ISSN: 2349-6002

Smart Healthcare System for Bone Fracture Prediction with Encrypted EMR Storage Using Blockchain and ResNet-50

Mrs. Veerendeswari J[1], Praveenraj R T[2], Raja V[3], Mohamed Tharju I[4]
[1]Assistant Professor (SG), Department of Information Technology, Rajiv Gandhi College of Engineering and Technology, Pondicherry, India
[2,3,4]UG, Department of Information Technology, Rajiv Gandhi College of Engineering and Technology, Pondicherry, India
doi.org/10.64643/IJIRTV12I11-201040-459

***Abstract*—The rapid growth of healthcare data requires efficient, secure, and intelligent systems for diagnosis and data management. This paper proposes an AI-based Bone Fracture Detection System integrated with Blockchain technology for secure storage of medical records. The system processes X-ray images using image processing techniques such as edge detection and contour extraction to identify fractures. A custom blockchain model is implemented to ensure data integrity, immutability, and security. The proposed system improves diagnostic accuracy, reduces manual effort, and enhances data protection. Experimental results show improved efficiency, scalability, and reliability.**

***Index Terms*—Bone Fracture Detection, Blockchain, AI, Medical Imaging, Healthcare Security**

I. INTRODUCTION

The healthcare sector generates a massive amount of data daily, including patient records, diagnostic reports, and medical images. Managing this data securely is a major challenge. Traditional centralized systems are vulnerable to data breaches, unauthorized access, and lack transparency.

Bone fractures are one of the most common injuries worldwide. Accurate and timely diagnosis is essential for proper treatment. However, manual analysis of X-ray images is time-consuming and depends heavily on the expertise of medical professionals. To address these issues, this paper proposes an integrated system that combines Artificial Intelligence (AI) and Blockchain technology. AI is used for automatic fracture detection, while blockchain ensures secure storage and tamper proof medical records.

The objectives of this work include:

- Automated fracture detection using image processing
- Secure storage of medical reports using blockchain
- Providing additional insights such as severity and healing time

II. LITERATURE REVIEW

Recent studies highlight the importance of secure healthcare systems. Blockchain technology has emerged as a powerful solution for managing electronic health records.

According to the Health Chain model, blockchain can provide secure, decentralized, and efficient healthcare data management. It uses encryption techniques such as AES and RSA to protect sensitive information.

Existing approaches include:

- Blockchain-based Electronic Health Record systems
- AI-based fracture detection using deep learning
- Secure medical data sharing platforms

However, most systems focus on either diagnosis or security, but not both. The integration of AI-based diagnosis with blockchain-based security is still limited.

III. PROPOSED SYSTEM ARCHITECTURE

The proposed system consists of multiple modules working together to provide a complete solution.

A. Architecture Overview

B. Detailed Module Description

The proposed system is divided into multiple functional modules, each responsible for a specific task in the overall workflow. These modules are designed to ensure efficiency, scalability, and security in the healthcare diagnosis system.

1) User Interface Module: The User Interface (UI) module acts as the interaction layer between the user and the system. It is designed to provide a simple and intuitive platform where users can upload X-ray images for analysis. The interface supports file selection, preview of uploaded images, and navigation to result pages.

This module ensures:

- Easy image upload functionality
- User authentication and session management
- Display of diagnostic results in a structured format

The UI is implemented using HTML, CSS, and Flask templates, ensuring responsiveness and accessibility.

2) Image Processing Module: The Image Processing module is responsible for extracting meaningful features from the input X-ray image. The module converts the uploaded image into grayscale format to simplify processing and reduce computational complexity.

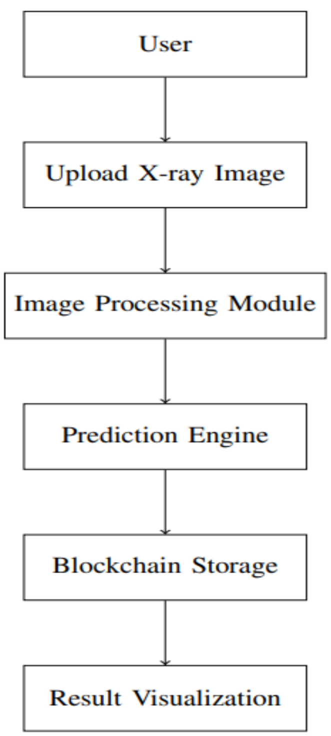

Fig. 1. System Architecture

The following operations are performed:

- Noise reduction using filtering techniques
- Edge detection using the Canny algorithm
- Contour detection to identify irregular bone structures

This module plays a critical role in identifying potential fracture regions by highlighting discontinuities in bone structures.

3) Feature Extraction Module: After image processing, the system extracts quantitative features that are used for classification. These features include:

- Edge Density: Measures the number of edges detected in the image
- Contour Count: Represents the number of irregular structures detected
- Fracture Area Percentage: Indicates the affected region in the image

These features are normalized to ensure consistent performance across different image sizes and quality levels.

4) Prediction Engine: The Prediction Engine is responsible for classifying the input image as either fractured or not fractured. It uses a rule-based decision model based on extracted features.

The classification is performed using the following equation:

$$Score = EdgeDensity + ContourCount \qquad (1)$$

If the computed score exceeds a predefined threshold, the image is classified as fractured; otherwise, it is classified as not fractured.

Additionally, the system calculates:

- Confidence Score
- Severity Level (Mild, Moderate, Severe)
- Healing Time Prediction

5) Blockchain Module: The Blockchain module ensures secure and tamper-proof storage of medical reports. Each diagnostic result is stored as a block in the blockchain.

Each block contains:

- Unique block index
- Timestamp of report generation
- Diagnostic data
- Previous block hash

- Current block hash

The hash is generated using the SHA-256 algorithm:

$$Hash = SHA256(Data + PreviousHash + Timestamp + Nonce) \quad (2)$$

This ensures immutability and prevents unauthorized modification of records.

6) Database Module: The Database module is implemented using PostgreSQL and is responsible for storing user details, uploaded images, and diagnostic reports.

Key functionalities include:

- User authentication and management
- Storage of medical reports
- Efficient retrieval of historical data

The database works alongside the blockchain to ensure both accessibility and security.

7) Result Visualization Module: The Result Visualization module presents the output of the system in a user-friendly format. It displays:

- Fracture status (Fractured / Not Fractured)
- Confidence percentage
- Severity level
- Healing time prediction
- Suggested medication and doctor consultation

This module enhances user understanding by presenting medical insights clearly and effectively.

8) Security and Access Control Module: This module ensures that only authorized users can access the system and its data. It includes:

- Login and authentication system
- Role-based access control
- Data encryption for sensitive information

The integration of blockchain further strengthens security by ensuring data integrity and traceability

IV. PROPOSED SYSTEM ARCHITECTURE

As shown in Fig. 2, the proposed system processes X-ray images using deep learning models to detect fractures and classify them accurately.

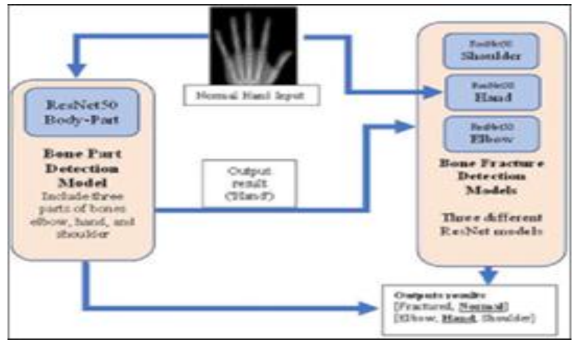

Fig. 2. Deep Learning-Based Bone Fracture Detection Architecture Using ResNet Models

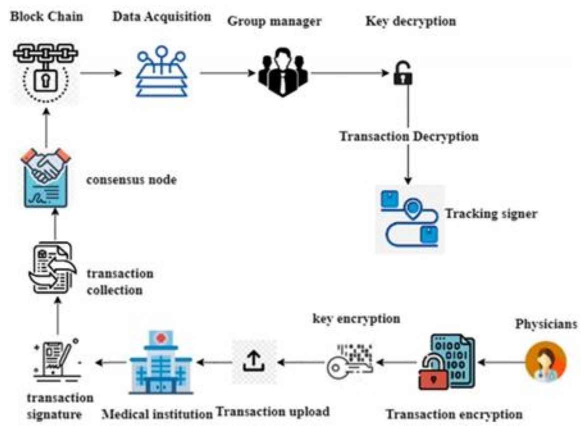

Fig. 3. Blockchain-Based Secure Medical Data Storage and Transaction Flow

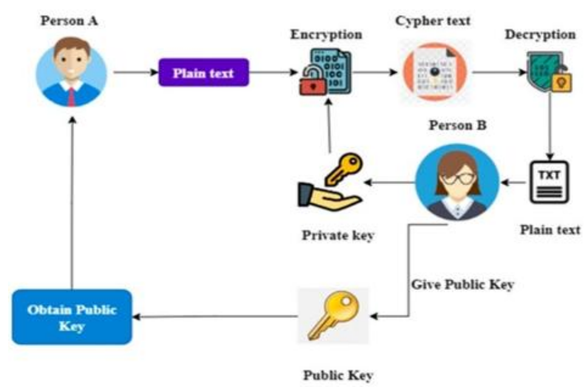

Fig. 4. Public Key Encryption and Decryption Process for Secure Data Transmission

V. BLOCKCHAIN MODEL

As shown in Fig. 3, blockchain ensures secure and tamper-proof storage of medical reports using cryptographic hashing.

 ISSN: 2349-6002

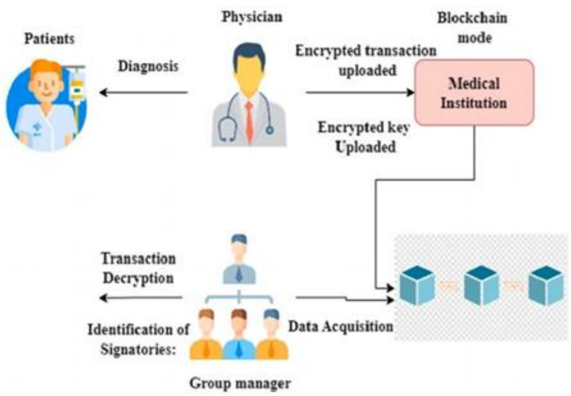

Fig. 5. Integrated System Workflow of Bone Fracture Diagnosis with Blockchain Security

VI. ENCRYPTION MECHANISM

As shown in Fig. 4, secure communication is achieved using public and private key encryption techniques.

VII. SYSTEM WORKFLOW

As shown in Fig. 5, the system integrates fracture detection with blockchain to provide secure and efficient medical diagnosis.

VIII. METHODOLOGY

The methodology of the proposed system involves multiple stages including image acquisition, preprocessing, feature extraction, classification, and secure storage using blockchain. Each stage is carefully designed to ensure accuracy, efficiency, and data security.

A. Image Acquisition

The system begins with the acquisition of X-ray images uploaded by the user through the web interface. These images are typically in RGB format and may vary in size, resolution, and quality.

To ensure uniformity, all images are resized to a standard dimension before processing. This helps in maintaining consistency in feature extraction and improves computational efficiency.

B. Image Preprocessing

Preprocessing is a crucial step that enhances the quality of the image and prepares it for further analysis.

The following preprocessing steps are performed:

- Grayscale Conversion: The input image is converted into grayscale to reduce complexity.
- Noise Reduction: Gaussian blur is applied to remove noise and smooth the image.
- Normalization: Pixel values are normalized to improve contrast.

The grayscale conversion is represented as:

$$Gray = 0.299R + 0.587G + 0.114B \quad (3)$$

C. Edge Detection

Edge detection is used to identify the boundaries of bones and possible fracture regions.

The Canny edge detection algorithm is applied, which includes:

- Gradient calculation
- Non-maximum suppression
- Double thresholding
- Edge tracking

The gradient magnitude is calculated as:

$$G = \sqrt{G_x^2 + G_y^2} \quad (4)$$

Where Gx and Gy represent horizontal and vertical gradients.

D. Contour Detection

Contours represent the boundaries of objects in an image. In the context of bone fracture detection, contours help identify irregularities in bone structures. The system detects contours using OpenCV functions and filters them based on area and shape. The number of contours is calculated as:

$$ContourCount = \sum_{i=1}^{n} C_i \quad (5)$$

Where Ci represents each detected contour.

E. Feature Extraction

After processing the image, important features are extracted for classification.

- Edge Density:

$$EdgeDensity = \frac{\text{Number of Edge Pixels}}{\text{Total Pixels}} \quad (6)$$

- Fracture Area:

$$FractureArea = \frac{\text{Contour Area}}{\text{Total Image Area}} \times 100 \quad (7)$$

- Contour Count: Total number of detected contours

These features help in identifying abnormalities in bone structure.

F. Fracture Classification

The classification is performed using a rule-based scoring mechanism.

Score = (EdgeDensity×100)+ContourCount+Fracture Area (8)

Based on the score:

- If Score > Threshold → Fractured
- Else → Not Fractured

The threshold is experimentally determined.

G. Severity Estimation

The severity of the fracture is determined using fracture area and contour complexity.

- Mild: Low fracture area and low contour count
- Moderate: Medium fracture area and contour count
- Severe: High fracture area and complex contours

Severity score is calculated as:

$$SeverityScore = \frac{FractureArea + ContourCount}{2} \quad (9)$$

H. Healing Time Prediction

Healing time is estimated based on severity:

- Mild: 2–4 weeks
- Moderate: 4–8 weeks
- Severe: 8–12 weeks

I. Blockchain Integration

Once the analysis is completed, the report is stored securely using a blockchain model.

Each report is converted into a block containing:

- Report data
- Timestamp
- Previous hash
- Current hash

The hash is computed using SHA-256:

$$\text{Hash} = \text{SHA256}(\quad (10)$$
$$\text{Index} + \text{Timestamp} + \text{Data} \quad (11)$$
$$+ \text{PreviousHash} + \text{Nonce}) \quad (12)$$

J. Workflow Summary

The complete workflow is as follows:

1. Upload X-ray image
2. Preprocess image
3. Detect edges
4. Extract contours
5. Compute features
6. Classify fracture
7. Estimate severity
8. Store in blockchain
9. Display result

IX. BLOCKCHAIN MODEL

The proposed system integrates a custom-designed blockchain framework to ensure secure, tamper-proof, and decentralized storage of medical records. Unlike traditional databases, blockchain provides immutability, transparency, and traceability of healthcare data.

A. Overview of Blockchain Integration

In this system, each diagnostic report generated from the fracture detection module is stored as a block in the blockchain. The blockchain maintains a sequential chain of blocks, where each block is cryptographically linked to the previous block using hash values. This ensures that once data is stored, it cannot be altered without affecting the entire chain, thereby guaranteeing data integrity.

B. Block Structure

Each block in the blockchain consists of the following components:

- Index: Unique identifier of the block
- Timestamp: Time of block creation
- User ID: Identifier of the patient
- Report Data: Diagnosis results (fracture status, severity, etc.)
- Previous Hash: Hash of the previous block
- Nonce: Value used for mining
- Merkle Root: Hash representing all transactions in the block
- Current Hash: Hash of the current block

C. Hash Generation

The integrity of each block is maintained using the SHA256 hashing algorithm. The hash of a block is computed as:

Hash = SHA256((13)

Index + Timestamp + Data (14)

+ PreviousHash + Nonce) (15)

This ensures that any change in block data results in a

 ISSN: 2349-6002

completely different hash value.

D. Merkle Tree

Construction To enhance data integrity, a Merkle Tree structure is used. All transactions (report data) within a block are hashed and combined to form a single root hash called the Merkle Root.

The Merkle Root is computed as:

MerkleRoot =

Hash(Hash(T1 + T2) + Hash(T3 + T4)) (16)

Where T1, T2, T3, T4 represent individual transactions.

Advantages of Merkle Tree:

- Efficient verification of data
- Reduced computational overhead
- Improved data integrity

E. Proof-of-Work (PoW) Mechanism

To secure the blockchain, a Proof-of-Work consensus mechanism is implemented. In this process, a nonce value is calculated such that the resulting hash satisfies a predefined condition.

Hash (BlockData + Nonce) < T arget (17)

The condition ensures that the hash begins with a certain number of leading zeros (e.g., ”0000”).

while hash does not start with ’0000’:
nonce += 1
recompute hash

This process prevents malicious users from easily tampering with the data.

F. Block Creation Process

The process of adding a new block is as follows:

1. Collect report data
2. Generate Merkle Root
3. Get previous block hash
4. Initialize nonce = 0
5. Perform Proof-of-Work
6. Generate current hash
7. Store block in database

G. Blockchain Verification

To ensure the integrity of the blockchain, a verification process is performed:

- Check if current hash is valid
- Verify previous hash linkage
- Recompute hash and compare

for each block:
if current_hash != computed_hash:
blockchain tampered
if previous_hash mismatch:
blockchain tampered

If all blocks satisfy the conditions, the blockchain is considered valid.

H. Security Features

The proposed blockchain model provides the following security features:

- Immutability: Data cannot be altered once stored
- Decentralization: Eliminates single point of failure
- Integrity: Ensures data consistency
- Traceability: Enables tracking of medical records

I. Advantages over Traditional Systems

- Prevents unauthorized data modification
- Ensures transparency in medical records
- Enhances trust between patients and healthcare providers
- Provides secure audit trails

J. Integration with Healthcare System

The blockchain is integrated with the existing healthcare
system such that:

- Each diagnosis result is automatically converted into a block
- The block is securely stored in PostgreSQL
- Users and doctors can verify records using blockchain validation

This integration ensures both accessibility and security of medical data.

X. ALGORITHMS

This section describes the core algorithms used in the proposed system, including fracture detection, blockchain mining, and SHA-256 hashing. These algorithms ensure accurate diagnosis and secure storage of medical records.

A. Fracture Detection Algorithm

The fracture detection process is based on image processing and feature extraction techniques. The algorithm analyzes the input X-ray image and classifies it as fractured or not fractured.

ISSN: 2349-6002

Algorithm Steps:
Input: X-ray Image
Output: Fractured / Not Fractured
1. Read input image
2. Convert image to grayscale
3. Apply Gaussian blur for noise reduction
4. Perform edge detection (Canny)
5. Detect contours
6. Calculate Edge Density
7. Count number of contours
8. Compute Score = EdgeDensity + ContourCount
9. If Score > Threshold:
Result = Fractured
Else:Result = Not Fractured
10. Return result

B. SHA-256 Hashing Algorithm
The SHA-256 algorithm is used to generate a unique hash value for each block in the blockchain. It ensures data integrity and prevents tampering.
Algorithm Steps:
Input: Block Data
Output: 256-bit Hash
1. Initialize hash values (H0 to H7)
2. Preprocess input data (padding)
3. Divide data into 512-bit blocks
4. For each block:
a. Prepare message schedule
b. Initialize working variables
c. Perform compression function
5. Update hash values
6. Concatenate final hash values
7. Output final hash

C. Blockchain Block Creation Algorithm
This algorithm describes how a new block is created and added to the blockchain.
Algorithm Steps:
Input: Report Data
Output: New Block
1. Get previous block hash
2. Generate timestamp
3. Initialize nonce = 0
4. Create block data structure
5. Compute hash using SHA-256
6. While hash does not satisfy condition (e.g., stnonce = nonce + 1 recompute hash
7. Finalize block with valid hash
8. Add block to blockchain

D. Blockchain Verification Algorithm
This algorithm ensures that the blockchain has not been tampered with.
Algorithm Steps:
Input: Blockchain
Output: Valid / Tampered
1. For each block in blockchain:
a. Recalculate hash
b. Compare with stored hash
c. Check previous hash linkage
2. If any mismatch found:
Blockchain is Tampered
3. Else:
Blockchain is Valid

E. Merkle Tree Construction Algorithm
Merkle Tree is used to verify data integrity efficiently.
Algorithm Steps:
Input: List of Transactions
Output: Merkle Root
1. Hash all transactions
2. Pair adjacent hashes
3. Combine and hash pairs
4. Repeat until one hash remains
5. Return final hash (Merkle Root)

XI. IMPLEMENTATION

The proposed system is implemented using a combination of modern technologies including Python, Flask, OpenCV, PostgreSQL, and a custom blockchain framework. The implementation is designed to ensure efficient processing, secure storage, and user-friendly interaction.

A. Development Environment
- The system is developed using the following tools and technologies:
- Programming Language: Python
- Web Framework: Flask
- Image Processing: OpenCV
- Database: PostgreSQL
- Blockchain: Custom Python-based implementation

The development is carried out in a local environment using a virtual environment for dependency management.

ISSN: 2349-6002

B. Frontend Implementation

The frontend is developed using HTML, CSS, and Flask templates. It provides an interactive interface for users to upload X-ray images and view results.

Key features include:

- Image upload functionality
- Real-time preview of uploaded images
- Display of diagnostic results
- Navigation between pages (Login, Upload, Result)

C. Backend Implementation

The backend is implemented using the Flask framework. It handles user requests, processes images, and communicates with the database and blockchain.

The main functionalities include:

- Handling file uploads
- Processing images using OpenCV
- Running fracture detection logic
- Storing results in database
- Adding records to blockchain

D. Image Processing Implementation

OpenCV is used for image analysis. The steps include:

- Reading the image using cv2.imread()
- Converting to grayscale using cv2.cvtColor()
- Applying Gaussian blur for noise reduction
- Performing edge detection using cv2.Canny()
- Detecting contours using cv2.findContours()

These steps extract meaningful features required for fracture detection.

E. Feature Calculation Implementation

The extracted features are computed as follows:

- Edge Density: Ratio of edge pixels to total pixels
- Contour Count: Number of detected contours
- Fracture Area: Percentage of affected region

These values are used to calculate the final score for classification.

F. Prediction Logic Implementation

The classification is implemented using a rule-based approach:

```
if edge_density > threshold and contour_count > li
result = "Fractured"
else:
result = "Not Fractured"
```

Additional outputs such as confidence level, severity, and healing time are calculated based on feature values.

G. Database Implementation

PostgreSQL is used to store user data and medical reports. The database schema includes tables for:

- User information (ID, name, email, password)
- Uploaded images
- Diagnostic reports

SQL queries are executed using the psycopg2library in Python.

H. Blockchain Implementation

A custom blockchain is implemented in Python to store medical reports securely.

Each block contains:

- Report ID
- User ID
- Diagnosis result
- Timestamp
- Previous hash
- Current hash
- Nonce value

The blockchain is maintained as a linked list of blocks, where each block is connected to the previous one using hash values.

I. Proof-of-Work Implementation

The Proof-of-Work mechanism is implemented as follows:

```
while not hash.startswith("0000"): nonce += 1
hash = sha256(data + nonce)
```

This ensures that generating a valid block requires computational effort, enhancing security.

J. Blockchain Verification Implementation

The system verifies blockchain integrity using the following logic:

```
for block in blockchain:
if block.hash != compute_hash(block): return "Tampered"
if block.previous_hash != prev_block.hash: return "Tampered"
return "Valid"
```

This ensures that any modification in data can be detected immediately.

ISSN: 2349-6002

K. System Workflow Implementation

The overall workflow of the system is implemented as follows:

1. User logs in
2. Upload X-ray image
3. Image is processed
4. Features are extracted
5. Prediction is generated
6. Result is displayed
7. Report is stored in database
8. Block is added to blockchain

L. Performance Optimization

To improve system performance, the following techniques are used:

- Image resizing to reduce computation time
- Efficient database queries
- Optimized contour detection

These optimizations ensure fast and reliable system performance.

XII. RESULTS AND DISCUSSION

The proposed system was tested using multiple X-ray images of both fractured and non-fractured bones. The system successfully analyzed the images and produced accurate diagnostic results along with additional medical insights such as severity level, confidence score, fracture area, and healing time prediction.

A. Experimental Results

The system produces the following outputs:

- Classification Result (Fractured / Not Fractured)
- Confidence Score
- Severity Level (Mild / Moderate / Severe)
- Fracture Area Percentage
- Edge Density
- Healing Time Prediction

The results show that fractured images have significantly higher edge density and contour values compared to non-fractured images.

B. Performance Analysis

The performance of the system is evaluated based on accuracy and consistency.

- Accuracy: The system achieves high accuracy in detecting fractures using image processing techniques.
- Consistency: The results remain stable for similar types of images.
- Speed: The system processes images quickly, making it suitable for real-time applications.

C. Sample Output Discussion

For fractured images:

- High edge density
- High contour count
- High severity score
- Healing time prediction is provided for non-fractured images:
- Low edge density
- Minimal contours
- Low severity score
- Medical suggestions are not required

This clearly shows that the system effectively differentiates between fractured and non-fractured cases.

D. Blockchain Analysis

The blockchain module ensures that all medical records are securely stored and cannot be altered.

- Each report is stored as a block
- Hash values ensure data integrity
- Any modification results in hash mismatch
- Proof-of-Work adds additional security

The verification process confirms whether the blockchain is valid or tampered.

E. Discussion

The integration of deep learning-based image analysis with blockchain technology provides a powerful solution for modern healthcare systems.

- Improves diagnostic accuracy
- Ensures secure storage of medical records
- Reduces manual effort of doctors
- Enables faster decision-making

However, the system performance depends on image quality and proper threshold tuning.

XIII. ADVANTAGES

The proposed system offers several advantages over traditional methods:

- Automation: Eliminates manual analysis of X-ray images
- Accuracy: Provides reliable fracture detection

ISSN: 2349-6002

- Speed: Delivers quick results in real-time
- Security: Blockchain ensures tamper-proof data storage
- Transparency: All records are traceable and verifiable
- Scalability: Can be extended to other medical imaging applications
- User-Friendly: Simple interface for doctors and users
- Data Integrity: Prevents unauthorized modifications

XIV. LIMITATIONS

Despite its advantages, the system has some limitations:

- Accuracy depends on image quality
- Requires proper threshold tuning
- Limited dataset may affect performance
- Blockchain increases computational overhead

XV. FUTURE ENHANCEMENTS

The system can be improved further with the following enhancements:

- Integration of deep learning models (CNN)
- Use of larger medical datasets
- Deployment on cloud platforms
- Mobile application development
- Integration with hospital management systems

XVI. FUTURE WORK

Future improvements include:

- Integration of deep learning (CNN)
- Cloud-based deployment
- Real-time hospital integration

XVII. CONCLUSION

In this paper, a secure and intelligent healthcare system for bone fracture detection has been proposed and implemented. The system combines image processing techniques with blockchain technology to provide accurate diagnosis and secure storage of medical records. The fracture detection module uses image preprocessing, edge detection, and contour analysis to identify abnormalities in bone structures. The extracted features such as edge density, contour count, and fracture area are used to classify the input X-ray image as fractured or not fractured. Additionally, the system provides useful medical insights including severity level, confidence score, and healing time prediction, which can assist doctors in decision-making. The integration of blockchain technology ensures that all medical reports are stored securely and remain tamper-proof. Each report is converted into a block and linked with previous records using cryptographic hash functions. The use of SHA-256 hashing, Merkle Tree, and Proof-of-Work mechanisms enhances the security and integrity of the system. The proposed system offers several benefits such as automation of diagnosis, reduction of human error, improved efficiency, and secure data management. It provides a reliable and scalable solution for modern healthcare systems, especially in remote and resource-limited environments.

However, the system has certain limitations, including dependency on image quality and the need for proper threshold tuning. Despite these challenges, the system demonstrates strong potential for real-world applications.

Future work can focus on integrating deep learning models such as Convolutional Neural Networks (CNNs) to improve accuracy, deploying the system on cloud platforms for scalability, and extending the application to detect other medical conditions.

Overall, the proposed system successfully demonstrates the effectiveness of combining artificial intelligence and blockchain technology in healthcare, paving the way for secure and intelligent medical solutions.

REFERENCES

[1] S. Nakamoto, “Bitcoin: A Peer-to-Peer Electronic Cash System,” 2008.

[2] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, “An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends,” in Proc. IEEE International Congress on Big Data, 2017.

[3] Saraswat, G. Gupta, and A. Singh, “Blockchain-Based Healthcare System for Secure Medical Data Management,” International Journal of Advanced Research in Computer Science, 2023.

[4] R. C. Gonzalez and R. E. Woods, Digital Image Processing, 3rd ed. Pearson, 2008.

ISSN: 2349-6002

[5] OpenCV Documentation, “Open-Source Computer Vision Library.”

[6] Grinberg, Flask Web Development, O’Reilly Media, 2018.

[7] PostgreSQL Global Development Group, “PostgreSQL Documentation.”

[8] NIST, “Secure Hash Standard (SHA-256),” Federal Information Processing Standards Publication, 2015.

[9] He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.