Agrociencia

BLOCKCHAIN-BASED CROP MONITORING USING AN INTERPLANETARY FILE SYSTEM

Jananee **Vinayagam**[1], Dhivya **Baskar**[2], Tamilvizhi **Thanarajan**[3*] Mahendran **Rajamanickam**[4]

[1]SRM Institute of science and Technology, Department of Computer science and Engineering, Ramapuram, Chennai, 600089, India.
[2]Karpaga Vinayaga College of Engineering and Technology. Department of Computer Science and Engineering. Chengalpattu, Tamil Nadu 603308, India.
[3]Panimalar Engineering College. Department of Computer Science and Engineering. Chennai, Tamil Nadu 600123, India.
[4]Koneru Lakshmaiah Education Foundation. Computer Science and Engineering. Guntur, Andhra Pradesh 522502, India.
* Author for correspondence: tamilvizhi.phd.it@gmail.com

Citation: Vinayagam J, Baskar D, Thanarajan T, Rajamanickam M. 2025. Blockchain-Based crop monitoring using an interplanetary file system. **Agrociencia.** https://doi.org/ 10.47163/agrociencia.v59i8.3553

Editor in Chief:
Dr. Fernando C. Gómez Merino

Received: July 10, 2025.
Approved: November 27, 2025.
Published in Agrociencia: December 04, 2025.

ABSTRACT

Continuous monitoring of crops is essential for early issue detection and data-driven decision making, which supports improved yield, quality, and environmental sustainability in agriculture. Blockchain technology has been used for continuous monitoring of agricultural fields, although its adoption remains limited due to economic and technological constraints. This study proposes an enhanced crop-monitoring framework with InterPlanetary File System (IPFS) technology to overcome the limitations of existing models. Experimental evaluation showed that the proposed model achieved 98 % accuracy and a 99.8 % confidentiality rate. System throughput increased from 3 kb s^{-1} for 10 kb of data to 307 kb s^{-1} for 3000 kb, demonstrating strong scalability. By integrating decentralized IPFS storage with blockchain-based traceability, the system ensures data integrity, cost-efficiency, and reliable real-time monitoring, offering a practical solution for precision agriculture. These findings demonstrate the potential of the IPFS-enabled framework as an efficient alternative to conventional blockchain-dependent models.

Keywords: Agriculture, blockchain technology, machine learning.

INTRODUCTION

Crop monitoring has become essential in modern agriculture, with studies indicating yield increases of up to 25 % and water savings of up to 30 % (Panek-Chwastyk *et al*. (2024). By integrating technologies such as satellite imagery and drones, farmers can detect crop stressors, including pests and diseases, with accuracies approaching 90 %, allowing for timely and effective interventions Shammi *et al*., (2024). Precise monitoring also optimizes resource allocation, saving fertilizer by up to 50 % and reducing pesticide usage by 20–30 %, which reinforces economic gains while supporting ecological balance through reduced chemical leaching and soil deterioration Padmini and Kuzhalvaimozhi, (2024). Additionally, early detection of yield fluctuations

supports informed decision-making, saving up to 30 United States Dollars (USD) per acre through optimized harvesting and marketing strategies. Overall, the integration of robust monitoring systems can improve returns on investment by up to 300 %, reinforcing their role as a cornerstone of modern farming operations Padmini and Kuzhalvaimozhi, (2024).

Information and communication technologies (ICTs) are widely used in agriculture and are closely related to process regulation and farming efficiency. Internet of Things (IoT)-based agricultural monitoring systems collect crop data in real time, allowing for more informed decisions about planting, watering, and pest control. By delivering precise, up-to-date data, IoT-based observation devices help farmers optimize the use of agricultural inputs Mahalingam and Sharma, (2023). By limiting hazardous chemicals, reducing waste, and maximizing resource utilization, these systems help farmers reduce their environmental impact. Real-time data enables close observation and regulation of crop growth conditions, resulting in higher-quality harvests that are less susceptible to disease and pests. Farmers can also increase their earnings by lowering the costs of labor, nutrients, water, and other supplies using IoT-based agricultural monitoring systems Mahalingam and Sharma, (2023)..

The various functions performed by IoT in agriculture include continuous crop monitoring, tracking and tracing agricultural products from seeding to selling, farming using Unmanned Aerial Vehicles (UAVs), supply chain management, precision farming, and aquaponics farms. To support and enhance these capabilities, Panek-Chwastyk *et al*. (2024) examined the viability of acclimatizing the Earth Observation for Agricultural Statistics (EOstat) crop monitoring system, primarily implemented for continuous observation of crop growth in Poland. The system used data collected from satellites and agrometeorological information obtained through the Copernicus program and utilized a machine learning algorithm and the Extreme Gradient Boosting Regressor (XGBoost) for the identification of each unit.

Padmini and Kuzhalvaimozhi (2024) integrated IoT and Wireless Sensor Networks (WSNs) for real-time crop monitoring. To overcome the problem of hotspots and address load balancing, they introduced the Energy and Delay Aware Routing (EDAR) organization for direction finding in IoT applications. Wei and Fang (2024) targeted the implementation of a system for monitoring plant health in vertical farms by using multispectral LEDs with near-infrared (NIR) and ultraviolet A (UVA) light sources. IoT was responsible for controlling the LED spectrum, and a camera was connected to the IoT device with switchable filters and RGB CMOS sensors to estimate UV-NDVI and SI-NDVI.

Effective plant monitoring remains essential for enhancing and sustaining agricultural productivity and supporting informed agricultural management. Maity *et al*. (2024) developed a system that utilizes Passive Infrared (PIR) sensors to detect animal intrusion, smoke, and soil moisture levels in agricultural fields. This system integrates three core components: Arduino-based IoT devices, a web server application, and a smartphone interface for real-time monitoring and control. Similarly, Eisfelder *et*

al. (2024) introduced an approach that leverages Sentinel-1 (S1) and Sentinel-2 (S2) time-series data to classify crop types and cropland, evaluating the performance of 33 Random Forest models.

Several studies have incorporated intelligent technologies to improve crop monitoring. Stephen *et al*. (2023) applied deep learning methods using multiple Convolutional Neural Network (CNN) architectures, while Omia *et al*. (2023) evaluated remote sensing through spectral imagery. Kumar Pradhan *et al*. (2024) developed an Artificial Intelligence (AI) and IoT-based system for continuous monitoring and environmental detection, and Akilan and Baalamurugan (2024) created a sensor-based framework using an Adaptive Gaussian Filter (AGF) and ResNet50. Additional advances include IoT and deep learning models such as MobileNetV2, EfficientNetB0, and Mobile-UNet (Morellos *et al*., 2024), Model-Based Safety Analysis (MBSA) for identifying module failures (Abdulhamid *et al*., 2024), and CNN-based approaches including VGG16, DenseNet121, MobileNetV2, and ResNet50 for agricultural monitoring (Peng *et al*., 2024).

Crop monitoring can be significantly strengthened through the integration of advanced image recognition techniques. Shammi *et al*. (2024) focused on soybean growth assessment by employing multiple technologies for yield prediction. Li and Shi (2024) introduced an end-to-end Semi-Supervised Object Detection (SSOD) approach based on Detection Transformer (DETR) for monitoring the growth of *Brassica chinensis*, incorporating two additional strategies to address class imbalance and support multi-task optimization. Similarly, Nduku *et al*. (2024) used UAV imagery to estimate crop height in two winter wheat farms, analyzing the resulting datasets with machine learning models.

The growing reliance on IoT in agriculture introduces significant security risks, as interconnected devices are increasingly vulnerable to data breaches and malicious attacks. To address these concerns and overcome the limitations of existing techniques, an IoT-based crop monitoring system incorporating InterPlanetary File System (IPFS) technology was proposed.

Objectives and hypothesis

The primary objective of this study is to design and validate a secure, scalable crop-monitoring framework that integrates Internet of Things (IoT) sensing, cryptographic data protection, blockchain-based traceability, and InterPlanetary File System (IPFS) decentralized storage for efficient agricultural data management. Specifically, the study aims to (i) enhance data confidentiality and integrity using AES encryption and SHA-256 hashing mechanisms, (ii) improve system throughput and reduce execution time compared with conventional blockchain-based architectures, and (iii) assess the effectiveness of Support Vector Machine (SVM) models for crop monitoring and predictive analysis using preprocessed sensor and image data.

The hypothesis of this study is that the integration of IPFS-based decentralized storage with blockchain traceability and cryptographic security mechanisms will result in

significantly higher throughput, lower processing time, and improved confidentiality while maintaining data reliability when compared with traditional centralized or blockchain-only crop monitoring systems.

MATERIALS AND METHODS

A prediction system was proposed that integrates a security device with cloud infrastructure to safeguard data from attackers. Existing approaches suffer from limitations in their data-filtering mechanisms, as they often fail to adequately remove noise, irrelevant attributes, or corrupted sensor readings, leading to inaccurate analytical results. The proposed model addresses these issues by incorporating min-max scaling for preprocessing, the Advanced Encryption Standard (AES) for secure data acquisition, and a Support Vector Machine (SVM) for data analysis. This integrated approach enhances security, integrity, efficiency, accuracy, and traceability within IoT-based crop monitoring systems.

Data acquisition

Crop data is acquired from the cloud and analyzed using cloud computing infrastructure. This information includes soil moisture, humidity, temperature, crop growth stages, and satellite imagery. Cloud-based storage ensures secure, scalable, and continuous access to agricultural data, acting as the central repository for heterogeneous inputs from IoT devices, sensors, and remote sensing sources. It supports efficient storage and retrieval of large datasets, enables real-time synchronization between field devices, preprocessing modules, and machine learning workflows, and provides fault tolerance and backup to minimize data loss. Additionally, cloud storage integrates seamlessly with IPFS, enhancing security, traceability, and decentralized access.

Preprocessing

MIN-MAX scaling (minimum–maximum normalization) is used for data preprocessing. In crop monitoring, variables such as temperature, soil moisture, and sunlight intensity each have different measurement ranges. For example, soil moisture may vary from 0 to 100 %, while temperature may range from -20 to 40 °C. Scaling these values to a common range (such as 0 to 1) facilitates comparison, pattern detection, and correlation analysis across environmental factors. By standardizing and harmonizing heterogeneous inputs, MIN-MAX scaling plays a key role in improving the performance of crop monitoring systems. The preprocessing equation using MIN-MAX scaling is given in the following equation:

$$X' = \frac{(x - x_{min})}{X_{max} - X_{min}}$$

Cryptographic analysis

Modifications can be detected by comparing newly generated hashes of updated material with the hashes previously stored on IPFS. This process removes the need for decryption, which often requires exposing private keys and introduces security risks, so it is only required when recovering earlier data. Because integrity checks can be performed without revealing the underlying content, this method also helps maintain a clear separation of concerns.

The function CREATEDATAFORIPFS takes a list of records corresponding to database rows. For each row, it computes a hash of the content and produces tuples (*idi*, *H*(*recordi*)), where *H*() denotes the hashing function. The full content is then encrypted using the function *E*(), and both components are stored together on IPFS. The system uses the SHA-256 algorithm to generate fixed 256-bit cryptographic hashes, which uniquely represent the input data. Even minimal changes in the input cause substantial differences in the resulting hash, a property known as the avalanche effect.

Hashing

The hashing process can be represented as follows:

$$H\,(D) = SHA - 256\,(D)$$

where *H* denotes the hash function based on the SHA-256 technique and *D* indicates the data entry.

In the encryption stage, the Password-Based Key Derivation Function 2 (PBKDF2) is used as a key derivation function with an adjustable computational cost designed to prevent vulnerability to brute-force attacks. It generates a cryptographic key from a confidential secret, eliminating the need to store the encryption key in a central database and thus improving overall security.

Key derivation

The derived key *K*, used for both encryption and decryption, is obtained as follows:

$$K = KDF\,(P, S, N, L)$$

where the user-provided password is denoted by *P*, and the key derivation function by *KDF*.

This method does not save any encryption key. Instead, it encrypts a predefined string *S* using the key derived from the user password *P*, and the resulting ciphertext is stored on the IPFS. When a user later provides a password *P′*, a corresponding key *K′* is generated using the same KDF function. The stored ciphertext is then decrypted with *K′*. If the decrypted value matches the original string *S*, the password is validated and the verification succeeds; otherwise, access is denied. After successful verification, the derived key *K* is used for encryption operations.

Encryption

The encrypted data (*ED*) can be expressed as follows:

$$ED = E(K, D)$$

where *E* denotes the encryption function, *K* represents the secret key, and *D* refers to the data being encrypted.

The process begins with the generation of combined data, which is stored on IPFS, producing a unique Content Identifier (CID) that represents the encrypted and hashed content. Once this CID is generated, the blockchain stage is initiated. The CID is recorded on the blockchain through a smart contract, which maintains an immutable list of all CIDs associated with IPFS-stored data. This ensures permanent, tamper-proof tracking of data integrity and enables auditable traceability by Krishnan N *et al*. (2024).

The primary objectives of this smart contract are to store CIDs, retrieve them, and support verification of persisted data. The DATAVERIFICATION function (Algorithm 1) performs an integrity check by comparing the hashes of the current centralized database entries (Dapi) with those stored in IPFS (Db). It determines whether each row exists on the blockchain and whether its stored hash matches the current one. New records are uploaded to IPFS, and the resulting CIDs are added to the blockchain. Records with mismatched hashes are flagged as corrupted or irregular. Although homogeneity checks rely solely on hash comparison without requiring decryption, system policies may allow corrupted data to be restored by decrypting the corresponding encrypted content stored under the associated CID.

Algorithm 1: DATAVERIFICATION
Input: Current database records Dapi
Output: Verification result and blockchain update
For each record in Dapi do:
  Compute current hash Hcurrent = SHA-256(record)
  Retrieve stored hash Hstored from IPFS blockchain ledger
  If Hcurrent == Hstored then
    Mark record as verified
  Else
    Flag record as corrupted or altered
    Upload new encrypted data to IPFS
    Store new CID on blockchain
  End If
End For

Monitoring

Once the data is collected and preprocessed, Support Vector Machines (SVMs) are used for both regression and classification tasks in agriculture due to their strong mathematical foundations. These work by identifying the optimal hyperplane that separates classes or predicts continuous outcomes, with the closest data points (support vectors) defining this boundary. The optimization process seeks the hyperplane that maximizes the margin between support vectors, often solved using quadratic programming or gradient-based methods.

In agricultural applications, SVMs are trained on feature vectors extracted from satellite imagery and sensor data, including vegetation indices, temperature, and soil moisture. These features represent the input vectors (x) for the SVM:

The SVM classification and regression functions are defined as:

$f(x) = w \cdot x + b$

where *w* is the weight vector defining the separating hyperplane, *x* is the feature vector, and *b* is the bias term.

The optimal hyperplane is obtained by minimizing the objective function:

$\min (1/2 \, ||w||^2 + C \sum \xi i)$

subject to:

$yi \, (w \cdot xi + b) \geq 1 - \xi i$

where *C* is the regularization parameter that controls the trade-off between maximizing the margin and minimizing classification errors, *xi* is the input sample, *yi* is the class label, and *ξi* represents the slack variable allowing misclassifications. Through optimization of the parameters w and b, SVMs can classify crops, detect pests, predict yields, and evaluate crop health effectively. Additionally, the proposed method enhances data integrity, robustness, and secure retrieval by preventing third-party interference. System privacy is improved through hash-based comparisons, allowing verification without decrypting the stored data.

Experimental setup

The proposed system was implemented and evaluated in a controlled laboratory setting. The hardware setup included a workstation with an Intel Core i7 processor, 16 GB RAM, and a 1 TB SSD running Ubuntu 22.04 LTS. IoT data were simulated using Arduino Uno boards equipped with DHT11 temperature-humidity sensors and soil moisture sensors. A 2.4/5 GHz dual-band Wi-Fi 6 router was used for data transmission between the IoT devices and the cloud server.

The cloud environment was deployed on Google Cloud Platform (GCP), using Cloud Storage Buckets for data storage and virtual machines with 4 vCPUs and 8 GB RAM for computation. InterPlanetary File System (IPFS) version 0.21.0 was installed on the Ubuntu server (version 0.21.0), and SHA-256 hashing was integrated for data integrity verification. The software stack consisted of Python 3.10 for preprocessing (Min-

Max scaling), TensorFlow for SVM-based machine learning, and OpenSSL for AES encryption/decryption. Network performance was monitored with Wireshark, and MATLAB R2022a was used to generate comparative analysis graphs.

Performance enhancement

The performance of the proposed system was evaluated using two primary metrics: total processing time and throughput, with the goal of assessing how different data sizes influence encryption/decryption speed and overall system efficiency.

Throughput

The throughput (T) is defined as the volume of data (D) processed per unit time (t) during encryption or decryption. The relative throughput for encoded data (ED) and decoded data (DD) is defined as:

$$TE = \frac{size(ED)}{t}$$

$$TD = \frac{size(DD)}{t}$$

Total time

The total time is calculated as:

$$t = t_e - t_i$$

representing the difference between the start time *ti* and completion time *te* of a given operation. Measurements were taken for encoding, decoding, hashing, and storing encrypted data on IPFS across data sizes ranging from 10 to 3000 kb.

Scalability

System scalability was assessed by evaluating overall throughput across all processing stages, using the same data-size range (10–3000 kb). This analysis reflects the system's capacity to maintain performance as workload increases.

Data integrity

While AES and similar algorithms ensure confidentiality, they do not inherently protect data integrity. Encrypted data can still be modified without detection, potentially leading to corrupted but decryptable outputs. The proposed system addresses this limitation by incorporating hash-based integrity verification. A hash value (the data's unique electronic fingerprint) is computed before encoding. After decoding, the hash is recomputed and compared with the original. Matching hashes confirm that no tampering occurred during storage or transmission. This mechanism provides robust

protection against unauthorized modifications, strengthening the overall security of the system.

Performance metrics

The evaluation of the proposed system relied on several standard performance metrics. Accuracy measures the proportion of correctly classified instances among all instances, precision quantifies the proportion of true positive predictions among all predicted positives, and recall (or sensitivity) captures the proportion of true positives identified among all actual positive instances. The F1 score, defined as the harmonic mean of precision and recall, provides a balanced measure that accounts for both prediction correctness and completeness.

Accuracy was evaluated based on the classification output of the Support Vector Machine (SVM) model using a labeled dataset of crop health samples. The dataset was divided into training (70%) and testing (30%) subsets. Accuracy was calculated as the ratio of correctly predicted samples to the total number of evaluated samples, expressed as:

$$\text{Accuracy} = \frac{(TP+TN)}{(TP+TN+FP+FN)}$$

where TP represents true positives, TN true negatives, FP false positives, and FN false negatives.

The confidentiality rate was evaluated by measuring the proportion of successfully encrypted and securely verified data transmissions relative to the total number of data packets processed by the system. Data integrity verification was performed through SHA-256 hash matching between locally generated hashes and hashes stored in the IPFS ledger following encryption and decryption processes. The confidentiality rate was expressed as:

Confidentiality Rate = (Number of securely verified encrypted records / Total transmitted records) × 100%

Multiple transmission trials were conducted across varying data volumes to obtain stable average values for both accuracy and confidentiality evaluations.

RESULTS AND DISCUSSION

The use of IPFS allowed agricultural data to be stored and accessed in a decentralized manner, ensuring availability, integrity, and resilience across multiple nodes. This distributed architecture reduced vulnerability to attacks compared with traditional centralized storage systems. The integration of Min-Max scaling as a preprocessing step further enhanced machine learning performance by normalizing heterogeneous

feature values into a common range, resulting in more stable, accurate, and robust models. SVMs were also effective in the final analytical stage of crop monitoring tasks, such as crop classification, pest detection, and yield prediction, thanks to their strong regression capabilities and ability to capture complex nonlinear patterns via kernel functions.

The system incorporated the InterPlanetary File System (IPFS) and cloud infrastructure to achieve scalable and distributed data management. By distributing information across multiple nodes rather than relying on a single centralized server, the architecture minimizes performance bottlenecks and supports larger datasets and higher transaction volumes without degradation in speed. Cost efficiency was achieved by avoiding the computational and energy demands of traditional blockchain consensus mechanisms; instead, IPFS combined with AES-based encryption provided lightweight security and integrity verification. Additionally, the use of cloud resources on a pay-as-you-go model limits upfront investment and ensures that resources can be scaled according to demand.

A positive correlation was observed between system throughput and data size. Throughput measured approximately 3.07 kb s^{-1} for a 10 kb dataset but increased sharply to over 307.12 kb s^{-1} when processing 3000 kb. Lower throughput at small data sizes was primarily due to the disproportionate impact of initial encoding and decoding overhead, which becomes negligible as data volumes grow. As dataset size increases, the system operates more efficiently, highlighting the scalability of AES for large data workloads. These results demonstrate that AES supports secure, efficient encoding and decoding even when handling substantial quantities of information.

AES was selected for its strong security guarantees and its ability to efficiently process large datasets. As a block-cipher algorithm supporting 128-, 192-, and 256-bit keys, AES enables parallel processing of data blocks, minimizing computational overhead during both encryption and decryption. This efficiency is especially advantageous in agricultural applications, where continuous streams of sensor readings, satellite imagery, and IoT data produce substantial data volumes. Compared with older algorithms such as Data Encryption Standard (DES) and Triple-DES, AES offers significantly higher throughput while maintaining robust resistance to brute-force and cryptanalytic attacks. Its lightweight computational requirements ensure practical performance even with multi-megabyte inputs, making AES a scalable and secure choice for real-time crop monitoring and large-scale agricultural data management.

The results indicate that processing time increases with data size. For a 10 kb dataset, encryption, decryption, and hashing required approximately 2472, 2511, and 1 ms, respectively, while storing the data on IPFS took 720 ms. When the dataset size increased to 3000 kb, processing times rose to 3546 ms for encryption, 3490 ms for decryption, 190 ms for hashing, and 5295 ms for IPFS storage (Figure 1).

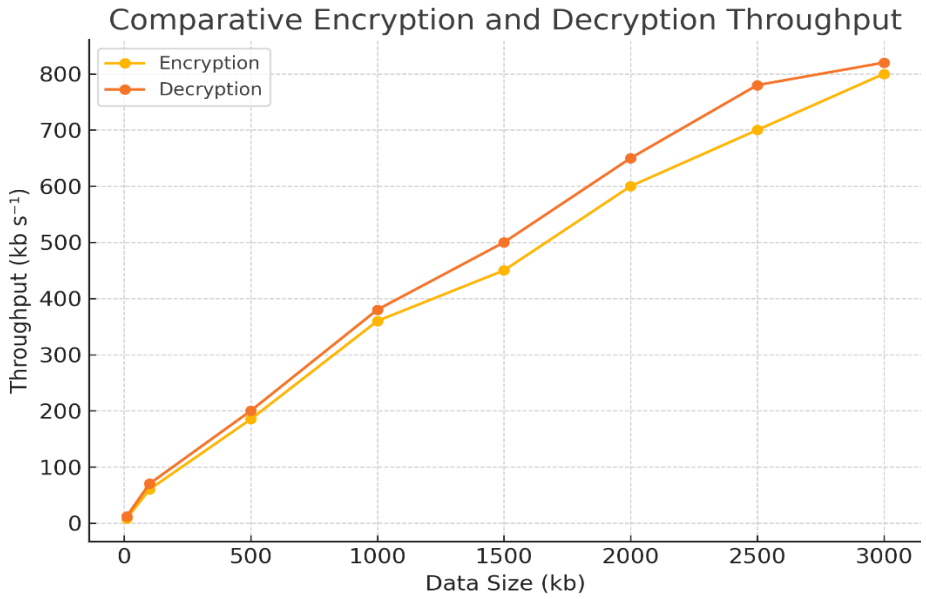

Figure 1. Comparative encryption and decryption throughput of the Advanced Encryption Standard (AES)-based module within the InterPlanetary File System (IPFS)-integrated crop monitoring system.

The consistent increase in computation time with larger datasets confirms the effectiveness of the AES and SHA-256 mechanisms. IPFS requires additional time for data storage due to network latency, data transfer, and processing overhead; however, this duration is acceptable in real-world scenarios since data preservation can be performed offline, unlike other systems.

The overall throughput of the model, including data encoding, hashing, and storage of encrypted IPFS data, was approximately 3 kb s^{-1} for a 10 kb dataset (Figure 2). As the dataset size increased to 3000 kb, throughput rose significantly to over 307 kb s^{-1}. These results demonstrate the model's ability to efficiently process large datasets, highlighting its scalability. The proposed model achieved 98 % accuracy, 99.8 % confidentiality rate, and reduced execution time to 9 ms. Compared to other systems, the model performed better across all evaluated metrics (Table 1 and Figure 3).

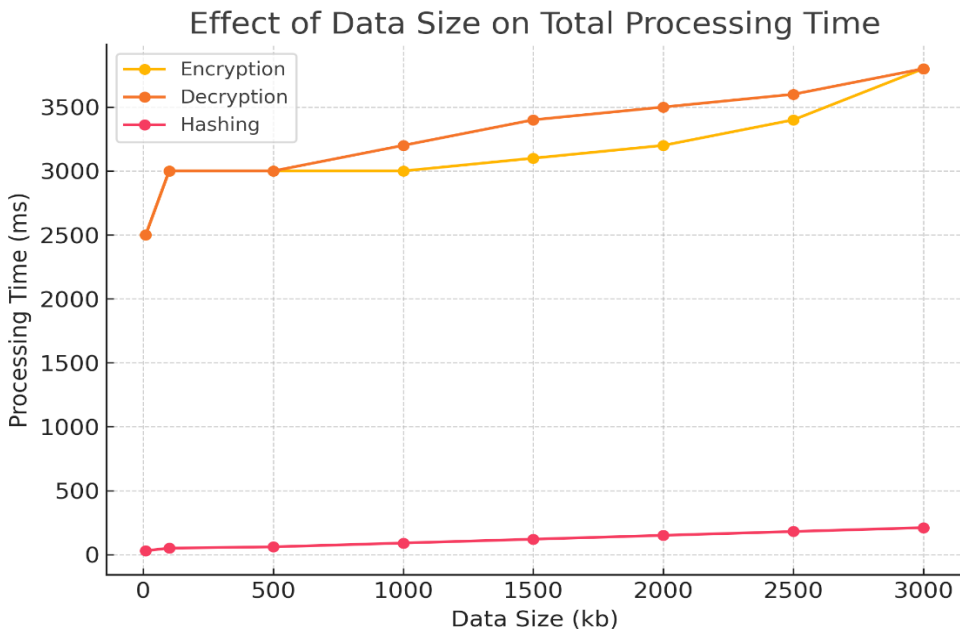

Figure 2. Effect of data size on total processing time for Advanced Encryption Standard (AES) and InterPlanetary File System (IPFS) operations.

Table 1. Comparative evaluation of various cryptographic techniques.

Technology	Encryption time (ms)	Decryption time (ms)	Execution time (ms)	Confidentiality rate (%)
Advanced Cipher Encryption-Blockchain (ACE-BC) (Alharbi, 2023)	-	-	-	97.54
Double Bilinear Diffie-Hellman (DBDH) (Goel and Neduncheliyan, 2023)	3.55	3.6	12	99.52
Secure Stream Encryption and Robust Block Algorithm (SSERBA) (Alotaibi Y 2024)	25	35	25	84
Elliptic Curve Integrated Encryption Algorithm (ECIEA) (Pourvahab and Ekbatanifard, 2019)	70	90	30	79
CLIENT (Ramamoorthi and Appathurai, 2023)	3.22	3.27	-	-
Elliptic Curve Cryptography-Secure Authentication Scheme (ECCSAS) (Velmurugadass *et al*., 2021)	50	53	50	76
Randomized Network Encryption for Cloud Blockchain (RNECB) (Mahalingam and Sharma, 2023)	2.7	2.6	11	98.89
Proposed model (InterPlanetary File System, IPFS)	2.5	2.4	9	99.8

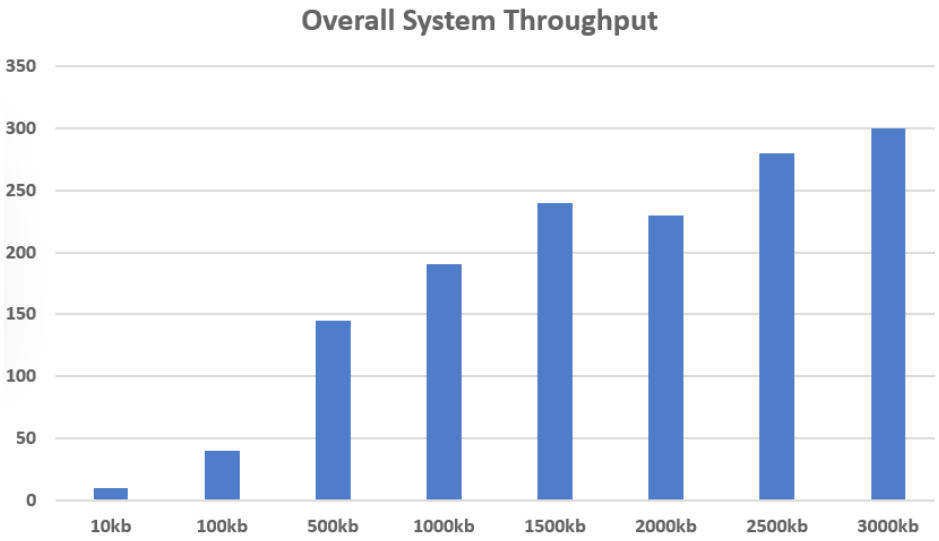

Figure 3. Overall throughput of the proposed crop monitoring system.

CONCLUSION

This study presented a blockchain-enabled crop monitoring framework that integrates the InterPlanetary File System (IPFS) to provide secure, scalable, and efficient management of agricultural data. The combination of decentralized storage with blockchain-based traceability strengthens data integrity, reduces costs, and supports reliable real-time monitoring. Although the results are promising, the proposed system has certain limitations. Its evaluation was conducted mainly with controlled datasets, and large-scale field validation using extensive IoT deployments is still required. Furthermore, while Advanced Encryption Standard (AES) offers strong security and scalability, computational delays may arise in resource-constrained devices when processing high-frequency sensor streams.

Future work will focus on incorporating advanced deep learning models for more robust multi-disease detection, integrating IoT sensors with mobile applications for real-time alerts, and applying federated learning to enhance privacy and collaborative training. Expanding the system with multilingual interfaces and region-specific datasets will further improve usability and promote broader adoption among small and marginal farmers, strengthening the inclusiveness and sustainability of the proposed framework.

FUTURE SCOPE

Future work will focus on extending the system to large-scale real-world field deployments, integrating advanced deep learning models for multi-disease detection,

and developing a mobile application interface for real-time farmer alerts. Additional enhancements will include the adoption of federated learning techniques to improve data privacy and collaborative model training across geographically distributed farms.

DATA AVAILABILITY

https://www.kaggle.com/datasets/osamajalilhassan/agriculture-crops-dataset

REFERENCES

Abdulhamid A, Rahman MM, Kabir S, Ghafir I. 2024. Enhancing safety in IoT systems: A model-based assessment of a smart irrigation system using fault tree analysis. Electronics 13 (6): 1156. https://doi.org/10.3390/electronics13061156

Akilan T, Baalamurugan KM. 2024. Automated weather forecasting and field monitoring using GRU-CNN model along with IoT to support precision agriculture. Expert Systems with Applications 249: 123468. https://doi.org/10.1016/j.eswa.2024.123468

Alharbi A. 2023. Applying access control enabled blockchain (ACE-BC) framework to manage data security in the CIS system. Sensors 23 (6): 3020. https://doi.org/10.3390/s23063020

Alotaibi, Y., Rajendran, B. and Rajendran, S., 2024. Dipper throated optimization with deep convolutional neural network-based crop classification for remote sensing image analysis. PeerJ Computer Science, 10, p.e1828. https://doi.org/10.7717/peerj-cs.1828

Eisfelder C, Boemke B, Gessner U, Sogno P, Alemu G, Hailu R, Mesmer C, Huth J. 2024. Cropland and crop type classification with Sentinel-1 and Sentinel-2 time series using Google Earth engine for agricultural monitoring in Ethiopia. Remote Sensing 16 (5): 866. https://doi.org/10.3390/rs16050866

Goel A, Neduncheliyan S. 2023. An intelligent blockchain strategy for decentralised healthcare framework. Peer-to-Peer Networking and Applications 16 (2): 846–857. 10.1007/s12083https://doi.org/-022-01429-x

Krishnan, N., Surendran, R. and Nathan, M., 2022, December. Crop tracker-A web application to sell or buy crops and predict crop price using machine learning. In 6th Smart Cities Symposium (SCS 2022), 2022: 152-156. IET. https://doi.org/10.1049/icp.2023.0386.

Kumar Pradhan A, Kumar D, Chaurasia H, Murmu S, Samal I, Kumar Bhoi T, Kumar GAK, Mondal B. 2024. Crop monitoring system integrating with Internet of Things and artificial intelligence. *In* Advances in Geographical and Environmental Sciences Springer Nature: Singapore, pp: 193–208. https://doi.org/10.1007/978-981-97-0341-8_10

Li H, Shi F. 2024. A DETR-like detector-based semi-supervised object detection method for *Brassica Chinensis* growth monitoring. Computers and Electronics in Agriculture 219: 108788. https://doi.org/10.1016/j.compag.2024.108788

Mahalingam N, Sharma P. 2023. An intelligent blockchain technology for securing an IoT-based agriculture monitoring system. Multimedia Tools and Applications 83 (4): 10297–10320. https://doi.org/10.1007/s11042-023-15985-8

Maity T, Paul S, Samanta J, Saha P. 2024. Design and development of IoT-based SmartTech-agri devices for smart agriculture crop field. Journal of The Institution of Engineers (India): Series B 105 (4): 753–762. https://doi.org/10.1007/s40031-024-01002-5

Morellos A, Dolaptsis K, Tziotzios G, Pantazi XE, Kateris D, Berruto R, Bochtis D. 2024. An IoT transfer learning-based service for the health status monitoring of grapevines. Applied Sciences 14 (3): 1049. https://doi.org/10.3390/app14031049

Nduku L, Munghemezulu C, Mashaba-Munghemezulu Z, Masiza W, Ratshiedana PE, Kalumba AM, Chirima JG. 2024. Field-Scale winter wheat growth prediction applying machine learning methods with unmanned aerial vehicle imagery and soil properties. Land 13 (3): 299. https://doi.org/10.3390/land13030299

Omia E, Bae H, Park E, Kim MS, Baek I, Kabenge I, Cho BK. 2023. Remote sensing in field crop monitoring: A comprehensive review of sensor systems, data analyses and recent advances. Remote Sensing 15 (2): 354. https://doi.org/10.3390/rs15020354

Padmini MS, Kuzhalvaimozhi S. 2024. Energy and delay aware routing model for smart crop monitoring application using internet of things. Computers and Electrical Engineering 116: 109207. https://doi.org/10.1016/j.compeleceng.2024.109207

Panek-Chwastyk E, Dąbrowska-Zielińska K, Kluczek M, Markowska A, Woźniak E, Bartold M, Ruciński M, Wojtkowski C, Aleksandrowicz S, Gromny E, *et al*. 2024. Estimates of crop yield anomalies for 2022 in Ukraine based on Copernicus Sentinel-1, Sentinel-3 Satellite Data, and ERA-5 agrometeorological indicators. Sensors 24 (7): 2257. https://doi.org/10.3390/s24072257

Peng M, Liu Y, Khan A, Ahmed B, Sarker SK, Ghadi YY, Bhatti UA, Al-Razgan M, Ali YA. 2024. Crop monitoring using remote sensing land use and land change data: Comparative analysis of deep learning methods using pre-trained CNN models. Big Data Research 36: 100448. https://doi.org/10.1016/j.bdr.2024.100448

Pourvahab M, Ekbatanifard G. 2019. Digital forensics architecture for evidence collection and provenance preservation in IaaS cloud environment using SDN and blockchain technology. IEEE Access 7: 153349–153364. https://doi.org/10.1109/access.2019.2946978

Ramamoorthi S, Appathurai A. 2023. Energy aware clustered blockchain data for IoT: An end-to-end lightweight secure and enroute filtering approach. Computer Communications 202: 166–182. https://doi.org/10.1016/j.comcom.2023.02.010

Reddy, P.S., Surendran, R., Divya, K., Raveena, S., Selvaperumal, S.K. and Lakshamanan, R., 2024, August. Accuracy analysis and comparison of crop yield prediction using NB algorithm and RNN. In AIP Conference Proceedings, 3161(1). p. 020363. AIP Publishing LLC. https://doi.org/10.1063/5.0229471.

Shammi SA, Huang Y, Feng G, Tewolde H, Zhang X, Jenkins J, Shankle M. 2024. Application of UAV multispectral imaging to monitor soybean growth with yield prediction through machine learning. Agronomy 14 (4): 672. https://doi.org/10.3390/agronomy14040672

Stephen A, Arumugam P, Arumugam C. 2023. An efficient deep learning with a big data-based cotton plant monitoring system. International Journal of Information Technology 16 (1): 145–151. https://doi.org/10.1007/s41870-023-01536-9

Velmurugadass P, Dhanasekaran S, Shasi Anand S, Vasudevan V. 2021. Enhancing blockchain security in cloud computing with IoT environment using ECIES and cryptography hash algorithm. Materials Today: Proceedings 37: 2653–2659. https://doi.org/10.1016/j.matpr.2020.08.519

Wei Z, Fang W. 2024. UV-NDVI for real-time crop health monitoring in vertical farms. Smart Agricultural Technology 8: 100462. https://doi.org/10.1016/j.atech.2024.100462