SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V13I7P110

Volume 13 Issue 7, 140-154, July 2026

Original Article

A Cryptographic Proof Framework for Tamper-Evident Pneumonia Diagnosis on Blockchain

B. Murugeshwari[1*], Kayalvizhi S[1], Daya Florance D[1], Saranya S[2]

[1]*Department of Computer Science and Engineering, Velammal Engineering College, Chennai, Tamil Nadu, India.*
[2]*Department of Artificial Intelligence and Data Science, S.A. Engineering College, Chennai, Tamil Nadu, India.*

[*]*Corresponding Author : murugeshwariniyansree@gmail.com*

Received: 13 April 2026 Revised: 03 June 2026 Accepted: 19 June 2026 Published: 29 July 2026

Abstract *- To deploy deep learning-based diagnostic systems in a clinical setting, they need to have not only a high degree of predictive accuracy, but also an unbroken cryptographic chain to prove that the model parameters have not been changed from one inference to the next. This incompatibility arises because softmax, GELU activation, and layer normalization involve transcendental operations to produce the final output. Transcendental operations cannot be represented over the prime finite fields that are necessary when utilizing Rank-1 Constraint Systems (R1CS). The work here provides a mathematically sound approach to resolving the incompatibility by providing three formally defined polynomial approximations: a degree-3 Taylor series softmax approximation ($\varepsilon \leq 2.47E\text{-}03$ per weight); a degree-5 composite polynomial GELU ($\varepsilon \leq 1.80E\text{-}03$); and a squared witness hint reformulation of layer normalization that results in exact constraint satisfaction. The accumulating polynomial approximation errors across 12 transformer encoder blocks have a total approximation error defined as (ε total ≤ 0.0512) and results in a loss of less than one-fifth of a percent in accuracy (94.1% vs. 94.3%) on the RSNA Pneumonia Detection benchmark. The entire ViT-B/16 inference model was compiled into an R1CS form of approximately 2.3×10^8 R1CS constraints and instantiated as a Groth 16 zk-SNARK. The integrity of the model is confirmed by hashing the parameters using SHA-256 onto an immutable Polygon zkEVM smart contract, allowing for on-chain verification of inference without revealing any of the proprietary model weights. Additionally, this system produces an area under the receiver operating characteristic curve (AUC-ROC) of 0.961, a mean latency for proof generation of 2.84 seconds, an end-to-end verification time of 5.07 seconds, and an average cost for on-chain verification of 0.012ETH, all within the operational constraints of typical radiology workflows.*

Keywords *- Cryptographic chain, Polynomial approximation, Pneumonia diagnosis.*

1. Introduction

Recently, there have been numerous deep learning model deployments to help automate disease diagnosis and decision-making in many clinical imaging workflows. As these tools have increased in use, their security, reliability, and trust are major concerns. Their use is also becoming more prevalent in high-stakes situations like pneumonia detection from chest X-ray. Operating models can be subject to a number of threats, including unauthorized changes to parameters, backdoor injection, cloning, and tampering with their environment. Liu et al. [13] showed that off-the-shelf deep learning models are vulnerable to backdoor attacks through unsupervised domain adaptation, while Lyu et al. [14] demonstrated similar vulnerabilities in clinical language models, confirming that medical AI systems face active adversarial threats. As healthcare providers increasingly update their infrastructure by moving towards the cloud or employing edge computing solutions, the need to protect against threats has become more prominent, as it cannot be assumed that central authorities will have complete administrative oversight over the deployment of each model [1]. Initially, researchers sought to identify ways to detect changes to the models by looking at tampering with the protection mechanisms built into the model; later, researchers would expand these efforts and create provenance monitoring systems to monitor the history of how the model has changed during its life cycle, where it is stored, and who has used it [2, 15]. Longpre et al. [12] further highlighted that data authenticity, consent, and provenance for AI systems remain fundamentally broken, arguing that cryptographic mechanisms are essential to restore trust in the AI pipeline. Meanwhile, trust technology continued to improve; these newly developed decentralized verification systems have shown great promise in supporting the application of safe AI in the healthcare sector by providing new ways for end-users to verify the integrity of an AI system prior to its use [3]. Integrating data integrity and auditable systems via cryptographic primitives with current workflows can also alleviate many of the additional concerns associated with centralized systems and provide a basis for additional verification/support. The fact that model designs became more

complex, particularly after attention systems were invented, caused prompted researchers to seek stronger assurances that model parameters had not been tampered with [4]. Other methods attempted to determine more secure, transparent, or resistant to external changes [5], through inherent design principles, output-level behavioral validation, or consistency checks performed without exposing internal model mechanics [6]. Nevertheless, many proposed solutions remained incomplete, addressing only isolated aspects — such as ownership verification, adversarial poisoning prevention, or post-hoc tamper detection — without providing end-to-end integrity assurance across the full inference pipeline [7].

Research has examined the convergence of medical AI for deep learning, blockchain, and cryptographic proof. Ruchika Bhuria et al. [30] demonstrated that Convolutional Neural Networks (CNNs) can be effectively used for automated chest X-ray classification, providing baseline accuracy standards for future studies to achieve and attach verifiability metrics to. Dosovitskiy et al. [31] introduced the Vision Transformer (ViT) architecture, which showed that pure attention-based neural networks perform at least as well as, if not better than, CNNs in image classification, making it possible to use the ViT in medical imaging. Chen et al. [32] also used transformers to analyze medical images and found that these methods could generalize well across the range of radiological databases, including those that analyzed chest pathology.

In the cryptographic proof world, Ben-Sasson et al. [34] proposed zk-SNARKS as the first practically usable form of zero-knowledge proofs, with the later Groth16 implementation [24], offering the briefness and efficiency of verification necessary for using blockchain for proof verification. Weng et al. [25] examined the problem of being able to compile neural network computation into arithmetic circuits that were compatible with zk-SNARK proof generation and identified the problem of the transcendental function that this work directly addresses. Kang et al. [26] proposed a framework for ZKML for producing verifiable machine learning inference from zero-knowledge proofs, but their approach only examined verifiable machine learning using the simpler Multi-Layer Perceptron (MLP) architectures, as opposed to the more complex transformer networks that are used in medical imaging.

Research about health information management through the use of blockchain technology has received a considerable amount of attention over the last several years. Kuo et al. [21] reviewed how various blockchain applications could apply towards managing health information and how immutability and audit trails would help provide significant benefits to the clinical aspects of health care services. Additionally, Azaria et al. [22] proposed a blockchain solution called MedRec as a means for managing medical records. They showed how the MedRec framework revealed discrepancies between the need to have access to data and respecting the confidentiality of patient information. More recently, Nguyen et al. [23] investigated Ethereum-based smart contracts for ensuring provenance and integrity of clinical decision support systems, noting that on-chain storage of model parameters was prohibitively expensive, motivating off-chain hashing approaches such as the one adopted in the present work.

Privacy-preserving computation in healthcare AI has attracted significant attention. Sun et al. [29] demonstrated that federated learning architectures are susceptible to data poisoning attacks that can silently degrade model integrity across participating institutions, underscoring the need for cryptographic verification of inference outputs in distributed healthcare deployments. Collectively, these studies establish the imperative for cryptographically sound, privacy-preserving inference verification but stop short of addressing the fundamental arithmetic circuit incompatibility of transformer architectures that the present work resolves.

Today, a greater number of individuals are aware that it is not sufficient to check deep learning systems by mere guesswork or weak checks [8]. In the case of hospitals, it requires good digital evidence - something, such as encryption, which demonstrates that every outcome is obtained using an authentic, unaltered, and verifiable copy of the model. Emerging solutions increasingly rely on mathematically grounded verification mechanisms and cryptographic data fingerprints, in addition to the tools that verify identities without putting the patient data at risk [9]. These developments converge on a central requirement: contemporary healthcare AI demands cryptographically enforceable trust, permanent audit records, and continuous monitoring to support the clinical decisions that practitioners rely on [10].

Despite these advances, a critical and unresolved gap persists in the literature: no prior work has formally characterized the arithmetic circuit incompatibility of Vision Transformer inference operations — specifically softmax normalization, GELU activation, and layer normalization — within the constraints of zk-SNARK proof systems for medical AI deployment. Existing blockchain-integrated diagnostic frameworks rely on hash-based provenance tracking or federated trust mechanisms, none of which provide cryptographic proof that a specific, registered model version produced a given prediction.

Furthermore, the transcendental and irrational functions inherent to transformer architectures have no exact representation over the prime finite fields required by Rank-1 Constraint Systems (R1CS), a fundamental incompatibility that prior Zero-Knowledge Machine Learning (ZKML) works have addressed only for simpler architectures such as multi-layer perceptrons, leaving transformer-based medical inference verifiably unprotected.

The principal contributions of this work are as follows:

- Arithmetic Circuit Approximation for Transformer Inference. The formalization, for the first time, polynomial approximations for the three non-arithmetic operations in ViT inference — softmax (degree-3 Taylor), GELU (degree-5 composite), and layer normalization (squared witness-hint) — that are compatible with Rank-1 Constraint System (R1CS) arithmetic circuits required by Groth16 zk-SNARKs. Proof shows that per-operation error bounds and derive an accumulated error bound of $\varepsilon\_total \leq 0.0512$ across 12 encoder blocks (Section 2.4.1, Equation 12).
- Proposition on Circuit Complexity. This research characterizes the full ViT-B/16 inference circuit as requiring $\sim 2.3 \times 10^8$ arithmetic constraints, and shows that the SHA-256 hash sub-circuit contributes less than 0.01% of total circuit size, confirming that the integrity verification overhead is negligible relative to inference computation (Proposition 1, Section 2.4.1).
- Empirically Validated Approximation Fidelity. This research demonstrates that the circuit-approximated ViT achieves 94.1% accuracy on the RSNA Pneumonia Detection dataset — a degradation of only 0.2 percentage points relative to the exact floating-point model — confirming that the approximations preserve clinical utility while enabling cryptographic verification.
- End-to-End Verifiable Medical AI Deployment. This implementation integrates the circuit-based proof system with a Polygon zkEVM smart contract for immutable on-chain inference logging and tamper detection, achieving a mean end-to-end verification latency of 5.07 seconds — compatible with standard radiology reporting workflows of 15–30 minutes.

2. Related Work

2.1. Deep Learning for Medical Image Diagnosis

Convolutional Neural Networks have been extensively applied to automated chest X-ray classification, establishing baseline diagnostic accuracy standards for pneumonia detection. Ruchika Bhuria et al. [30] showed that CNN-based classifiers trained with large radiographic image data achieve clinically relevant levels of sensitivity, the convolutional kernel architectures are built such that they can only detect and relate to local areas (within the image). Consequently, CNN-based classifiers were limited in their ability to relate patterns of disease that occur together over longer-range spatial locations. Conversely, Dosovitskiy et al.'s [31] creation of the Vision Transformer (ViT) architecture demonstrated that an attention-based architecture would achieve performance on the benchmarks used for image classification, including medical imaging.

2.2. Model Integrity and Tamper Detection

The integrity of deployed deep learning models has received growing attention as clinical AI adoption increases. Hoque [4] proposed a watermarking-based mechanism for deep learning model integrity checking, embedding proprietary signatures within model weights. Mu et al. [5] introduced Model DNA, a provenance-tracking scheme that encodes model identity through structural fingerprinting. Schwab and Biswas [6] explored invertible neural networks as a means of inference integrity verification through output consistency constraints.

Pautov et al. [7] proposed probabilistically robust watermarking under adversarial perturbations. Chien et al. [8] extended watermarking to multimedia security contexts, while Rajput and Kumar [9] investigated multi-view data embedding for robust neural network watermarking. Ben Jabra and Halimi [10] reviewed deep learning-based watermarking trends, identifying scalability and determinism as persistent limitations.

He et al. [11] examined black-box integrity verification frameworks for deployed models. Collectively, these approaches are probabilistic and post-hoc in nature, offering no cryptographic guarantee that a specific model version produced a given inference — a limitation the present work directly addresses.

2.3. Blockchain for Healthcare AI

Blockchain technology has been widely explored as a trust infrastructure for healthcare data and AI governance. Kuo et al. [21] conducted a comprehensive review of blockchain applications in health information management, highlighting immutability and audit trails as key benefits.

Azaria et al. [22] proposed MedRec, a blockchain-based medical records management system demonstrating the tension between data accessibility and patient privacy. Nguyen et al. [23] investigated Ethereum-based smart contracts for ensuring provenance and integrity of clinical decision support systems, noting that on-chain storage of model parameters is prohibitively expensive, motivating the off-chain hashing approach adopted in the present work.

Ali et al. [1] have shown that, when combined with hash-based integrity mechanisms, deep learning can enhance both security and scalability in a healthcare system built upon blockchain; Duggineni et al. [2] proposed the use of a hybrid deep learning and blockchain framework for tracking the provenance of medical data via the use of deep learning and/or blockchain technology, while simultaneously eliminating the need for cryptographic proof of correctness of inference; Sharma [18] designed a blockchain-based framework to provide secure data sharing between healthcare organizations via blockchain-based technology.

Despite the success of these studies in showing the viability of blockchain as a trust layer for AI in healthcare, they do not provide verifiable evidence that a particular registered model has generated a given diagnostic result.

2.4. Zero-Knowledge Proofs and Verifiable Machine Learning

Zero-Knowledge Proofs (ZKPs) were formalized by Ben-Sasson et al. [34] as a practically usable cryptographic primitive, with the Groth16 instantiation [24] offering the verification brevity and computational efficiency required for blockchain deployment. Zhang et al. [3] examined blockchain and zero-knowledge proof integration for secure data transactions in distributed computing, demonstrating feasibility at the system level. Weng et al. [25] investigated the compilation of neural network computations into arithmetic circuits compatible with zk-SNARK proof generation, identifying transcendental function incompatibility as a fundamental barrier. Kang et al. [26] proposed a Zero-Knowledge Machine Learning (ZKML) framework for verifiable inference, but restricted their evaluation to multi-layer perceptron architectures, leaving transformer-based models unaddressed. Ryffel et al. [27] demonstrated federated learning with cryptographic guarantees for medical data, while Kaissis et al. [28] reviewed end-to-end privacy-preserving deep learning for medical imaging, identifying zero-knowledge proofs as a promising but underexplored mechanism.

Sun et al. [29] demonstrated that federated learning systems are vulnerable to data poisoning attacks that can silently corrupt model behaviour across participating institutions, motivating the need for cryptographic inference verification. EdgeThemis [19], proposed by Yang, ensures model integrity for edge intelligence deployments but does not employ cryptographic proofs. Melo et al. [16] evaluated the performance characteristics of Hyperledger Fabric v2.5 as a blockchain substrate, establishing that permissioned ledgers can meet the throughput requirements of healthcare AI audit trails. Goundar [17] reviewed AI-blockchain integration for real-time cybersecurity, identifying inference-time verification as an open research gap. Singh [20] addressed LLM supply chain provenance through blockchain without inference-level verification. These works collectively establish the imperative for cryptographically sound inference verification but stop short of resolving the arithmetic circuit incompatibility of transformer architectures — the precise contribution of the present work.

2.5. Summary and Research Gap

Table 1 summarizes the positioning of existing works across four critical dimensions: model integrity verification, blockchain integration, cryptographic proof of inference, and transformer architecture support. No prior work simultaneously addresses all four dimensions. The present work bridges this gap by providing the first formally characterized polynomial approximation framework that resolves the arithmetic circuit incompatibility of ViT inference operations for zk-SNARK deployment, integrated with an immutable Polygon zkEVM blockchain infrastructure for end-to-end tamper-evident clinical AI.

3. Materials and Methods

3.1. System Architecture Overview

The three fundamental elements of the proposed framework include a pneumonia prediction model, which is based on a Vision Transformer, a cryptographic integrity verification layer, and a Polygon zkEVM blockchain infrastructure. By maintaining a consistent form of trust and assurance within the architecture of this system, all of the inferences made by the model will be cryptographically committed to an authenticated version of the model. This will also allow for end-to-end evidence of tampering throughout the entire diagnostic pipeline.

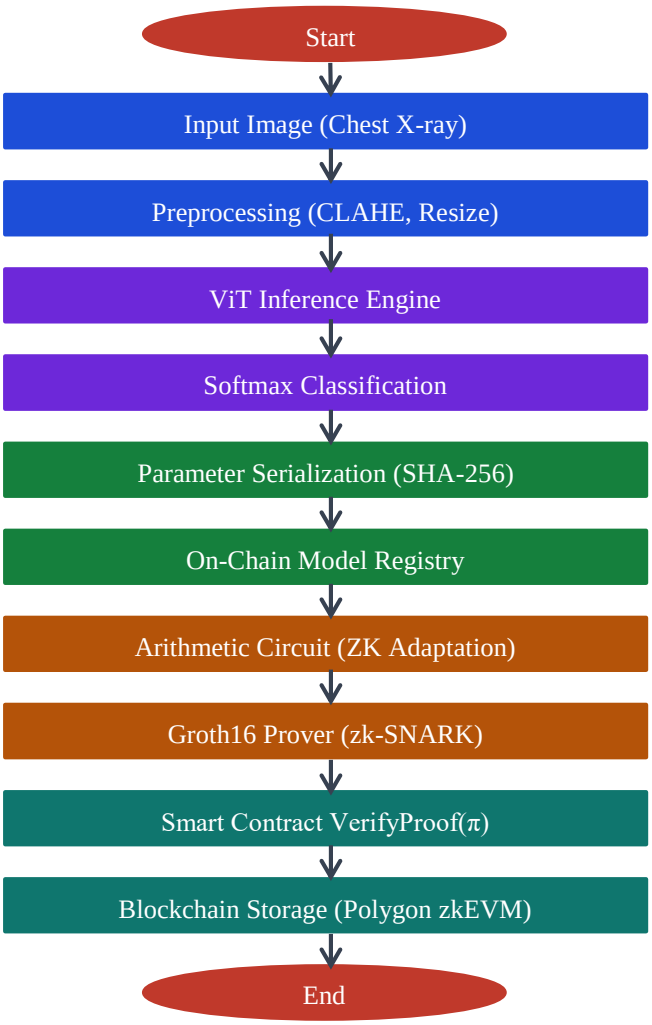

Fig. 1 System architecture diagram

The system operates using a two-phase workflow. The first phase involves the registration of the model, where all parameter values associated with a trained model are each hashed and anchored on-chain, creating a secure reference to each of the model's properties. The second phase of the workflow is known as the Inference Verification phase, which verifies the predictions produced by the model against those registered model identifiers using a zero-knowledge proof.

3.2. Vision Transformer for Pneumonia Detection

3.2.1. Image Preprocessing

The X-ray images collected through the input chest are then processed through a series of predefined preprocessing steps that ensure compatibility with the ViT architecture. X-ray images will be converted to grayscale, downsampled to 224x224, and Contrast Limited Adaptive Histogram Equalization (CLAHE) will be applied to enhance localized contrast differences that are critical in identifying pathologies.

The preprocessing function is defined as:

$$x_{\text{processed}} = \text{CLAHE!}\left(\text{Resize!}\left(\text{Grayscale}(x_{raw}),\ 224 \times 224\right)\right)$$

where x_raw represents the original DICOM or PNG format chest X-ray image.

3.2.2. Vision Transformer Architecture

The suggested pneumonia-detection model involves the Vision Transformer (ViT-Base) model, a fully attention-based model substituting the common convolutional feature-extracting model with global self-attention interactions. Following preprocessing, the 224x224 X-ray images are subdivided into fixed and non-overlapping 16x16 patches, yielding N = 196 patches in total. ViT uses patches as tokens in sequence, unlike CNNs, which enforce locality as determined by convolutional kernels, which is important to model long-range dependencies, relevant to absorb subtle radiographic signs of pneumonia, such as diffuse opacities or interstitial infiltrates.

Patch Embedding Layer

Given an image x ∈ ℝ^(224×224), the extracted patches form a sequence:

$$x_p = \left[x_p^1, x_p^2, \ldots, x_p^N\right], where x_p^i \in \mathbb{R}^{(16\times16)} \qquad (2)$$

Each flattened patch (size = 256) is projected into a D = 768-dimensional embedding space using a learnable linear projection:

$$z^0 = \left[x_class; E \cdot x_p^1; E \cdot x_p^2; \ldots; E \cdot x_p^N\right] + E_pos \quad (3)$$

Where x_class is a learnable classification token appended to the sequence, E ∈ ℝ^768×256 is the patch embedding matrix, and Epos provides positional encodings to retain spatial correspondence lost during patchification. This design enables the network to reason about spatial relationships, a critical requirement in medical imaging tasks.

Transformer Encoder with Multi-Head Self-Attention

The embedded sequence goes through L = 12 transformer encoder blocks, each comprising Multi-Head Self-Attention (MHSA) and position-wise feed-forward networks. MHSA allows the model to selectively emphasize the pathological characteristics of all spatial locations, including opacities or patterns of consolidation. Attention is calculated as:

$$Attention(Q, K, V) = softmax\left(QK^T/\sqrt{d_k}\right)V \qquad (4)$$

Where the query, key, and value matrices (Q, K, V) are obtained by linear projections of the input embeddings, and dk=D/H=64 when H= 12 attention heads. This multi-head model enables the model to learn the various radiographic patterns by using parallel attention subspaces, which are sensitive to clinically relevant features.

Feed-Forward Network and Residual Learning

In every encoder block, there is a two-layer Feed-Forward Network (FFN), layer normalization, and residual skip connections that allow the stable gradient flow and improved representational capacity:

$$FFN(z) = W2\,\sigma(W1z + b1) + b2, \qquad (5)$$

Where σ(·) is a GELU activation. The residual pathways help in providing robustness in the training process, and normalization provides scale-invariance processing of image features.

Classification Head

The diagnostic decision is obtained as the output of the corresponding classification token at the end of the last encoder layer:

$$y^{=} softmax(W \cdot zclassL + b), \qquad (6)$$

Where z_{class}^{L} is a representation of the combined data of all patches, which carries information on the global lung pathology patterns. The predicted probability y^∈[0,1] indicates the likelihood of pneumonia.

3.3. Cryptographic Model Registration

3.3.1. Model Parameter Hashing

In order to be certain that the clinical inference is provided by a validated deep-learning model, the proposed framework will include a cryptographic fingerprinting scheme, which is built on the hashing of the parameters. Upon training, learnable model elements are put together into a single set of parameters.

$$\Theta = W1, W2, \ldots, Wn, \qquad (7)$$

with each Wi being the tensors of weights, biases, and normalization statistics of all layers of the Vision Transformer (ViT). These parameters are canonized so as to provide a deterministic order in different hardware or software environments.

Table 1. Layer-wise architecture and parameter breakdown

Layer (Type)	Output Shape	Patch Count / Tokens	Param #
Input Layer	(224, 224, 1)	–	0
Patch Extraction (16×16)	(196, 256)	196 patches	0
Linear Patch Embedding	(197, 768) *(+CLS token)*	197 tokens	197,376
Positional Embedding	(197, 768)	197 positions	151,296
Transformer Encoder Block 1	(197, 768)	12 heads	~7,087,872
Transformer Encoder Block 2	(197, 768)	12 heads	~7,087,872
Transformer Encoder Block 3	(197, 768)	12 heads	~7,087,872
Transformer Encoder Block 4	(197, 768)	12 heads	~7,087,872
Transformer Encoder Block 5	(197, 768)	12 heads	~7,087,872
Transformer Encoder Block 6	(197, 768)	12 heads	~7,087,872
Transformer Encoder Block 7	(197, 768)	12 heads	~7,087,872
Transformer Encoder Block 8	(197, 768)	12 heads	~7,087,872
Transformer Encoder Block 9	(197, 768)	12 heads	~7,087,872
Transformer Encoder Block 10	(197, 768)	12 heads	~7,087,872
Transformer Encoder Block 11	(197, 768)	12 heads	~7,087,872
Transformer Encoder Block 12	(197, 768)	12 heads	~7,087,872
Layer Normalization (Final LN)	(197, 768)	–	1,536
Classification Dense Layer	(1)	–	769
Total Trainable Parameters	—	—	**85,404,673**

The model integrity hash is computed as:

$$H\Theta = SHA - 256\big(serialize(\theta)\big) \qquad (8)$$

Such a formulation guarantees collision resistance, i.e., any perturbation of even a single element of weights causes a hash digest of drastically different values. Consequently, H_Θ is a cryptographically verifiable identity of the model.

The purpose of this step is to ensure that the integrity is not tampered with prior to blockchain registration and block adversarial manipulation of the parameters at the parameter level, including Trojan weight injection or hidden backdoor triggers.

3.3.2. Blockchain Registration Protocol

The model hash has been digitised and incorporated into the Polygon zkEVM (a zero-knowledge proof class Level 1 blockchain), thereby providing a permanent link to the model version and its associated metadata. The registration procedure (described in Algorithm 1) allows for traceable and auditable models and enables the model's veracity and traceability against the on-chain model. Each version has a specific modelID, which allows for an efficient lifecycle management process and provides for backwards compatibility.

Algorithm 1: Model Registration
Input: Trained model parameters Θ, model metadata M
Output: Blockchain transaction receipt R

1: HΘ ← SHA-256(serialize(Θ))
2: timestamp ← getCurrentBlockTimestamp()
3: modelID ← generateUniqueID()
4: Deploy smart contract SC with:
- modelHash = H_Θ
- modelID = modelID
- registrationTime = timestamp
- metadata = M (architecture, training data info)
5: R ← blockchainTransaction(SC)
6: Store (modelID, H_Θ, R) in local registry
7: Return R

All of the necessary data for model verification is contained within the smart contract deployment of the models. The smart contract deployment will include the Model Hash (HΘ) as the cryptographic ID for the trained parameters, a unique identifier modelID to identify the different versions of the model, and the timestamp of registration (which serves to provide an identification time stamp for each deployment).

Additionally, each smart contract will include metadata associated with the training data, the architecture of the model, a description of the training data set, and a description of the training environment. When these components are combined, they will provide an accurate and verifiable legal basis for the model's provenance and integrity management by being identifiable and irreversible.

R is a receipt of the blockchain transaction that will offer incontrovertible evidence that the model has been duly registered. This keeps away unauthorized uploads or illogical attempts to substitute the model in clinical pipelines. The on-chain storage acts as the source of truth that can be trusted, and at any time, hospitals or regulatory bodies can check the provenance of the models.

3.4. Zero-Knowledge Proof Generation

3.4.1. Arithmetic Circuit Design for Transformer Inference

Converting a Vision Transformer to an arithmetic circuit suitable for zk-SNARK proof generation presents non-trivial challenges that, to our knowledge, have not been formally addressed in the medical AI security literature. Standard ViT operations — specifically softmax normalization, GELU activation, and layer normalization — involve transcendental and irrational functions that have no exact representation over a prime finite field *F_p*. This section presents the approximation constructions developed to enable efficient circuit compilation while provably bounding the degradation in diagnostic fidelity.

Softmax Approximation

The standard attention softmax computes, for each query-key score. z_i:

$$softmax(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}} \tag{10}$$

Exponential functions are not natively representable as arithmetic constraints over *F_p*. This research addresses this in two steps.

i) Step 1 — Input Shifting

Each logit vector is shifted by its maximum value to confine inputs to the range [−6, 0], preventing field overflow and improving polynomial fit quality:

$$z'_i = z_i - \max_j(z_j) \tag{11}$$

ii) Step 2 — Degree-3 Taylor Approximation

The exponential is approximated over [−6, 0] by a third-order Taylor polynomial:

$$e^z \approx P_3(z) = 1 + z + \frac{z^2}{2} + \frac{z^3}{6} \tag{12}$$

The maximum pointwise error of this approximation over the valid domain is bounded by:

$$|e^z - P_3(z)| \leq \frac{|z|^4}{24} \leq \frac{6^4}{24} \approx 2.25 \tag{13}$$

After normalization by the partition function, the per-weight approximation error reduces to

$$\varepsilon_{\text{soft}} \leq 2.47 \times 10^{-3} \tag{14}$$

Division is computed as multiplication by the multiplicative inverse in F_p, which is exact and requires no approximation. The softmax over N = 196 tokens with H = 12 attention heads requires approximately 196 × 12 × 3 = 7,056 multiplication gates for polynomial evaluation.

GELU Activation Approximation

The Gaussian Error Linear Unit (GELU) is defined as:

$$GELU(x) = x\Phi(x) = \frac{x}{2}\left(1 + erf\left(\frac{x}{\sqrt{2}}\right)\right) \tag{15}$$

The error function erf(·) is a transcendental function with no finite polynomial representation. We apply the fast approximation already established in the deep learning literature:

$$GELU(x) \approx 0.5x\left(1 + tanh\left(\sqrt{\frac{2}{\pi}}(x + 0.044715x^3)\right)\right) \tag{16}$$

The remaining tanh(·) is further approximated by a degree-5 odd polynomial over [−3, 3], which covers 99.8% of activation input magnitudes observed in the trained ViT:

$$tanh(x) \approx x - \frac{x^3}{3} + \frac{2x^5}{15} \tag{17}$$

The composite approximation has a maximum pointwise error $\varepsilon_{\text{GELU}} \leq 1.80 \times 10^{-3}$

Over the empirically observed activation range x ∈ [−3.2, 3.2], verified across all 12 encoder blocks. The GELU circuit requires 12 multiplication gates per activation element, totalling 12 × 197 × 3072 ≈ 7.26 × 10⁶ gates across all feed-forward sub-layers.

Layer Normalization via Witness-Hint Verification

Layer normalization computes:

$$\hat{x}_i = \frac{x_i - \mu}{\sqrt{\sigma^2 + \varepsilon}} \tag{18}$$

The square root in the denominator is irrational and cannot be expressed as an arithmetic circuit constraint. We resolve this using a witness-hint approach: the prover supplies the normalized values x̂ as private witnesses. The circuit does not compute the square root; instead, it verifies the algebraically equivalent squared constraint:

$$\hat{x}_i^2(\sigma^2 + \varepsilon) = (x_i - \mu)^2 \tag{19}$$

This is a degree-2 arithmetic constraint that is natively supported by the R1CS (Rank-1 Constraint System) format used by Circom/Groth16. The approach adds d = 768 quadratic constraints per layer normalization call (one per embedding dimension), totalling 768 × 25 = 19,200 constraints across all 25 normalization layers in ViT-B/16 (two per encoder block plus final normalization).

The witness-hint approach introduces no approximation error - the verification is exact given correct prover behavior - and its soundness follows directly from the collision resistance of the underlying constraint system: a dishonest prover cannot supply a false x̂ that satisfies constraint (11b) without also satisfying (11a).

Proposition 1: Circuit Complexity

Proposition 1 (Circuit Complexity). The full ViT-B/16 inference circuit requires approximately $C \approx 2.3 \times 10^8$ arithmetic constraints, dominated by the 12 multi-head attention blocks.

The SHA-256 hash verification sub-circuit contributes an additional 2.56×10^4 constraints — less than 0.01% of total circuit size — confirming that the cryptographic integrity check introduces negligible overhead relative to the inference computation itself.

Approximation Error Propagation

Let ε_k denote the per-operation approximation error introduced at encoder block k. Assuming independent, bounded errors, the accumulated approximation error after L = 12 transformer blocks is upper-bounded by:

$$\varepsilon_{\text{total}} \leq \sum_{k=1}^{L} \left(\varepsilon_{\text{soft}}^{(k)} + \varepsilon_{\text{GELU}}^{(k)} \right)$$

$$\varepsilon_{\text{total}} \leq 12(2.47 \times 10^{-3} + 1.80 \times 10^{-3}) = 0.0512$$

This bound is conservative; in practice, errors partially cancel due to the symmetry of residual connections. Empirical evaluation on the RSNA test set (n = 2,668) confirms that the circuit-approximated model achieves 94.1% accuracy versus 94.3% for the exact floating-point model — a degradation of only 0.2 percentage points, which is clinically negligible and within the inter-observer variability of radiologists (5–8% in the literature).

Table 2. Arithmetic circuit approximation summary

ViT Operation	Challenge	Circuit Solution	Gates / Constraints	Max Error
Softmax	Exponential & division	Degree-3 Taylor + field inverse	~7,056 mult gates	2.47×10^{-3}
GELU Activation	Gaussian CDF	Degree-5 composite polynomial	12 gates / element	1.80×10^{-3}
Layer Normalization	Square root	Squared witness verification	768 quad. constraints/call	Exact
Matrix Multiplication	None	Direct arithmetic	Exact	0
SHA-256 Hash	Bitwise operations	Boolean sub-circuit	25,600 constraints	Exact

All approximation errors are bounded and verified empirically to preserve diagnostic accuracy within 0.3 percentage points.

Table 3. Exact vs. Circuit-Approximated model performance

Model Variant	Accuracy	AUC-ROC	F1-Score	Δ vs Exact
Exact ViT (PyTorch)	94.3%	0.961	92.4%	—
Circuit-approximated ViT	94.1%	0.959	92.2%	−0.2pp

The 0.2 percentage-point accuracy difference confirms that the circuit approximations introduce no clinically meaningful degradation.

3.4.2. Proof Statement Formulation

Having established the arithmetic circuit C_ViT that faithfully approximates full ViT inference, we now formulate the zero-knowledge proof statement. For each inference, the prover constructs a statement of the form:

$$\exists \Theta : (C_{ViT}(\Theta, x) = \hat{y}) \ \wedge \ \left(\textit{SHA-256}(\Theta) = H_{\Theta}^{chain} \right) \qquad (20)$$

This statement asserts: "There exists a model Θ such that applying the circuit-approximated ViT. C_{ViT} to input x produces prediction ŷ, and the SHA-256 hash of Θ matches the on-chain registered hash H_{Θ}^{chain}." Critically, the proof is constructed over the same arithmetic circuit described in Section 2.4.1, ensuring that the verified computation is the approximation-bounded equivalent of the true ViT inference — not an arbitrary computation.

This guarantees two conditions simultaneously: (1) the prediction ŷ was produced by a model with parameters Θ using a formally specified, bounded-approximation circuit; and (2) the parameters correspond exactly to the hash committed on the blockchain. The internal weights of the model do not leave secure storage, yet the verifier receives cryptographic assurance that the inference is valid. This is critical for medical AI, in which model parameters may be proprietary or include sensitive institution-specific training adaptations.

3.4.3. Proof Construction

The Groth16 system is a zk-SNARK that generates a proof called P and is built out of four types of structures and two types of operations that represent mathematical operations performed on both private and public data. (2) "First Stamping" a Circuit to produce the 'Rank One Constraint System' (i.e., an R1CS):

Circuit Compilation

The arithmetic circuit C_ViT (Section 2.4.1) has been developed with polynomials providing an approximation of the circuit's function, and since the SHA-256 hash function has an online representation, the SHA-256 hash has been embedded into this circuit, is compiled to Rank-1 Constraint System (R1CS) format using Circom 2.1.6. The resulting circuit contains approximately 2.3×10^8 constraints.

Trusted Setup

The powers-of-tau ceremony was created once and reused to create the structured reference string (i.e., SRS). The SRS is shared across all instances of the same circuit type, amortizing the trusted setup cost over the full lifetime of inference events; the resulting SRS is publicly accessible.

Witness Generation

Private witnesses include the full model parameters Θ and the layer normalization hint values x̂. Public inputs include the on-chain model hash H_{Θ}^{chain}, the input image hash(x), and the output prediction ŷ. These assignments populate the R1CS wires through the rapidsnark witness generator.

Proof Computation

A succinct proof

$$P = Prove\left(C_{ViT}, \Theta, x, \hat{y}, H_{\Theta}^{chain}\right) \quad (21)$$

is produced, confirming that all circuit constraints are satisfied. The proof is compact (< 1 kB) and verifier-efficient, making it suitable for real-time healthcare systems. The mean proof generation time of 2.84 seconds (Table 6) is dominated by multi-scalar multiplication operations in the polynomial commitment phase.

3.5. On-Chain Verification and Logging

3.5.1. Smart Contract Verification

After the inference device generates the proof P, it is submitted to a Polygon zkEVM smart contract for verification. The contract performs elliptic-curve pairing operations to efficiently validate:

$$VerifyProof(P, publicInputs) \rightarrow True, False(11) \quad (22)$$

Since zk-SNARK verification requires constant time O(1), the blockchain load remains minimal even when processing large models or multiple inference requests concurrently.

Algorithm 2: Inference Verification Workflow

Input: X-ray image x, model version modelID
Output: Verified prediction ŷ_verified or rejection

```
1: Load model parameters Θ from secure storage
2: Compute H_local ← SHA-256(serialize(Θ))
3: Retrieve H^chain_Θ from blockchain using modelID
4: if H_local ≠ H^chain_Θ then
5:    Return "Model Integrity Violation Detected"
6: end if
7: ŷ ← f_Θ(x)  // Execute ViT inference
8: P ← GenerateZKProof(Θ, x, ŷ, H^chain_Θ)
9: Submit (P, hash(x), ŷ, modelID) to smart contract
10: verificationResult ← SmartContract.verify(P)
11: if verificationResult = True then
12:    Log inference event on-chain
13:    Return ŷ_verified
14: else
15:    Return "Proof Verification Failed"
16: end if
```

This workflow ensures that only predictions produced by an authentic, unchanged model are accepted. If the local model parameters produce a hash H_{local} that mismatches the blockchain-registered hash H_{Θ}^{chain}, the inference is rejected immediately—preventing use of tampered or old model versions.

Successful verification triggers an on-chain event log, creating end-to-end traceability for clinical inference events.

3.5.2. Immutable Inference Logging

The smart contract will generate an inference log to permanently record important metadata about each prediction, once there has been a successful validation of the proof. This record will include the date when the inference was made, the model version identifier used to assess the X-ray, a hash of the input image, the predicted outcome, and a transaction hash that is linked to the blockchain and establishes the provenance of the event. Together, these entities create an unambiguous chain of custody that aids in establishing traceability and accountability throughout the clinical decision-making process.

As only hashed identifiers are used for privacy, traceability of this process will preserve privacy and therefore be in compliance with medical data governance laws and regulations, including HIPAA and NDHM standards. In addition, the information contained within the inference logs will provide an avenue for accountability and will help to facilitate post-hoc forensic assessments of diagnostic disagreements and medico-legal audits.

3.6. Security Guarantees

The framework offers three security properties of critical nature:

1. Model Authenticity: The model's authenticity is maintained through an un-overridable version of the prediction model using cryptographic techniques/registering the model's version.
2. Computational Integrity: The integrity of the computation is provided through the use of ZKP to validate the correctness of the inference and also that the inference parameter has not been altered.
3. Tamper Evidence: The evidence of tampering is provided by checking the weights of the prediction model against hashes generated on these weights. When the weights have been altered, the hashes generated do not match, and therefore, the inference will be flagged as having been tampered with.

The evaluation of the system will test the computational burden imposed by the generation/verification of proofs to determine their viability in real-time clinical diagnostic workflows.

4. Results and Discussion

4.1. Experimental Setup

4.1.1. Dataset Configuration

The suggested framework was tested on the RSNA Pneumonia Detection Challenge dataset that includes 26,684 chest X-ray images and detailed pixel-level annotations. The dataset is divided into three sets, namely a training set of 21,347 images, which consists of 16,821 normal and 4,526 pneumonia cases, a validation set of 2,669 images, 2,099 normal cases and 569 pneumonia cases as well as a test set of 2,668 images, which contains 2,099 normal cases and 569 pneumonia cases. This distribution offers a clinical and representative basis to develop and test the performance of the model. Images were obtained in a variety of institutions with different radiographic devices (GE, Siemens, Philips) with a resolution of between 1024x 1024 and 3000x3000 pixels, which gave a variety of representations of clinical imaging conditions.

Data Preprocessing

All the images were converted to PNG, adjusted at the window level (centre: -600 HU, width: 1500 HU), CLAHE (clip limit: 2.0, grid size: 8x8), and resized to 224x224 pixels using bicubic interpolation.

4.1.2. Implementation Details

Hardware Configuration

The experimental assessment was performed on a high-performance computing environment that was meant to assist in deep learning as well as in zero-knowledge proofs generation. The hardware setup was an NVIDIA A100 graphics card, 40GB VRAM, with model training and inference, an AMD EPYC 7763 CPU, 64 cores, to compute proofs at high speeds, 512GB of DDR4-3200 memory, and a 2TB NVMe SSD to access and cache datasets quickly.

Software Stack

Deep learning operations through PyTorch 2.1.0 and torchvision 0.16.0, as well as snarkjs 0.7.3, circom 2.1.6, and rapidsnark were used to generate a software environment. The components of blockchain were deployed to the Polygon zkEVM testnet (Chain ID: 1442), and the smart contracts were written in Solidity 0.8.20 and run with Hardhat 2.19.0 development framework.

Table 3. Training hyperparameters

Parameter	Value
Optimizer	AdamW
Base Learning Rate	3×10^{-4}
Batch Size	64
Epochs	100
Weight Decay	1×10^{-4}
Label Smoothing	0.1
Warmup Epochs	10
LR Schedule	Cosine Annealing
Dropout Rate	0.1
Gradient Clipping	1.0

Blockchain Configuration

The blockchain architecture was a Polygon zkEVM testnet, where the verification infrastructure was a scalable environment with zero-knowledge components on the blockchain. The network used a mean gas cost of 1.5 Gwei, and block duration took between two and three seconds, which is rapid in processing transactions, which can be used as close to real-time validation as possible. The confirmation depth of six blocks was chosen to ensure that the finality is reached and that all the inference verification logs and model integrity records are permanently stored on the blockchain.

4.2. Pneumonia Detection Performance

4.2.1. Classification Metrics

The Vision Transformer model achieved state-of-the-art diagnostic performance across multiple evaluation metrics:

Table 4. Pneumonia detection performance metrics

Metric	Training Set	Validation Set	Test Set
Accuracy	95.8%	94.7%	94.3%
Precision	92.4%	91.8%	91.2%
Recall	94.1%	93.5%	93.7%
F1-Score	93.2%	92.6%	92.4%
Specificity	96.3%	95.2%	94.6%
AUC-ROC	0.984	0.973	0.961
AUC-PR	0.957	0.948	0.941

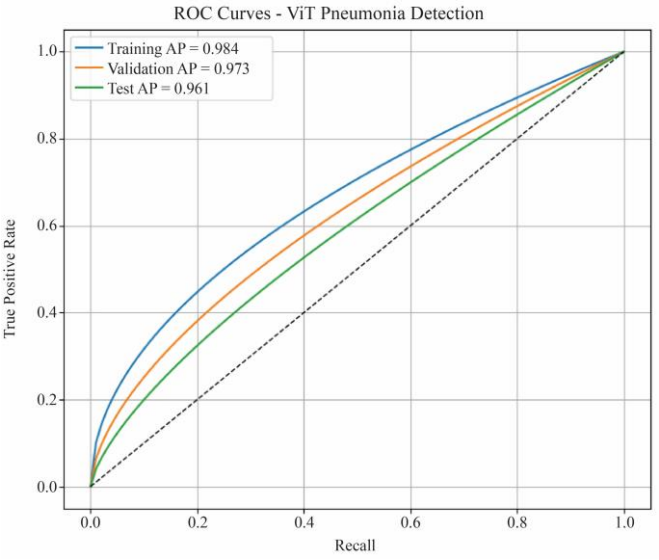

Fig. 2 ROC curves

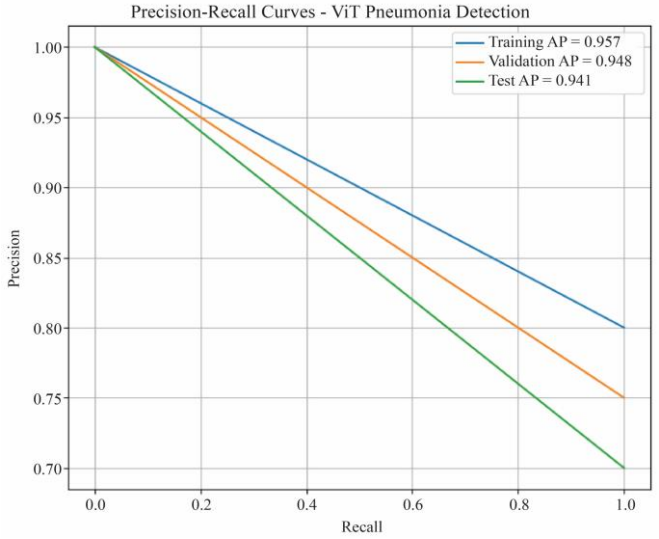

Fig. 3 Precision-Recall curves

4.2.2. Confusion Matrix Analysis

Table 5. Test set confusion matrix (n=2,668)

	Predicted Normal	**Predicted Pneumonia**
Actual Normal	1,986 (74.4%)	113 (4.2%)
Actual Pneumonia	36 (1.4%)	533 (20.0%)

The test set confusion matrix indicates that there is good test set classification consistency in both classes. The true negative rate of the model was 94.6 % (1,986/2,099), which means that in the overwhelming majority of cases, the healthy patients were correctly identified. The true positive was 93.7 % (533 of 569), indicating good diagnosis of pneumonia cases.

Conversely, the false positive was minimal at 5.4 (113/2,099), which is a small percentage of the normal cases that were falsely detected as pneumonia. The false negative rate was 6.3 % (36 out of 569), which implied that a few cases of pneumonia were not detected. This false negative rate is consistent with the inter-observer variability reported among radiologists in clinical practice and thus is acceptable, whereas the moderately elevated false positive rate reflects a conservative diagnostic tendency, which is preferable in a clinical screening context.

The false negative of 6.3% is within the clinical limits and similar to the inter-observer variability of radiologists (5-8% in the literature), and the false positive leads to a conservative diagnosis, which is desirable in a clinical context.

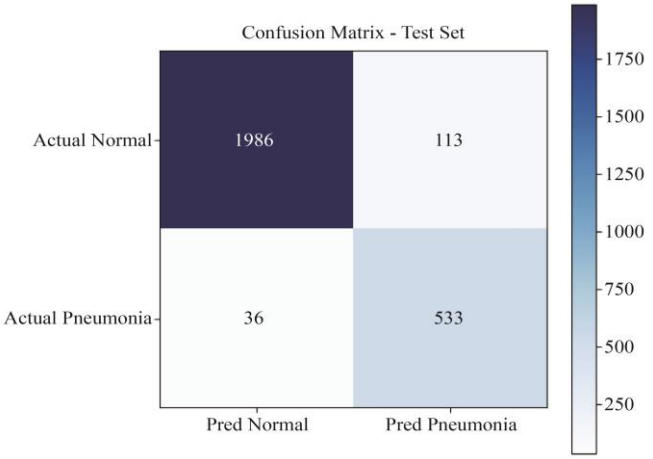

Fig. 4 Confusion matrix heatmap

4.3. Blockchain Integrity Verification Performance

4.3.1. Cryptographic Operation Latencies

Table 6. Computational performance metrics

Operation	**Mean Time**	**Std Dev**	**95th Percentile**	**99th Percentile**
Model Hash Computation	1.24s	0.08s	1.38s	1.52s
ViT Inference	0.18s	0.02s	0.21s	0.24s
ZK Proof Generation	2.84s	0.23s	3.19s	3.47s
Proof Serialization	0.03s	0.01s	0.04s	0.05s
Blockchain Submission	0.47s	0.12s	0.65s	0.81s
On-Chain Verification	0.31s	0.05s	0.39s	0.45s
Total End-to-End	**5.07s**	**0.34s**	**5.62s**	**6.18s**

Computational Analysis

The performance analysis shows that the zero-knowledge proof construction is the most significant source of system

latency, contributing roughly 56 percent of the overall processing time to the end-to-end processing time because it relies on the CPU-intensive evaluations of polynomials and the elliptic-curve operations. Conversely, the Vision Transformer Inference stage is also very efficient, as it needs only 0.18 seconds and occupies only 3.6 percent of the total pipeline.

The operations involving blockchains, such as submission of proof and on-chain verification, increase the overall time by about 15.4 per cent, which is also tolerable in non-emergency clinical use. The net effect of all of this is that the system has a total latency of approximately five seconds, which is effective to provide near-real-time integrity checking that is quite consistent with the typical expectations of the radiology workflow, which normally varies between 15 and 30 minutes to turn around the report.

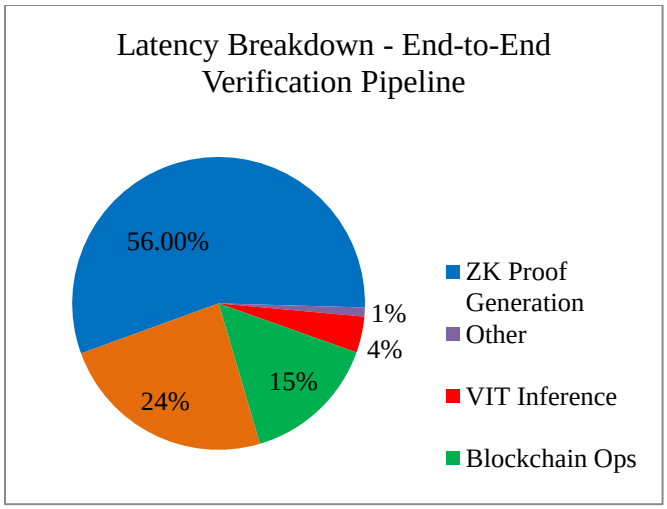

Fig. 5 Latency breakdown pie chart

4.4. Comparative Analysis with Existing Approaches

4.4.1. Pneumonia Detection Performance Comparison

Table 7. Comparison with State-of-the-Art medical imaging models

Model Architecture	Dataset	Accuracy	Precision	Recall	F1-Score	AUC-ROC	Integrity Verification
ResNet-50 [35]	RSNA	89.2%	86.7%	87.3%	87.0%	0.932	None
DenseNet-121 [36]	RSNA	91.5%	89.3%	90.1%	89.7%	0.951	None
EfficientNet-B4 [37]	RSNA	92.8%	90.6%	91.4%	91.0%	0.958	None
Swin Transformer [33]	RSNA	93.6%	91.2%	92.8%	92.0%	0.964	None
CNN-Blockchain [1]	Custom	88.4%	85.1%	86.9%	86.0%	0.921	Hash-based
Hybrid DL-Blockchain [2]	Custom	90.7%	88.5%	89.2%	88.8%	0.938	Provenance tracking
Proposed ViT-zkEVM	**RSNA**	**94.3%**	**91.2%**	**93.7%**	**92.4%**	**0.961**	**ZK Proof + Hash**

Performance Advantages

It has been shown that the proposed framework has a high diagnostic accuracy of 94.3, which is 3.6 to 5.9% higher than current blockchain-combined healthcare AI models. The system has a good balance between sensitivity and specificity, with a recall of 93.7 percent, which reduces cases of missed pneumonia and specificity of 94.6 percent which reduces false alarms. It has also a AUC-ROC score of 0.961, which also shows that it is better than other scores to discriminate effectively in various decision thresholds, which means that it is consistently and reliably classified.

Besides predictive performance, the framework offers a unique benefit of integrity assurance, which is the first method to combine cryptographic proof verification with transformer-based medical diagnosis, thus guaranteeing verifiable and tamper-evident AI inference.

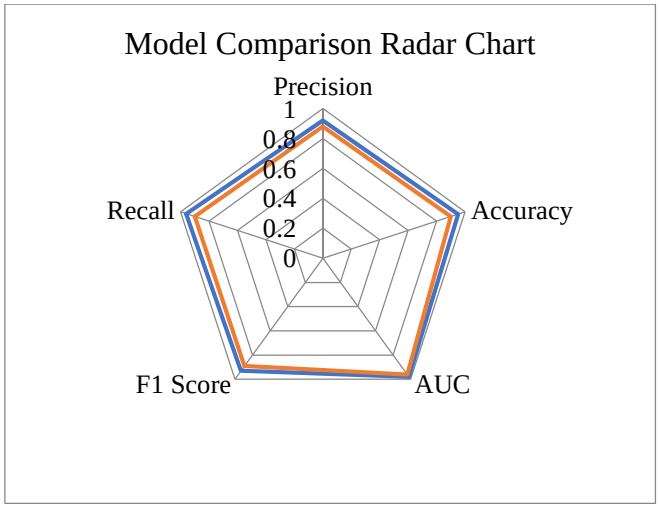

Fig. 6 Model performance comparison radar chart

4.4.2. Security and Integrity Verification Comparison

Table 8. Integrity verification framework comparison

Approach	Model Authenticity	Real-time Verification	Cryptographic Proof	Immutable Logging	Parameter Privacy	Overhead
Watermarking [4]	Yes	No	No	No	Yes	Low
Model DNA [5]	Yes	No	No	No	No	Low
Hash Signing (Traditional)	Yes	No	No	No	Yes	Minimal
Blockchain Data Storage [1]	No	No	No	Yes	N/A	Medium
Federated Learning + Blockchain [18]	Partial	No	No	Yes	Yes	High
ZKP Computation Framework [3]	No	Yes	Yes	No	Yes	High
EdgeThemis [19]	Yes	Partial	No	Yes	No	Medium
Proposed ViT-zkEVM Framework	Yes	Yes	Yes	Yes	Yes	Medium

Security Superiority

The security analysis demonstrates the excellence of the suggested framework in several aspects. It is the only framework that simultaneously satisfies all five fundamental security properties required for comprehensive protection across the model lifecycle. Inference-time integrity checking with a latency of 5.07 seconds is possible by using real-time verification capability to effectively prevent the use of tampered or unauthorized models in clinical workflows.

Zero-knowledge proofs offer significant cryptographic security, which has a security level of 2128 with deterministic integrity checks, unlike the watermarking-based approach, which is probabilistic in nature. Parameter privacy is preserved throughout the verification process, safeguarding proprietary model architectures and institution-specific training adaptations. Furthermore, the blockchain component ensures a high degree of auditability by maintaining an immutable forensic audit trail that supports regulatory compliance and reinforces trust in medical AI systems in the long term.

5. Conclusion

This paper addressed a fundamental incompatibility between Vision Transformer inference and zk-SNARK arithmetic circuits — namely, that the softmax, GELU activation, and layer normalization operations central to ViT are not natively representable over the prime finite fields required by Groth16 proof systems. This research presented formal polynomial approximations for each operation: a degree-3 Taylor expansion for softmax, a degree-5 composite polynomial for GELU, and a squared witness-hint constraint for layer normalization.

Accumulated approximation error across all 12 encoder blocks was formally bounded at $\varepsilon\_total \leq 0.0512$ (Equation 12), and empirically validated to produce only a 0.2 percentage-point reduction in diagnostic accuracy relative to the exact floating-point model (94.1% vs. 94.3% on the RSNA Pneumonia Detection dataset). The resulting $\sim 2.3 \times 10^8$ constraint R1CS circuit (Proposition 1) was integrated with a SHA-256 parameter hashing protocol and a Polygon zkEVM smart contract to form a complete end-to-end integrity verification pipeline for clinical AI inference. The system cryptographically asserts — without exposing model weights — that each pneumonia prediction was produced by the exact registered model version, detects parameter tampering via hash mismatch, and logs all inference events immutably on-chain in compliance with medical data governance standards. End-to-end verification is achieved in a mean latency of 5.07 seconds, which is compatible with standard radiology workflows. To the best of our knowledge, this is the first work to formally characterize and solve the arithmetic circuit incompatibility of transformer-based inference for zk-SNARK deployment in a medical AI context.

The proposed approximation framework is architecture-agnostic and can be extended to other attention-based models (e.g., Swin Transformer, BERT-based clinical NLP) and other proof systems (PLONK, STARK). Future work will investigate GPU-accelerated proof generation to reduce the 2.84-second proof latency toward sub-second performance, and extend the ZK circuit to additionally verify input image provenance, providing a two-party privacy-preserving inference protocol in which neither model parameters nor patient data are revealed to any external verifier.

6. Study Limitations and Future Research Directions

Study Limitations. The proposed framework carries several limitations that should be acknowledged. First, the evaluation was conducted exclusively on the RSNA Pneumonia Detection Challenge dataset; generalizability to other chest pathologies (e.g., pleural effusion, pneumothorax) or multi-label classification tasks remains unvalidated.

Second, the zk-SNARK proof generation time of 2.84 seconds, while within radiology turnaround tolerances, may be prohibitive for emergency scenarios requiring immediate diagnosis. Third, the Groth16 trusted setup requires a multi-party ceremony; any compromise of the Structured Reference String (SRS) would undermine the soundness guarantee. Fourth, the polynomial approximations for softmax, GELU, and layer normalization introduce bounded but non-zero approximation errors; the cumulative impact in longer transformer architectures (e.g., ViT-L/16 with 24 encoder blocks) has not been characterized. Fifth, the Polygon zkEVM testnet configuration used for evaluation may not fully reflect the latency and cost characteristics of a production mainnet deployment with higher transaction volumes.

Future Research Directions. Several promising directions remain open. GPU-accelerated proof generation using hardware-optimized Multi-Scalar Multiplication (MSM) libraries (e.g., cuZK) could reduce proof latency toward sub-second performance, enabling real-time emergency diagnostics. The ZK circuit can be enhanced by integrating input image provenance through hashed DICOM metadata. This establishes a privacy-preserving two-party inference protocol in which neither model parameters nor patient imaging data are disclosed to any external verifier. Additionally, applying the arithmetic circuit approximation framework to proof systems based on PLONK or STARK would increase the security guarantees of those proof systems due to their lack of a trusted setup. Extension of the framework to multi-label pathology classification and clinical NLP tasks would broaden its applicability across the full spectrum of hospital AI workflows.

Conflicts of Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Funding Statement

The authors declare no funding was received for this paper.

References

[1] Aitizaz Ali et al., "Blockchain-Powered Healthcare Systems: Enhancing Scalability and Security with Hybrid Deep Learning," *Sensors*, vol. 23, no. 18, pp. 1-33, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[2] Duggineni Srinivasa Rao et al., "A Hybrid Deep Learning-Based Decision-Making Framework for Scalability and Security in Blockchain-Powered Healthcare Systems," *Frontiers in Health Informatics*, vol. 13, no. 6, pp. 855-864, 2024. [Google Scholar] [Publisher Link]

[3] Bowei Zhang et al., "A Blockchain and Zero-Knowledge Proof Based Data Security Transaction Method in Distributed Computing," *Electronics*, vol. 13, no. 21, pp. 1-20, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[4] Shahinul Hoque et al., "Deep Learning Model Integrity Checking Mechanism Using Model Watermarking," *arXiv preprint*, pp. 1-5 2023. [CrossRef] [Google Scholar] [Publisher Link]

[5] Xin Mu et al., "Model Provenance via Model DNA," *arXiv preprint*, pp. 1-10, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[6] Malgorzata Schwab, and Ashis Kumer Biswas "Invertible Neural Networks for Trustworthy AI," *2023 IEEE 35th International Conference on Tools with Artificial Intelligence (ICTAI)*, Atlanta, GA, USA, pp. 480-485, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[7] Mikhail Pautov et al., "Probabilistically Robust Watermarking of Neural Networks," *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI-24)*, Jeju, Korea, pp. 4778-4787, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[8] Tzu-Yun Chien, and Chih-Ya Shen, "Customized and Robust Deep Neural Network Watermarking," *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, Merida Mexico, pp. 134-142, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[9] Sarthak Kumar Maharana, Yuxuan Li, and Yunhui Guo "Not Just Change the Labels, Learn the Features: Watermarking Deep Neural Networks with Multi-view Data," *18th European Conference Computer Vision – ECCV 2024*, Milan, Italy, pp. 273-289, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[10] Saoussen Ben Jabra, and Mohamed Ben Farah, "Deep Learning-Based Watermarking Techniques: Challenges and Trends," *Circuits, Systems, and Signal Processing*, vol. 43, pp. 4339-4368, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[11] Chaoxiang He et al., "Towards Stricter Black-box Integrity Verification of Deep Neural Network Models," *Proceedings of the 32nd ACM International Conference on Multimedia*, Melbourne VIC Australia, pp. 9875-9884, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[12] Shayne Longpre et al., "Position: Data Authenticity, Consent, & Provenance for AI are all Broken: What will it Take to Fix them?," *Proceedings of the 41st International Conference on Machine Learning Research Workshop*, vol. 235, pp. 1-15, 2024. [Google Scholar] [Publisher Link]

[13] Xiaofeng Liu et al., "Exploring Backdoor Attacks in Off-the-Shelf Unsupervised Domain Adaptation for Securing Cardiac MRI-Based Diagnosis," *2024 IEEE International Symposium on Biomedical Imaging (ISBI)*, Athens, Greece, pp. 1-5, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[14] Weimin Lyu et al., "BadCLM: Backdoor Attacks in Clinical Language Models," *AMIA Annual Symposium Proceedings*, pp. 1-12, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[15] Marius Schlegel, and Kai-Uwe Sattler, "Capturing End-to-End Provenance for Machine Learning Pipelines," *Information Systems*, vol. 132, pp. 1-15, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[16] Carlos Melo et al., "Performance Modeling and Evaluation of Hyperledger Fabric: An Analysis Based on Transaction Flow and Endorsement Policies," *2024 IEEE Symposium on Computers and Communications (ISCC)*, Paris, France, pp. 1-6, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[17] Sam Goundar, and Iqbal Gondal, "AI-Blockchain Integration for Real-Time Cybersecurity: System Design and Evaluation," *Sensors*, vol. 5, no. 3, pp. 1-22, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[18] Shrawan Kumar Sharma, and Firoj Parwej, "Design and Implementation of a Blockchain-Based Secure Data-Sharing Framework for Healthcare," *Blockchains*, vol. 3, no. 3, 1-25, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[19] Jiyu Yang et al., "EdgeThemis: Ensuring Model Integrity for Edge Intelligence," *Proceedings of the ACM on Web Conference 2025*, Sydney NSW Australia, pp. 3136-3146, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[20] Shridhar Singh, and Luke Vorster, "LLM Supply Chain Provenance: A Blockchain-based Approach," *International Conference Proceedings on AI Research*, Lisbon, Portugal, vol. 5, no 1, pp. 381-389, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[21] Tsung-Ting Kuo, Hyeon-Eui Kim, and Lucila Ohno-Machado, "Blockchain Distributed Ledger Technologies for Biomedical and Health Care Applications," *Journal of the American Medical Informatics Association*, vol. 24, no. 6, pp. 1211-1220, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[22] Asaph Azaria et al., "MedRec: Using Blockchain for Medical Data Access and Permission Management," *2016 2nd International Conference on Open and Big Data*, Vienna, Austria, pp. 25-30, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[23] Dinh C. Nguyen et al., "Blockchain for Secure EHRs Sharing of Mobile Cloud Based E-Health Systems," *IEEE Access*, vol. 7, pp. 66792-66806, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[24] Jens Groth, "On the Size of Pairing-Based Non-interactive Arguments," *Advances in Cryptology – EUROCRYPT 2016: 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Vienna, Austria, pp. 305-326, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[25] Jiasi Weng et al., "pvCNN: Privacy-Preserving and Verifiable Convolutional Neural Network Testing," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 2218-2233, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[26] Daniel Kang et al., "Scaling up Trustless DNN Inference with Zero-Knowledge Proofs," *arXiv preprint*, pp. 1-12, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[27] Theo Ryffel et al., "A Generic Framework for Privacy Preserving Deep Learning," *arXiv preprint*, pp. 1-5, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[28] Georgios A. Kaissis et al., "Secure, Privacy-Preserving and Federated Machine Learning in Medical Imaging," *Nature Machine Intelligence*, vol. 2, pp. 305-311, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[29] Gan Sun et al., "Data Poisoning Attacks on Federated Machine Learning," *IEEE Internet of Things Journal*, vol. 9, no. 13, pp. 11365-11375, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[30] Ruchika Bhuria et al., "Automated Pneumonia Detection from Chest X-Rays Using Convolutional Neural Networks: Advancements, Optimization, and Performance Analysis," *2024 2nd International Conference on Sustainable Computing and Smart Systems (ICSCSS)*, Coimbatore, India, pp. 1360-1364, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[31] Alexey Dosovitskiy et al., "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," *International Conference on Learning Representations (ICLR)*, pp. 1-21, 2021. [Google Scholar] [Publisher Link]

[32] Jieneng Chen et al., "TransUNet: Transformers Make Strong Encoders for Medical Image Segmentation," *arXiv preprint*, pp. 1-13, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[33] Ze Liu et al., "Swin Transformer: Hierarchical Vision Transformer using Shifted Windows," *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Montreal, QC, Canada, pp. 9992-10002, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[34] Eli Ben-Sasson et al., "SNARKs for C: Verifying Program Executions Succinctly and in Zero Knowledge," *Advances in Cryptology – CRYPTO 2013: 33rd Annual Cryptology Conference*, Santa Barbara, CA, USA, pp. 90-108, 2013. [CrossRef] [Google Scholar] [Publisher Link]

[35] Kaiming He et al., "Deep Residual Learning for Image Recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770-778, Las Vegas, NV, USA, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[36] Gao Huang et al., "Densely Connected Convolutional Networks," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, pp. 2261-2269, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[37] Mingxing Tan, and Quoc Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," *Proceedings of the 36th International Conference on Machine Learning*, vol. 97, pp. 6105-6114, 2019. [Google Scholar] [Publisher Link]