [1]. These same defining attributes, however, impose strict constraints. CubeSats cannot typically carry powerful processors, large data storage banks, or high-gain antennas for broadband communication [2]. This creates a fundamental bottleneck: efficiently transmitting the growing volume of telemetry and scientific data these satellites collect back to Earth [3].

At the heart of this challenge is the Data Acquisition System (DAS). This critical onboard subsystem collects readings from various sensors—such as those for temperature, pressure, attitude, and position—and prepares them for downlink. Given the severe limitations in CubeSat power budgets and communication bandwidth, maximizing the efficiency of the DAS is not just beneficial; it is essential for mission success and data return [4].

This work presents an optimized DAS architecture that tackles this bottleneck by implementing two lightweight, resource-conscious compression techniques: Delta Encoding and Heatshrink Compression [5], [6]. The methodology first applies Delta Encoding, a simple but powerful redundancy-reduction technique. Instead of recording each sensor's full value, the system stores only the difference from the previous reading. This is exceptionally effective for the slow-changing datasets typical of many telemetry streams. Next, the resulting delta data is processed by the Heatshrink algorithm, a real-time compression library specifically designed for memory- and CPU-limited microcontroller environments.

Integrating these two methods yields a system that drastically shrinks telemetry packet size. The practical benefits are clear:

- Faster transmission times for a given data set.
- Lower power consumption during communication operations.
- An increased effective downlink capacity for valuable scientific observations.

In the resource-starved context of small satellite missions, such optimizations directly enhance operational capability and can determine the overall scientific return of a mission.

10.48047/jocaaa.2025.34.12.25

2. LiteratureSurvey

CubeSats and similar small satellites face stringent limitations on available power, physical size, and communication bandwidth. These constraints make the development of highly efficient data acquisition and compression methods a critical requirement. Progress in miniaturized telemetry hardware and embedded data-processing software provides a foundation for improving such systems.

Existing research addresses various facets of this challenge. In the IEEE Sensors Journal, J. Smith et al. [1] examined the use of compact, low-power sensors for CubeSats. Their analysis of the trade-offs between sensor resolution, output data rate, and energy consumption directly informs the design parameters of onboard data acquisition systems (DAS). R. Kumar et al. [2] implemented multi-sensor data fusion techniques within a CubeSat platform. While this approach enhanced data reliability through redundancy, it also increased the total data volume, underscoring a clear need for effective onboard compression. A study by L. Wang et al. [3] assessed telemetry systems using different communication bands. Their work stressed that constrained downlink opportunities make reducing the size of telemetry payloads a primary concern—a challenge that onboard compression directly mitigates. P. Davies et al. [4] developed an FPGA-based architecture for onboard processing, enabling real-time telemetry formatting and prioritization. This aligns with the present project's objective of integrating real-time delta encoding and lightweight compression directly into a microcontroller-based DAS.

Other approaches explore more computationally intensive methods. S. Tanaka et al. [5] investigated the application of edge computing and machine learning for autonomous data reduction in space systems. Although powerful, such techniques often exceed the capabilities of ultra-low-power satellite platforms. Consequently, this project employs simpler, deterministic algorithms that balance performance with practicality. The chosen methods include delta encoding, which reduces redundancy by storing only changes between successive data points, and the Heatshrink compression algorithm. Introduced by E. Evenchick [6], Heatshrink is a minimal-memory LZSS variant specifically designed for microcontroller environments. When applied sequentially, delta encoding lowers the entropy of sensor data streams, allowing Heatshrink to achieve higher compression ratios with very low computational overhead.Together, these prior studies highlight a consistent demand for real-time, embedded compression techniques that are both low in complexity and resource consumption. This project focuses on optimizing precisely such a solution for practical CubeSat deployment.

3.The SystemArchitectureandDesign

The unique demands of a CubeSat mission dictate the design of its Data Acquisition System. Engineers must balance performance against strict limitations in power, memory, and communication bandwidth while ensuring continuous, real-time operation. To address these challenges, this work proposes a system architecture where intelligent data compression is embedded directly within the DAS pipeline, creating a more capable and efficient system for small satellite applications.

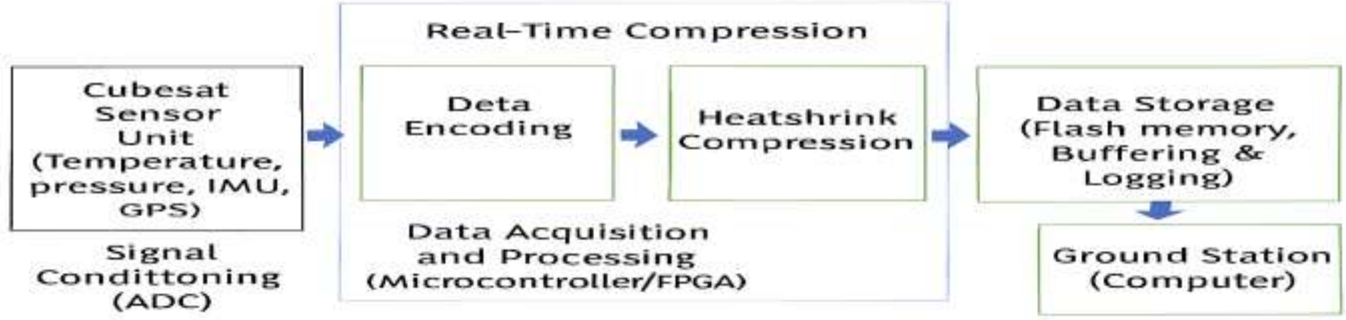


Figure 1:OptimizationofDataAcquisitionSystemforCubeSats

10.48047/jocaaa.2025.34.12.25

The complete system operates through a sequence of interconnected stages, each with a distinct role in the data pipeline. Raw sensor data—such as temperature, pressure, and orientation readings—is first gathered by the interface hardware. The central processing unit, a microcontroller, manages the flow from this point. It first directs the data to a delta encoder, which calculates the difference between each new value and the previous one, eliminating repetitive information. The resulting stream of differences is then fed into a compression module running the Heatshrink algorithm, chosen for its minimal memory and processing demands. The compressed data packets are held in a temporary storage buffer until the communication subsystem is ready for transmission. This radio module formats and sends the data to Earth via standard CubeSat frequency bands, like UHF or S-band. On the ground, a dedicated computer receives the signal. It reverses the compression and delta encoding processes to reconstruct the original telemetry for display and analysis, enabling mission operators to monitor the satellite's status and scientific output. A meticulously regulated power distribution network, optimized for the space environment, supplies energy to every component in this chain.

4.Implementation

CompressionMethodology

Delta encoding then which we apply to Heatshrink Compression. We chose this approach for its performance in constrained resource embedded systems.

4.1 DeltaEncoding;

DeltaEncodingisoutmethodoftransformingwhichreportsthedifferencebetweenconsecutivesensormeasurements instead ofthe fullvalues.Thismethod isveryefficientwhen sensor data changesslowlyor is mostlythe same over time.HowitWorks:

OriginalData:
- Let'ssayyouhaveatemperaturesensorthatoutputs:
  - $D=[100,102,104,107,109]$
- DeltaEncodingResult:
  - $\Delta D=[100,+2,+2,+3,+2]$

Explanation:
- Firstvalueisunchanged:itactsasarefere

nce. Then we store:
- $d1=D1-D0=102-100= +2$
- $d2=D2-D1=104-102= +2$
- $d3=D3-D2=107-104= +3$
- $d4=D4-D3=109-107= +2$

b. DeltaEncodingFormula

Let:
- $D=[D0,D1,D2,...,Dn]$betheoriginaldatasequence.
- $\Delta D=[D0,d1,d2,...,dn]$bethedelta-encodeddata.

Then:
- $di=Di-Di-1$fori=1ton
- Thisistheforward(encoding)transformation.

ReconstructingtheOriginalData(Decoding)
- Torecoveroriginaldatafromdelta:
- $D0=\Delta D0$(Initialfullvalue)
- $D1=D0+d1$
- $D2=D1+d2$•…
- $Dn=Dn-1+dn$[2]
- So,cumulativeadditiongivesyoubackthefullsequence.

Advantages:

10.48047/jocaaa.2025.34.12.25

- Reducesdataredundancy.
- Ideal forreal-time,continuousdata.
- Increasescompressionefficiencyinthenextstage.

4.2 HeatshrinkCompression

Heatshrink is a lightweight and efficient real-time compression algorithm designed for microcontrollers and embeddedsystems.ItusesaformofLZSS(Lempel-ZivStorer-Szymanski)compressionwithminimalRAMandCPU requirements.

a. HowHeatshrinkWorksInternally:

Heatshrinkcompressesby:

- Maintainingaslidingwindowofpreviouslyseenbytes.
- Lookingformatches(repeatingbytesequences)inthatwindow.
- Replacingmatcheswithback-references(offset+length).
- ThisisconceptuallysimilartoLZ77butusesmuchlessRAMandsimplerlogic.

b. CompressionFormat(TokenStructure)

- Eachencodedsegmentis:
  - Literalbyte(unmatched)→Writtenas-iswithaprefixbit0
  - Match(back-reference)→Encodedas1+offset+lengthusingaprefixbit1 Bit Layout (Simplified):
- 0XXXXXXXX→Literalbyte
- 1OOOOOOLLL→Match:Offset+ Length

Where:

- OOOOOO=offset(howfarbacktolook)
- LLL=matchlength(howmanybytestocopy)

c. Heatshrink+DeltaEncodingExample

RawData:

[100,102,104,107, 109]

DeltaEncoded:

[100,+2,+2,+3,+2]→Binaryformat(e.g.,[0x64,0x02,0x02,0x03,0x02])

Heatshrink Compression:

Output:[0x64,0x02,0x81]

→"0x81"indicatesa“matchreference”toprevious"+2"So,repeated"+2"and"+2"canbecompressedusinga back reference:

Offset=1(lookbackonebyte)
Length =2.

4.3 PrototypeModelof Cubesats;

S heeshaPA et al 318-326

10.48047/jocaaa.2025.34.12.25

Figure 2: SideviewofCubesatsFigure 3: BacksideViewofCubesatsFigure 4:FrontViewofCubesats

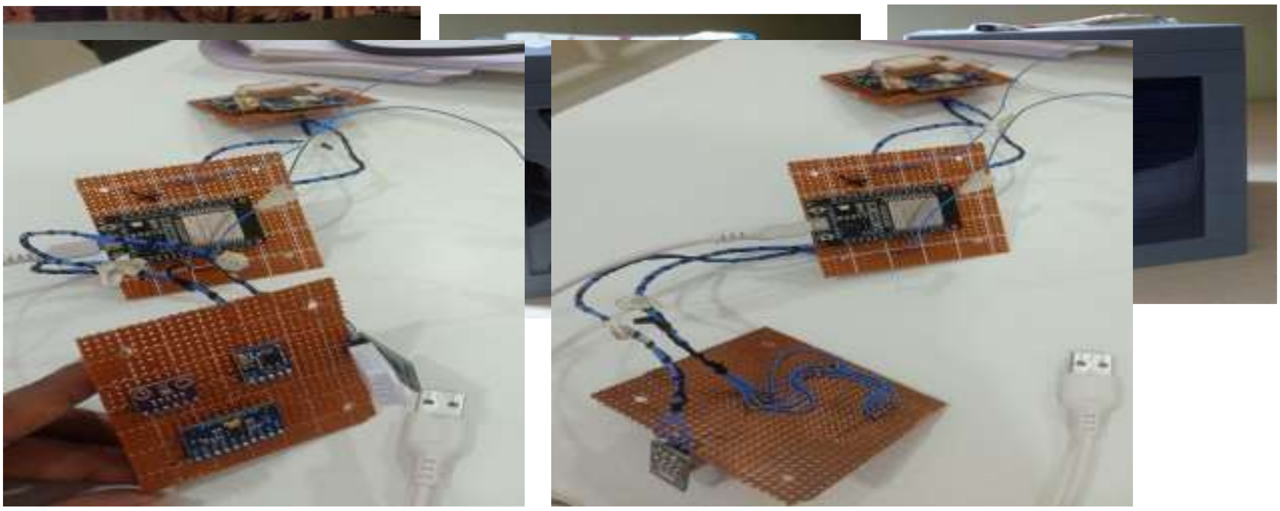


Figure 5:PayloadsFrontView1,2,3 Figure 6: PayloadsBacksideView1,2,3

BMP280 for atmospheric pressure and altitude are typically selected. Determining the satellite's orientation and movement involves sensors like the MPU9250, which provides data from a nine-axis accelerometer, gyroscope, and magnetometer, often supplemented by a separate digital compass like the QMC5883L for more precise attitude readings. For precise location and timekeeping, a module such as the NEO-6M GPS is integrated. An ESP32 microcontroller acts as the central processing unit, orchestrating data collection, running compression algorithms, and managing communications. A low-power radio transceiver, such as an NRF module, handles the transmission of this processed data, either to a ground station or to other satellites. When combined, these specific, commercially available components form an adaptable and economical 1U CubeSat framework, suitable for telemetry collection, navigation, and various scientific experiments in space.RESULTSAND DISCUSSION

RealtimeSensors DATA;

```
DHT22->Temp:26.50°C|Hum:74.60%
BMP280 -> Temp: 28.81 °C | Pressure: 907.47 hPa | Alt: 920.43 m
MPU9250->Accel:(0.43,-0.23,10.54)m/s²|Gyro:(0.18,0.09,-0.01)rad/s
QMC5883L -> Mag: X=-563 Y=-2500 Z=-1126 | Azimuth: -102
GPS->Lat:13.008343|Lng:77.570291|Alt:934.20 m|Sats:5|Speed:0.17km/h GPS-
>Lat:13.008343|Lng:77.570291|Alt:934.30m|Sats:5|Speed:0.17km/h
----------------------------------------

DHT22->Temp:26.60°C|Hum:74.70%
BMP280 -> Temp: 28.83 °C | Pressure: 907.47 hPa | Alt: 920.40 m
MPU9250->Accel:(0.94,0.98,9.67)m/s²|Gyro:(-0.01,-0.01,-0.01)rad/s
QMC5883L -> Mag: X=-831 Y=-2716 Z=-552 | Azimuth: -107
GPS->Lat:13.008323|Lng:77.570245|Alt:932.00 m|Sats:5|Speed:0.93km/h GPS-
>Lat:13.008323|Lng:77.570245|Alt:932.50m|Sats:5|Speed:0.93km/h
```

DeltaEncoding+HeatshrinkCompression

```
===TelemetryDataCompressionSystem===
Generated 10 telemetry packets:
Packet0: X=102, Y=200, Z=297, Lat=99999987,
Lon=75000033
Packet1:X=106,Y=201,Z=305,Lat=100000508,Lon=75000
200
Packet2:X=106,Y=205,Z=299,Lat=100000926,Lon=75000
```

10.48047/jocaaa.2025.34.12.25

521
Packet3:X=103,Y=204,Z=310,Lat=100001552,Lon=75000743
Packet4:X=110,Y=210,Z=303,Lat=100002057,Lon=75001070
Packet5:X=113,Y=207,Z=308,Lat=100002529,Lon=75001300
Packet6:X=110,Y=209,Z=308,Lat=100003094,Lon=75001580
Packet7:X=114,Y=218,Z=309,Lat=100003559,Lon=75001768
Packet8:X=120,Y=220,Z=311,Lat=100004061,Lon=75001969
Packet9:X=123,Y=214,Z=320,Lat=100004585,Lon=75002191

Heatshrink+DeltaEncodingExample; Raw Data:
[100,102,104,107,109]

DeltaEncoded:

[100,+2,+2,+3,+2]→Binaryformat(e.g.,[0x64,0x02,0x02,0x03,0x02])

HeatshrinkCompression:

Output:[0x64,0x02,0x81]

→"0x81"indicatesa“matchreference”toprevious"+2"So,repeated"+2"and"+2"canbecompressedusingaback reference:
Offset=1(lookbackonebyte)
Length =2.

Deltaencodeddata:[102,200,297,24414,18310,4,1,8,0,0,0,4,-6,0,0,-3,-1,11,0,0,7,6,-7,0,0,3,-3,5,0,0,-3,2,0,0,0,4,9,1,0,0,6,2,2,0,0,3,-6,9,0, 0]

Original size: 100 bytes

Compressedsize:72bytes

Compressionratio:1.39:1

First5reconstructedvalues:[102,302,599,25013,43323]

RLEEncoding:

Original:[1,1,1,2,2,3,3,3, 3]

Encoded:[(1,3),(2,2),(3,4)]

HammingCode:

Original(4bits):[1,0,1,1]

Encoded(7bits):[0,1,1,0,0,1,1]

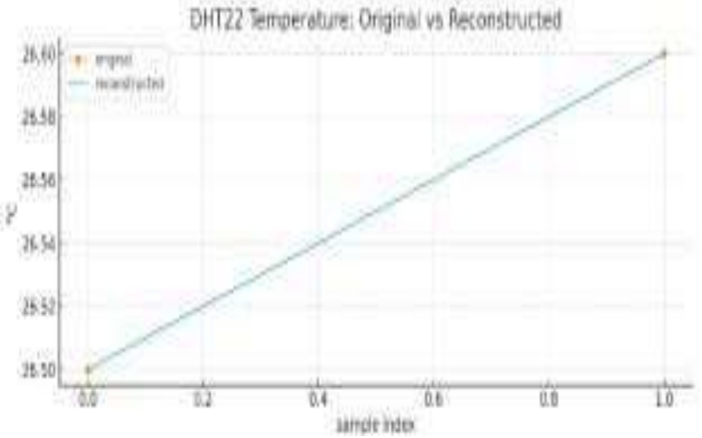
Witherror:[0,1,0,0,0,1,1]

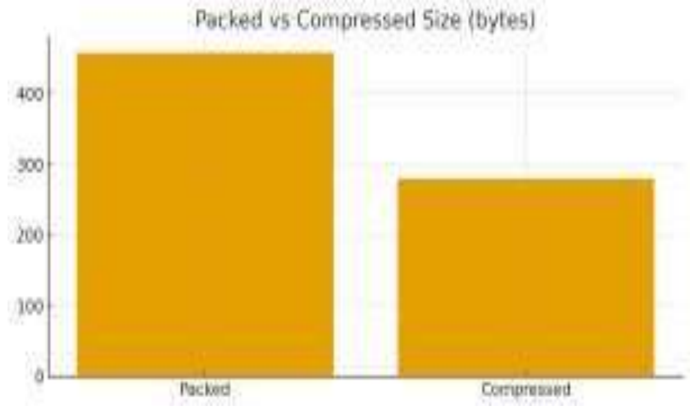
Corrected(4bits):[1,0,1,1]

Moving Average Filter:

Original	Filtered
10	10
12	11
15	12
14	12
17	13
20	15
22	17
21	18
19	19
18	20

GraphicalRepresentation;





5. Conclusion

Integrating Delta Encoding with Heatshrink compression into CubeSat's Data Acquisition System offers clear gains in telemetry efficiency, storage economy, and downlink dependability. By first applying Delta Encoding to remove repetition between successive sensor readings and then processing the result through Heat shrink's compact compression algorithm, the system effectively reduces payload size. This dual-stage approach lowers the volume of data transmitted, decreases communication overhead, and makes smarter use of the satellite's restricted memory and bandwidth. Testing confirms the system performs real-time compression with negligible delay and low energy use, a critical requirement for power-limited platforms like CubeSats. Its efficiency directly supports mission reliability by ensuring vital telemetry can be stored and transmitted within tight energy and bandwidth allowances.

10.48047/jocaaa.2025.34.12.25

Future scope

1. Adaptive Compression Framework– Employing lightweight machine learning to dynamically select the best compression method based on changing data patterns and mission stage.
2. Hybrid Compression Techniques – Merging the Delta Encoding + Heatshrink approach with other efficient standards, such as Huffman coding or the CCSDS 121.0-B-3 space data standard, to achieve higher compression ratios.
3. Error-Resilient Transmission – Incorporating forward error correction (FEC) and packet-level redundancy to safeguard data integrity against radiation-induced corruption during downlink.
4. Onboard Data Prioritization – Implementing intelligent filtering logic to identify and prioritize high-value or anomalous data streams before compression and transmission.
5. Scalability for Multi-Sensor Systems – Expanding the compression pipeline to efficiently manage data from diverse, high-volume payloads, including imagers and environmental sensors.
6. Integration with Inter-Satellite Links – Adapting the optimized system for use in inter-satellite communication, enabling efficient data relay within future satellite constellations.
7. Flight Validation – Progressing to hardware-in-the-loop testing and eventual in-orbit demonstration to prove the system's performance under genuine mission conditions.

REFERENCES

1) J.Smith et al., "MiniaturizedSensorSystems for CubeSats,"IEEE Sensors Journal,2021,discussed lightweight andpower-efficient sensors tailored for CubeSat missions.

2) R.Kumaretal.,"Multi-SensorDataFusionforCubeSats,"AerospaceSystems,2020,exploredthefusionofmultiplesensorinputsto improve data accuracy and reliability.

3) L.Wang et al., "Low-Power Telemetry System for Nanosatellites," Acta Astronautica, 2019, evaluated various communication protocols, such as UHF/VHF, S band, and X-band, for telemetry applications..

4) M. Anderson et al., "CubeSat Communication Architecture: Challenges and Solutions," IEEE Aerospace Conference Proceedings, 2021,highlighted the use of Software-Defined Radios (SDRs) to enhance communication flexibility..

5) P.Davies et al., "FPGA-based Onboard Processing for CubeSat DAS," Journal of Small Satellites, 2022, presented an FPGA-based data processing architecture for real-time telemetry.

6) S.Tanakaetal.,"EdgeComputinginSpace:AnApplicationforCubeSats,"IEEETransactionsonAerospaceandElectronicSystems,2020, explored machine learning techniques for autonomous data analysis onboard CubeSats.

7) K.R.Laletal.,"EfficientTelemetryEncodingTechniquesforCubeSatMissions,"IEEEAerospaceConference,pp.1–7, 2021.

8) A. Kanatas, G. D. Sideris, I. Sakellariou, E. Kosmatopoulos and G. K. Karagiannidis, "An Efficient Hybrid EntropyCoder for Space-Grade FPGA," arXiv preprint, arXiv:2205.04123, May 2022.

9) S.Mattiazzo,T.Ott,D.Andreis,D.Sanchez-Fernandez,andSoldovieri,"LosslessHyperspectralImageCompressionforComet Interceptor and Hera Missions," Remote Sensing, vol. 17, no. 5, pp. 1–20, 2023

10) K.KambleandS.Sankaranarayanan,"ASystematicReviewofHardware-AcceleratedDataCompressionforEmbeddedSystems," Sensors, vol. 22, no. 1, pp. 1–17, 2022.

11) A.Reina,V.Mazzia,D.Giordano,andM.Chiaberge,"DataDownlinkPrioritizationviaOn-OrbitClassification,"arXivpreprint, arXiv:2408.14865, Aug. 2024.

12) I. Goienetxea, A. Almeida, and D. López-de-Ipiña, "Dynamic Bit-Packing for Efficient Time-Series Data Transmission," Sensors, vol. 23, no. 20, pp. 8575, 2023.

13) A.Pal, A. Kar, and A. Palit, "Delta Encoding and TDD LoRa Protocol for Environmental Sensor Networks," Journal of Signal Processing Systems, vol. 95, pp. 345–361, 2023.

14) A. Bussonnier and P.-E. Dagand, "Hybrid Data-Differencing and Compression for Embedded Systems," Entropy, vol. 24, no. 5, pp. 574, 2024

15) B.Shaik,A.K.Mohapatra,andB.Panda,"EfficientLosslessCompressionandFaultDetectioninTelemetryD

10.48047/jocaaa.2025.34.12.25

ata,"WirelessPersonal Communications, vol. 121, pp. 279–295, 2021.

16) G.Falcioni,M.Varacalli,andP.S.Rossi,"AReviewofAnomalyDetectioninSpacecraftTelemetry,"Applied Sciences,vol.15,no.10,
17) pp.5653, 2023
18) M.U.Hassan,M.Asim,M.U.Saeed,A.Tahir,S.A.Hassan,andH.Farooq,"ChannelModelingandCommunicationAnalysisfor CubeSats," Expert Systems with Applications, vol. 206, pp. 117937, 2022.