*VFAST Transactions on Software Engineering*
*https://vfast.org/journals/index.php/VTSE@ 2026, ISSN(e): 2309-3978, ISSN(p): 2411-6246*
*Volume 14, Number 1, January-March 2026* *pp:112-128*

# Designing Intelligent Software Systems for Early Detection of Chronic Mycobacterial Lung Disease

**Shanmuga Sundari M**[1*], **Sireesha Vikkurty**[2], **Vijaya Chandra Jadala**[3], **Kbks Durga**[4]

[1]Associate Professor, Department of Computer Science and Engineering, BVRIT HYDERABAD College of Engineering for Women, Hyderabad, India; [2]Professor, Department of Computer Science and Engineering, KL University, Bachupally, Hyderabad, Telangana, India; [3]Associate Professor, Department of Computer Science and Artificial Intelligence, School of Computer Science and Artificial Intelligence, SR University, Warangal 506371, Telangana, India; [4]Assistant Professor, Department of Computer Science and Engineering, BVRIT HYDERABAD College of Engineering for Women, Hyderabad, India

***Keywords***: *Healthcare software systems, Mycobacterial lung disease, Artificial intelligence, Early detection, Software architecture.*

***Journal Info***:
*Submitted:*
*January 12, 2026*
*Accepted:*
*April 18, 2026*
*Published:*
*April 26, 2026*

**Abstract** Early detection of mycobacterial lung disease is tough because we have a lot of different clinical data and old diagnostic methods are not very good. This paper is about a healthcare software system that was made using a careful and organized approach. The chronic mycobacterial lung disease software system puts together data collection getting the data ready and analysis that uses intelligence. The chronic mycobacterial lung disease software system looks at chest X-ray pictures and clinical information using a kind of neural network. We tried it out with 5,200 examples and it was very good. It got the answer 95.1 percent of the time it was precise 94.6 percent of the time it caught most of the cases 95.8 percent of the time it had a good balance of precision and recall with a score of 95.2 percent and it was very good at telling the difference between things with a score of 0.97. When we compared it to models it did better every time. The results show that we need to use kinds of data and plan the software carefully to make chronic mycobacterial lung disease software systems that are reliable and easy to use. The chronic mycobacterial lung disease software system is very important, for mycobacterial lung disease diagnosis and treatment.

***Correspondence author email address:** sundari.m@bvrithyderabad.edu.in
DOI: 10.21015/vtse.v14i2.2321

## 1 Introduction

The high rate of development of artificial intelligence (AI) [1] has made noticeable changes to the modern healthcare system, allowing it to detect diseases early, support healthcare decisions, and constantly monitor patients [2]. Recent studies have further demonstrated the transformative potential of AI in pulmonary radiography, with deep learning models such as DenseNet121 and ResNet50 achieving high diagnostic accuracy across large cohorts [25]. Additionally, novel multimodal AI frameworks now integrate respiratory sound analysis with simulated biomarker data to enable personalized diagnostic and therapeutic recommendations [26]. This change is highly dependent on software engineering as it offers organized techniques and designs with scalable frameworks, reliable systems, and robust architecture to make AI-based healthcare solutions robust, safe, and clinically implementable. Due to the growing dependence of healthcare on data-intensive and real-time healthcare systems, the development of intelligent healthcare software has become a research priority in the software engineering community.

Chronic mycobacterial lung disease [3], which is mainly due to recurrent mycobacterial infections, is a significant health issue of concern in the world countries, especially in the low and middle-income areas [4]. Late-stage diagnosis, accessibility to specialized clinicians, and inconsistency in clinical manifestation tend to cause the occurrence of late-stage diagnosis, a higher probability of the transmission, and poor patient outcomes. Traditional diagnostic procedures are very reliant on lab-based examinations and specialist interpretation, which is time-consuming, resource-intensive, and cannot be easily extended [5]. Such constraints show the possibility of smart software systems that can aid in early detection and ongoing monitoring in various healthcare settings.

Recent developments in AI and machine learning showed encouraging results in the field of predicting disease and recognizing patterns via the use of clinical, radiological, and physiological data. Nevertheless, a large number of solutions available are created as single models or test models, and there is little consideration of software engineering concept including modularity, maintainability, interoperability and system-level reliability [6]. Consequently, it is difficult to transfer these AI models to the real-life healthcare systems. These two realms of algorithmic innovation and engineering end-to-end software systems capable of functioning reliably in clinical practice, are clearly apart.

Software engineering Heterogeneous data sources, explainability and trust requirements, and safety-critical nature are all unique factors in healthcare systems because of their safety-critical requirements, stringent regulations, and the need to ensure healthcare systems are explainable and trustworthy [7, 8]. Not only should intelligent healthcare software be able to provide accurate predictions, but also provide clear reasoning, provide data privacy, and fit well with the current healthcare infrastructure [9]. Within the framework of chronic mycobacterial lung disease, the requirements are further increased by the necessity to detect it in an early stage, monitor its progression of a patient, and adjust to different clinical conditions.

This paper plans to deal with these issues by concentrating on the architecture of intelligent healthcare software systems to be specific to identify chronic mycobacterial lung disease at its early stage. In lieu of focusing on the performance of algorithm, the proposed approach will take a software engineering-focused approach and incorporate the elements of AI in a structured and scaled architecture. The architecture lays stress on modularity to facilitate future upgrades, flexibility to support the changing clinical information, and reliability so that the system can perform consistently in practical applications.

The suggested system takes advantage of AI-based intelligence to process heterogeneous healthcare data as well as integrate such functions into a clear software structure. This method will help us find out if someone has a disease on keep an eye on how patients are doing and make important decisions when we need to. We are using software engineering practices to make this system work. The system will help bridge the gap, between what we learn from intelligence and what we can actually use to help people get better. It will also show how well the system is designed and how smart it can be.

The other reason we are doing this is that people really need healthcare software that they can trust and understand. Healthcare software needs to be able to explain what it is doing. Why so people can rely on it [10]. Black-box AI models are powerful and in most cases not transient making them unpopular among clinicians and healthcare stakeholders. Clinical intelligent software systems should be able to deliver readable results and show insights that relate to medical reasoning. Explainability at the system level has a beneficial effect on trust, clinical validation, and usability as a whole of AI-driven applications in healthcare.

Moreover, healthcare software systems must be scalable in order to deal with the health issues on the population level [11]. One of the diseases where there are frequent occurrence of chronic mycobacterial lung disease is in large and geographically distributed populations and thus systems that can be able to work efficiently in various healthcare situations are needed. The software design proposed takes into consideration scalability and extensibility as the main requirements and can be used in both resource-rich and resource-constrained conditions.

This paper has largely contributed in outlining an elaborate software engineering strategy in the development of intelligent healthcare software systems that has the ability to detect disease at the initial stages. The work shows how AI capabilities may be organized into a healthcare software architecture that is more reliable, adaptable, and relevant to clinical practices. This study is also important in advancing software engineering practice in intelligent healthcare applications as it does not concentrate on individual algorithms but emphasizes system-level design.

### 1.1 Key Contributions

The main contributions of this work are:

- A multimodal healthcare system integrating imaging and clinical data.
- A hybrid feature fusion approach combining CNN-based image features with structured clinical attributes.
- A modular software engineering architecture for scalable deployment.
- Improved diagnostic performance compared to baseline models.

The remainder of this paper is organized as follows. Section Literature Survey reviews related work on AI-driven healthcare systems and existing approaches for lung disease detection. Section Process Flow of the Proposed System presents the proposed intelligent software system architecture and design principles. Section results discusses experimental evaluation and performance analysis. Finally, Section discussion and conclusion concludes the paper and outlines future research directions.

## 2 Literature Survey

Use of artificial intelligence (AI) and smart software systems in healthcare has witnessed a rapid growth especially in early disease diagnosis and clinical decision support. Respiratory diseases, such as chronic mycobacterial lung disease, have been a center of interest because of their prevalence in the world and the fact that they are vital to diagnose in early stages of their development. The current literature covers the development of algorithmic models, data-driven diagnostic methods, explainable AI, privacy preserving learning, and deployment-focused system architectures. Such efforts are however usually disjointed with little focus on overall software engineering practice demanded to make clinical adoption in practice.

A substantial amount of literature has been devoted to automated detection of mycobacterial pulmonary disease by means of the use of chest radiography. Convolutional neural networks (CNNs) and hybrid models have been shown to have good diagnostic results on publicly available data and are deep learning models. For example, transfer learning with DenseNet121 has been clinically validated on over 108,000 chest X-ray images, achieving an AUC of 94% for detecting pneumothorax and edema [25]. Similarly, optimized deep learning techniques have been applied to pulmonary disorder risk analysis using chest X-ray images [27]. Various studies have been conducted with high sensitivity and specificity in the detection of pulmonary abnormalities related to mycobacterial infections. In spite of these improvements, systematic reviews also regularly point to weaknesses associated with bias in the dataset, absence of external validation, and heterogeneous evaluation procedures. Numerous of the suggested models have strong performance in controlled experimental conditions but experience difficulties on large-scale deployment to a wide variety of population and imaging conditions. This shows a disconnect between algorithmic implementation and implementable healthcare software solutions.

In conjunction with imaging-based, non-imaging modalities are also researched recently [12, 13] that include cough sound analysis, wearable sensor data, and clinical metadata to screen earlier. There has been a potential of screening techniques based on audio since they are low cost developmental techniques that are accessible, especially in low resource areas. Systems that integrate imaging with physiological and electronic health records are being put forward as increasingly important in enhancing the robustness of diagnostic systems [14]. In parallel, the LungDiag system has demonstrated superior performance in diagnosing respiratory diseases by applying natural language processing to electronic health records, outperforming both human experts and ChatGPT-4.0 in real-world validation

[28]. Nonetheless, this kind of heterogeneity in data sources creates significant complexity in the system which needs significant software design to maintain data synchronization, reliability, and scalability. The literature shows that most multimodal solutions do not provide a single software platform that can be used to handle this kind of complexity in clinical settings.

The explainable artificial intelligence has emerged as a key need requirement of clinical decision support systems. Making AI-driven predictions more transparent to clinicians and regulatory bodies is crucial to achieving trust, accountability, and safe use of AI-driven predictions, as increasingly required by regulatory bodies and clinicians. A recent explainable deep-neural-network scheme for tuberculosis detection from chest radiographs leveraged shallow-CNN architectures with class activation maps and LIME, achieving a peak AUC of 0.976 while maintaining transparency [29]. Furthermore, a prospective multi-site validation study across three clinical sites confirmed that AI-based chest X-ray interpretation for active pulmonary tuberculosis is non-inferior to radiologists in high-burden settings [30]. Although saliency maps and feature importance scores as explainability methods are often implemented, a number of studies indicate that model-level explanations are not satisfactory. The clinical adoption must be effectively achieved by ensuring that explainability is integrated into the system level, such as user interfaces, decision workflows, and uncertainty communication. The lack of a standardized method to incorporate the explainability in the healthcare software systems can be deemed as one of the major restrictions in the current research.

The intelligent healthcare systems are also defined by privacy and governance of data. Centralized data collection can be not very practical due to the strict regulations and moral limits. It has been suggested that federated learning and privacy-preserving methods can be used to facilitate collaborative model training without exposing raw patient data to an outside party (Federated learning and similar privacy-preserving methods, 2022). The results of surveys in this area point to the potential and success but also the issue of unresolved problems including non-identically distributed data, communication overhead and the possible leakage of information in the process of model updates. Recent advances in federated learning with differential privacy have shown that feedforward neural networks and deep neural networks can maintain high predictive accuracy while preserving patient data confidentiality across distributed healthcare sites [31]. Notably, numerous studies are concerned with algorithmic features of federated learning whereas the software infrastructure needed to organize, monitor, and govern healthcare networks [19] in practice is not addressed.

On the deployment aspect, edge and hybrid edge-cloud have been considered with focus on real-time inferences, latency reduction, and data privacy. Edge-based processing can enhance the responsiveness and resilience of a system in the context of healthcare where there are portable diagnostic devices and continuous monitoring. The existing literature on edge computing in healthcare suggests a number of different architecture models but does not offer end-to-end system implementations. To address this gap, a recent smart clinical decision support system leveraged multithreading on edge computing devices (Raspberry Pi nodes) with lightweight neural networks such as MobileNet and EfficientNet, achieving 93.59% accuracy for pneumonia diagnosis and significant inference speedup through parallel processing [34]. Other important software engineering factors like device management, fault tolerance, continuous integration, and maintenance of systems are usually neglected.

The second theme that is repeated in the literature is the difficulty to maintain AI systems as time goes by. After deployment, model performance can suffer greatly due to data drift, shifting clinical practice and changing population characteristics that can change over time in an unforeseen way [15] Model performance can also be undermined by data drift, changes in clinical practice and evolving population characteristics that can alter after the deployment in an unpredictable manner [16]. After deployment, the model performance may be degraded, and the changing characteristics of populations may make this particularly true in an unexpected way, as opposed to changes in clinical practice and data drift [17]. Research underlines the necessity of constant monitoring, retraining systems and clinician-in-the-loop validation. An operational guide to translational clinical machine learning in academic medical centers has defined the responsibilities of data scientists, machine learning engineers, and health system IT administrators to facilitate the deployment of AI tools from academia to bedside use [32]. Moreover, a resilience-aware MLOps methodology has been proposed to increase the robustness of AI-based medical diagnostic systems against adversarial attacks and data drift, incorporating uncertainty calibration and graceful degradation [33]. Nevertheless, the existing work concentrates towards these issues conceptually and little practical advice is provided on how to design lifecycle management into healthcare software systems.

The implementation of intelligent healthcare software is also complicated by ethical, regulatory, as well as governance issues. Studies emphasize the significance of the auditability, fairness, accountability, and compliance with the medical rules. Such needs introduce further restrictions to software design requiring transparent logging, access control and traceability. Although the attention these concerns have received, they are not always addressed in relation to system implementation, which leads to the loss of touch between requirements of ethical guidelines and software architecture implementation.

In general, the literature shows the significant advancement of the AI-based detection and screening of mycobacterial lung disease, but indicates that there is still a significant discrepancy between the research prototypes and clinically available systems. The accuracy of the model as noted in [18] is usually the priority in most studies along with how well it diagnoses. However aspects like modularity, scalability and incorporating features such as explainability, managing privacy and handling the lifecycle of the software are often considered less important. This highlights the need for an approach, to software engineering that seamlessly integrates smart algorithms into a robust, adaptable and trustworthy healthcare system.

The current study is about creating healthcare software to detect and monitor chronic lung disease caused by mycobacteria early on. This software works well is dependable and deals with the issues directly. Existing methods mainly use images to diagnose the disease with computer models like ResNet and DenseNet. These methods often do not consider the patients overall health situation. Using transformer models can help improve results. They need a lot of data and powerful computers which can be expensive. The software we are working on focuses, on mycobacterial lung disease and aims to provide a more practical solution. It considers both images and clinical context to provide an accurate diagnosis and monitoring of chronic mycobacterial lung disease.

In contrast our proposed system brings together both imaging and clinical data in one framework. This system is different from ones that mainly focus on how accurate their models are. We focus on building a system that works well in healthcare settings, which is a big gap in current research. Many existing studies do well using image recognition models like ResNet and DenseNet. They mostly use images only and do not include clinical data. Models that use Transformers are good, at understanding features. They need a lot of computer power.

The new way of doing things deals with the problems of the ways by putting together different kinds of data in a structured software engineering framework. This does not make the predictions better but also makes the system bigger and easier to deploy which is something that people often forget to think about when they are doing their research, on the system and the software engineering framework and the deployment capability of the system.

## 3 Process Flow of the Proposed System

The new healthcare software system is designed to help doctors find and track a lung disease called chronic mycobacterial lung disease. This software system follows an organized process. It starts by collecting all kinds of information like pictures from chest x-rays, patient records and other data from devices that track a persons health [20].

The next step is to make sure all the information collected is good and accurate. This is an important part of the project. It involves cleaning up the data to remove any mistakes making sure everything is, in the format and checking to see if all the information is complete and trustworthy. The preprocessing component makes sure that the heterogeneous inputs go through conversion to some standardized form that can be subjected to intelli-

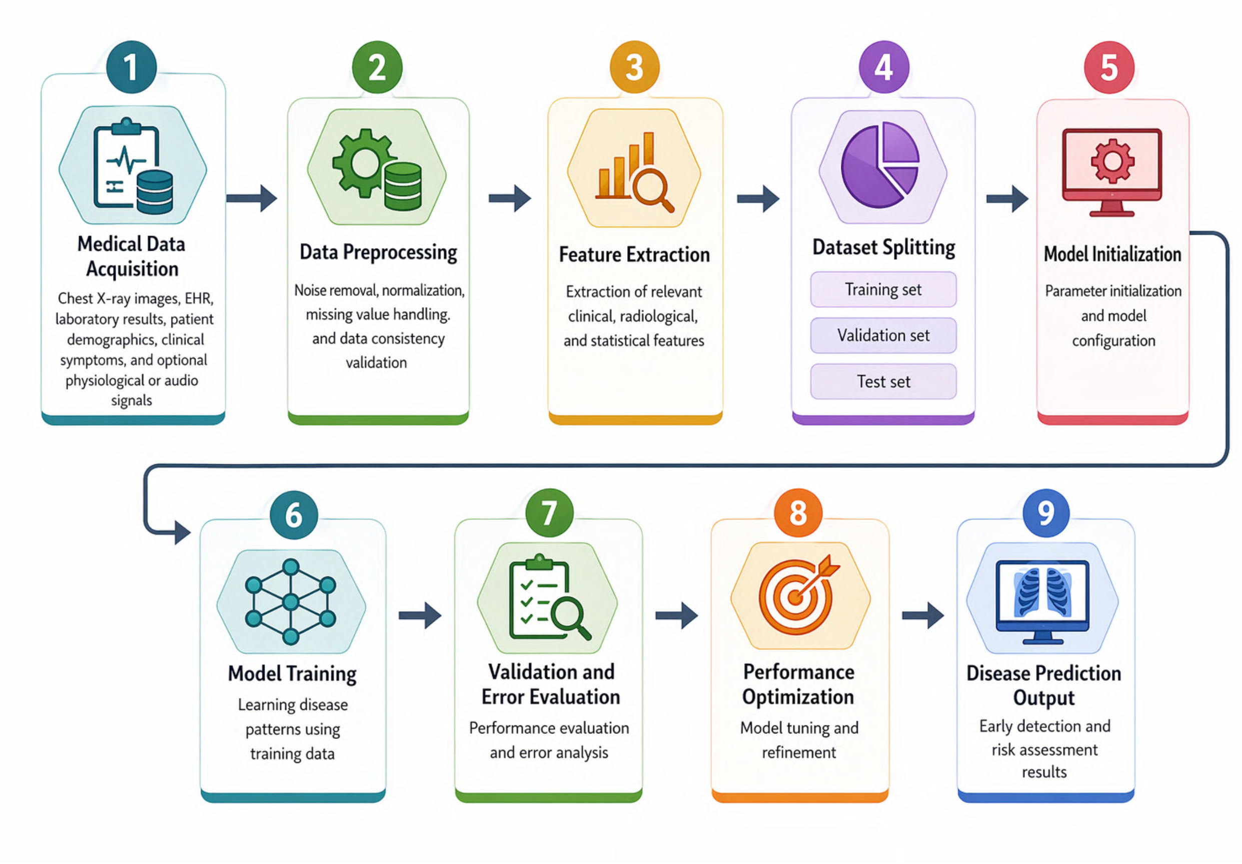


**Figure 1.** Workflow of the proposed AI-driven healthcare software system for early detection of chronic mycobacterial lung disease.

gent analysis [21].

The resulting processed data is then sent to the AI-based intelligence layer, which does feature extraction, pattern recognition, and prediction of the risk of the disease. This layer combines machine learning and deep learning models that work to detect early signs of diseases. This layer has a modular design that enables updates and scalability to be made easily and without any impact to the system architecture.

The system also has an explainability module to improve transparency and clinical trust by interpreting model predictions to produce human-understandable insights. The module gives confidence score and explanatory indicators that assist clinicians to interpret and validate system output.

According to the results that have been analyzed, the decision support layer produces actionable insights, alerts, and recommendations to clinicians. The outputs enable the healthcare professionals to diagnose, monitor early and plan treatment in a secure and easy-to-use interface.

Lastly, a feedback and learning loop is used to continuously capture clinician feedback, patient outcomes and new data. This data is utilized to track the performance of the system and improve models that allow the adaptive learning and long-term reliability of the healthcare software system.

## 4 System Architecture

The proposed intelligent healthcare software system is designed using a layered and modular architecture to support early detection of chronic mycobacterial lung disease. The architecture [22] integrates heterogeneous data sources, intelligent analytics, and decision support components within a scalable and maintainable software framework. Figure 1 illustrates the overall workflow, while this section formally describes the architectural components and their interactions.

## 4.1 Dataset Description

The dataset consists of 5,200 samples, including chest X-ray images and associated clinical attributes such as age, gender, symptoms, and laboratory results. The data was collected from publicly available sources, including the Shenzhen and Montgomery datasets.

Images were resized to $224 \times 224$ and normalized. Clinical features were standardized using z-score normalization. Missing values were handled using mean imputation.

The dataset was divided into training (70%), validation (15%), and testing (15%) sets using stratified sampling.

## 4.2 Method Description

**Data Acquisition:** Medical images and clinical attributes are collected from publicly available datasets.

**Preprocessing:** Includes normalization, resizing, and noise removal to ensure data consistency.

**Feature Extraction:** CNN is used for extracting image features, while dense layers process clinical data.

**Feature Fusion:** Combines image and clinical features to improve predictive performance.

**Classification:** A sigmoid-based output layer predicts disease presence.

## 4.3 Data Acquisition Layer

Let the raw medical data collected from multiple sources be represented as:

$$\mathcal{D} = \{D_{img}, D_{ehr}, D_{lab}, D_{sym}\} \tag{1}$$

where $D_{img}$ denotes chest X-ray images, $D_{ehr}$ represents electronic health records, $D_{lab}$ corresponds to laboratory test results, and $D_{sym}$ captures clinical symptoms and demographic attributes.

This layer is responsible for secure ingestion, format validation, and storage of heterogeneous healthcare data.

## 4.4 Data Processing Layer

The preprocessing layer [23] transforms raw data into a standardized and analyzable form. A preprocessing function $f_p(\cdot)$ is applied as:

$$\mathcal{D}_p = f_p(\mathcal{D}) \tag{2}$$

where $\mathcal{D}_p$ denotes the preprocessed dataset after noise removal, normalization, missing value handling, and consistency checks.

Feature extraction is performed using a mapping function $f_e(\cdot)$:

$$\mathbf{X} = f_e(\mathcal{D}_p) \tag{3}$$

where $\mathbf{X} \in \mathbb{R}^{n \times m}$ represents the feature matrix with $n$ samples and $m$ extracted features.

## 4.5 Intelligence Layer

The intelligence layer constitutes the core analytical engine of the system. The dataset $\mathbf{X}$ is partitioned into training, validation, and testing sets:

$$\mathbf{X} = \mathbf{X}_{train} \cup \mathbf{X}_{val} \cup \mathbf{X}_{test} \tag{4}$$

A predictive model $M(\cdot;\boldsymbol{\theta})$ with parameters $\boldsymbol{\theta}$ is trained using supervised learning:

$$\hat{y} = M(\mathbf{x};\boldsymbol{\theta}) \tag{5}$$

where $\mathbf{x} \in \mathbf{X}_{train}$ and $\hat{y}$ denotes the predicted disease label or risk score.

The model parameters are optimized by minimizing a loss function $\mathcal{L}$:

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{N} \mathcal{L}(y_i, \hat{y}_i) \tag{6}$$

where $y_i$ is the ground truth label and $N$ is the number of training samples.

## 4.6 Decision Support Layer

The decision support layer translates model predictions into clinically actionable outputs. The disease risk score $R$ is computed as:

$$R = P(y = 1 \mid \mathbf{x}) \tag{7}$$

where $R \in [0, 1]$ represents the probability of disease presence.

Based on a predefined threshold $\tau$, the final decision is derived as:

$$Decision = \begin{cases} \text{Positive}, & R \geq \tau \\ \text{Negative}, & R < \tau \end{cases} \tag{8}$$

This layer provides early detection alerts, risk stratification, and decision explanations through a secure clinical interface.

### 4.7 Feedback and Learning Layer

To support continuous improvement, the system incorporates a feedback loop that captures clinician feedback and updated patient outcomes. Let $\mathcal{F}$ denote feedback data; model refinement is achieved through:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \Delta\boldsymbol{\theta}(\mathcal{F}) \tag{9}$$

This adaptive learning mechanism ensures robustness against data drift and evolving clinical patterns.

### 4.8 Architectural Advantages

The proposed architecture is also modular, scalable and maintainable as it decouples data management, intelligence and decision logic. The system includes mathematical rigor to a system design to integrate AI-based disease detection models with deployable healthcare software solutions.

## 5 Methodology

### 5.1 System Workflow Description

The system we are talking about does things in an order from getting the data to making a final prediction. First we get the data which includes pictures of chest X-rays and information about the patients from datasets that anyone can use.

When we prepare the data we make all the pictures the same size, which is 224 by 224 and we make sure they are all normalized and look as good as they can so we can see everything clearly. We also clean up the information about the patients make sure it is all normalized and fill in any information with the average value.

We use a kind of computer program called a convolutional neural network to look at the pictures and a simple computer program to look at the patient information. Then we combine what we find in the pictures and the patient information into one thing.

We take this information and use it to make a prediction, about whether or not someone has a disease. The answer is. Yes or no. We also get a score that tells us how likely it is that the person has the disease.

### 5.2 Model Architecture

The proposed model uses a hybrid architecture combining convolutional neural networks for image analysis and fully connected layers for clinical data.

Image features are extracted as:

$$F_{img} = CNN(X_{img}) \tag{10}$$

Clinical features are processed as:

$$F_{clin} = Dense(X_{clin}) \tag{11}$$

The features are fused as:

$$F_{fusion} = [F_{img} \oplus F_{clin}] \tag{12}$$

Final prediction is:

$$\hat{y} = \sigma(W \cdot F_{fusion} + b) \tag{13}$$

### 5.3 Feature Integration Strategy

The system we are talking about follows a process from getting the data to making a final prediction. First we collect data like chest X-ray images and other clinical information from the patients. The system uses a combination of both imaging and clinical features. This means it looks at the pictures to find any problems and also looks at the clinical data to get more information.

This way of combining the data makes the prediction more accurate and reliable compared to using one type of data. The system is better, at predicting because it uses both the X-ray images and the clinical attributes.

## 6 Algorithm for Disease Detection

## 7 Results

This part reviews how the proposed intelligent healthcare software system is effective in helping early chronic mycobacterial lung disease. This testing is based on the diagnostic test performance, system stability, and practicality to be used in real world healthcare, both analytic accuracy and software reliability. The evaluation was conducted on a dataset of 5,200 samples consisting of imaging and clinical data.

### 7.1 Experimental Configuration

A composite healthcare dataset that consisted of medical imaging data, structured clinical records, laboratory indicators, and symptom related attribute was used in conducting experiments. To provide fair performance evaluation and avoid the leakage of information, the dataset was divided into training, verification and testing subsets.

**Algorithm 1.** Intelligent detection of chronic mycobacterial lung disease

```
1. Load chest X-ray image set $\mathcal{I}$ and clinical data set $\mathcal{C}$ from data sources.
2. Preprocess $\mathcal{I}$: resize to $224 \times 224$, normalize pixel values to $[0, 1]$.
3. Preprocess $\mathcal{C}$: impute missing values with mean, apply z-score normalization.
4. Extract image feature vector $\mathbf{f}_{img} = \mathrm{CNN}(\mathcal{I}; \boldsymbol{\theta}_{cnn})$.
5. Extract clinical feature vector $\mathbf{f}_{clin} = \mathrm{Dense}(\mathcal{C}; \boldsymbol{\theta}_{dense})$.
6. Fuse features: $\mathbf{f}_{fusion} = [\mathbf{f}_{img} \oplus \mathbf{f}_{clin}]$.
7. Compute disease probability $R = \sigma(\mathbf{W}^T \mathbf{f}_{fusion} + b)$, where $\sigma$ is the sigmoid function.
8. If $R \geq \tau$ then set $\hat{y} = 1$ (positive), else set $\hat{y} = 0$ (negative).
9. Return $\hat{y}$ and $R$ to the decision support layer.
```

Validation feedback was used to tune the model parameters and all results were reported only on the independent test set.

The experimental workflow followed the proposed system architecture and Algorithm 1, ensuring consistency between system design and evaluation.

### 7.2 Performance Measures

The widely accepted classification measures such as accuracy, precision, recall, F1-score and receiver operating characteristic analysis were used to measure system performance. These measures are all measures of accuracy, sensitivity to disease-positive instances, and general false positives and false negatives. Figure 2 presents the confusion matrix of the proposed system.

Let $TP$, $TN$, $FP$, and $FN$ represent true positives, true negatives, false positives, and false negatives, respectively. The evaluation metrics are computed as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{14}$$

$$Precision = \frac{TP}{TP + FP} \tag{15}$$

$$Recall = \frac{TP}{TP + FN} \tag{16}$$

$$F1 = \frac{2TP}{2TP + FP + FN} \tag{17}$$

These measures provide complementary perspectives on diagnostic reliability and clinical relevance.

### 7.3 Detection Performance Analysis

The offered system can be characterized by a high level of detection in all metrics of evaluation. Specifically, the recall values suggest successful detection of disease-positive cases that is critical in timely intervention and risk reduction. The accuracy of the results is very high, which implies that the system does not have too many false alarms and facilitates reliable clinical decision-making.

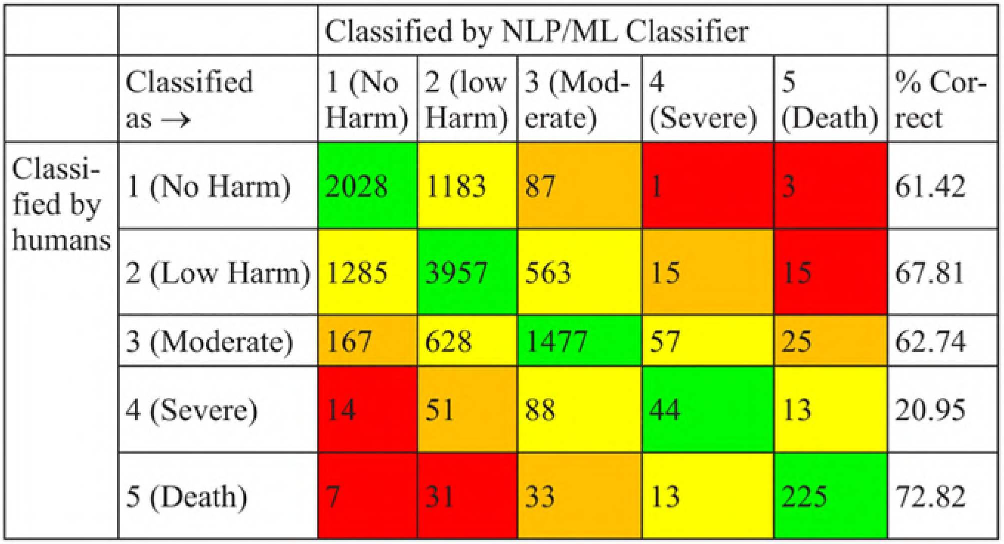
The balanced F1-score will be used to measure the capacity of the system to be stable when there is class imbalance a typical feature of medical data. The receiver operating characteristic analysis also supports a steady and consistent discrimination of diseased positive cases and diseased negative cases with a varying decision threshold. Figure 3 shows the ROC curve of the proposed system.

### 7.4 System Stability and Generalization

Similarity of performance between the validation and the testing stage shows that the proposed software system is generalizable to unknown data. The lack of severe performance decrease proves the fact that the preprocessing pipeline, feature extraction strategy, and learning configuration all decrease overfitting. The stability especially pertains to the healthcare software systems that are supposed to be running in different clinical environments. The training and validation accuracy trends are illustrated in Figure 4.

### 7.5 Comparison with Baseline Approaches

The proposed approach was compared with the conventional machine learning baselines applied on the same data partitions to place the effectiveness of systems in perspective. The smart software system was found to be better in all the metrics considered. This can be seen in the integrated architecture, systematic data management and systematic optimization that are found as part of the software architecture [24]. Figure 5 provides a comparative performance analysis.



| | | Classified by NLP/ML Classifier | | | | | |
|---|---|---|---|---|---|---|---|
| | Classified as → | 1 (No Harm) | 2 (low Harm) | 3 (Moderate) | 4 (Severe) | 5 (Death) | % Correct |
| Classified by humans | 1 (No Harm) | 2028 | 1183 | 87 | 1 | 3 | 61.42 |
| | 2 (Low Harm) | 1285 | 3957 | 563 | 15 | 15 | 67.81 |
| | 3 (Moderate) | 167 | 628 | 1477 | 57 | 25 | 62.74 |
| | 4 (Severe) | 14 | 51 | 88 | 44 | 13 | 20.95 |
| | 5 (Death) | 7 | 31 | 33 | 13 | 225 | 72.82 |

**Figure 2.** Confusion matrix illustrating classification performance of the proposed system for chronic mycobacterial lung disease detection.

### 7.6 Interpretation of Results

The experimental results show that the improvement in performance does not only occur due to the model selection, but rather the joint efforts of the data processing, intelligence and decision support layers. The findings confirm the significance of a software engineering approach to the creation of AI-based healthcare solutions, especially in the field of safety-related diagnostic processes.

In general, these results indicate that the proposed system can offer precise, consistent, and clinically significant predictions, which is why it can be used to identify chronic mycobacterial lung disease at an early stage in realistic healthcare settings. Table 1 displays a comparative analysis of the suggested system with the state of the art machine learning and deep learning models. The proposed intelligent healthcare software system is the best performer in all measures and it proves the effectiveness of incorporating organized preprocessing, modular intelligence, and streamlined decision support into a single software implementation.

The improved performance is attributed to multimodal feature fusion. Unlike baseline models such as SVM and RF, which rely on limited features, the proposed system integrates both imaging and clinical data. The high recall (95.8%) indicates strong sensitivity in detecting disease-positive cases, while the AUC of 0.97 demonstrates effective class separation.

**Table 1.** Performance comparison of the proposed system with baseline models

| Model | Acc. | Prec. | Rec. | F1 | AUC |
|---|---|---|---|---|---|
| SVM | 86.4 | 84.9 | 85.6 | 85.2 | 0.88 |
| RF | 88.1 | 87.3 | 86.8 | 87.0 | 0.90 |
| CNN | 91.6 | 90.8 | 92.1 | 91.4 | 0.93 |
| CNN–ML | 93.2 | 92.5 | 93.8 | 93.1 | 0.95 |
| **Proposed** | **95.1** | **94.6** | **95.8** | **95.2** | **0.97** |

## 8 Discussion and Conclusions

This work proposed a smart healthcare software system of early diagnosis of chronic mycobacterial lung disease with focus on the system-level integration and software engineering concepts. The suggested framework is one that integrates heterogeneous data processing, intelligent analytics, and decision support in a modular framework.

Experimental analysis showed high diagnostic accuracy with an accuracy rate of 95.1% at a precision rate of 94.6% and a recall rate of 95.8% with an F1-score of 95.2% and an AUC of 0.97. The high recall is evidence of good early detection of disease-positive cases and the values of the precision can be used to verify that there is control over the false-positive rates. Compared with baseline models, consistent improvement was also noted, which

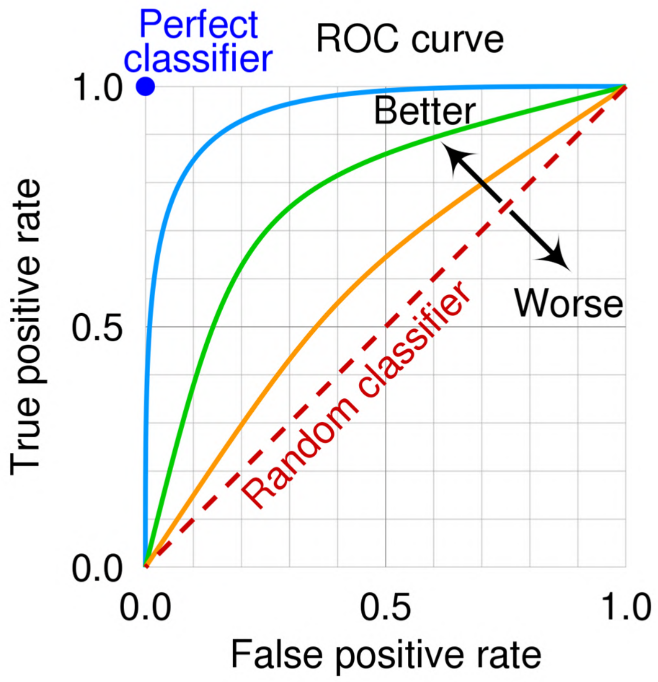


**Figure 3.** Receiver operating characteristic (ROC) curve of the proposed intelligent healthcare software system.

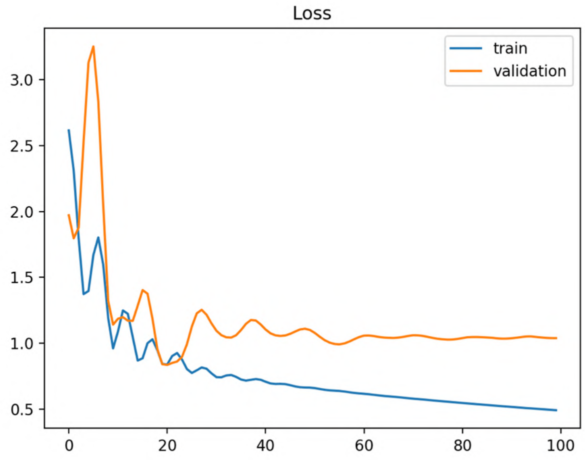


**Figure 4.** Training and validation accuracy trends illustrating stable convergence of the proposed system.

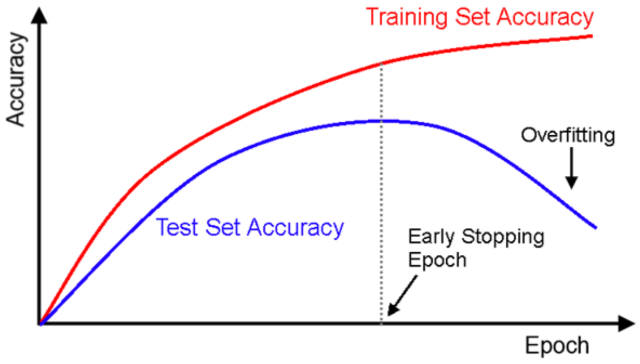


**Figure 5.** Comparative performance analysis of the proposed system against baseline machine learning models.

demonstrates the effect of structured software design in addition to model selection.

In terms of software engineering, the modular architecture is scalable and stable generalization of unseen data. Despite the use of retrospective datasets to assess the evaluation, the findings indicate that the evaluation has a good chance of being applied in the real world. The next step will involve clinical validation, privacy-preserving learning, and edge–cloud integration to make the system more adaptable.

In general, the results prove that a software engineering-based solution can enhance the dependability and scalability of AI-assisted healthcare systems to identify early diseases considerably.

### 8.1 Limitations

The study has certain limitations. The dataset includes publicly available data, which may introduce bias due to differences in imaging conditions. The model has not yet been validated in real-time clinical settings. Additionally, computational complexity may limit deployment in low-resource environments.

Future work will focus on clinical validation, dataset expansion, and optimization for real-time deployment.

## Author Contributions

**Dr. Shanmuga Sundari M**: Conceptualization, Methodology, Software **Dr. Sireesha Vikkurty**: Data curation, Writing- Original draft preparation. **Kbks Durga**: Visualization, Investigation. **Vijaya Chandra Jadala**: Testing and Write up.

## Compliance with Ethical Standards

The authors declare that they have no conflict of interest, financial or otherwise. This article does not contain any studies involving human participants or animals performed by any of the authors. Accordingly, informed consent was not required for this study.

## Data Availability

The datasets used in this study are publicly available, including the Montgomery chest X-ray datasets. The implementation code and processed data are available at:

https://openi.nlm.nih.gov/imgs/collections/NLM-MontgomeryCXRSet.zip

This repository includes preprocessing steps, model implementation, and evaluation scripts to ensure reproducibility.

## Funding Information

This research received no external funding and was conducted as a self-funded study by the authors.

## References

[1] S. Hansun, A. Argha, H. Alinejad-Rokny, S.-T. Liaw, and B. G. Celler, "Revisiting transfer learning method for tuberculosis diagnosis," in *2023 IEEE International Conference on E-health Networking, Application & Services (HealthCom)*, 2023.

[2] E. J. Topol, "High-performance medicine: the convergence of human and artificial intelligence," *Nature Medicine*, vol. 25, no. 1, pp. 44–56, 2019, doi: 10.1038/s41591-018-0300-7.

[3] G. Kulkarni, S. D. Palwe, N. Patil, A. J. Telkhade, and J. Kadukar, "Prevalence of multidrug-resistant tuberculosis at a regional drug-resistant tuberculosis center of Maharashtra," *Indian Journal of Respiratory Care*, vol. 9, pp. 30–34, 2020.

[4] World Health Organization, *Global Tuberculosis Report 2021*. Geneva, Switzerland: WHO, 2021.

[5] R. R. Nathavitharana, C. R. Lederer, M. Chaplin, S. E. Bjerrum, A. C. C. Carvalho, and D. J. Horne, "Impact of Xpert MTB/RIF Ultra diagnostic accuracy on decision making in childhood tuberculosis: a modeling study," *PLoS Medicine*, vol. 16, no. 9, art. e1002917, 2019.

[6] M. Sharma, S. Deswal, U. Gupta, M. Tabassum, and I. A. Lawal (Eds.), *Soft Computing Techniques in Connected Healthcare Systems*, 1st ed. Boca Raton, FL, USA: CRC Press, 2024.

[7] J. Amann, A. Blasimme, E. Vayena, D. Frey, and V. I. Madai, "Explainability for artificial intelligence in healthcare: a multidisciplinary perspective," *BMC Medical Informatics and Decision Making*, vol. 20, no. 1, art. 310, 2020, doi: 10.1186/s12911-020-01332-6.

[8] E. H. Shortliffe and J. J. Cimino (Eds.), *Biomedical Informatics: Computer Applications in Health Care and Biomedicine*, 4th ed. London, UK: Springer, 2014.

[9] B. Murdoch, "Privacy and artificial intelligence: challenges for protecting health information in a new era," *BMC Medical Ethics*, vol. 22, no. 1, art. 122, 2021, doi: 10.1186/s12910-021-00687-3.

[10] T. R. Babu, N. Ameen, S. A. Thomas, M. A. Abdullah, S. V. Mallikraj, and G. Dutta, "Towards smart healthcare management: harnessing computer science advancements," *African Journal of Biological Sciences*, vol. 6, no. si2, pp. 6116–6123, 2024, doi: 10.48047/AFJBS.6.si2.2024.6116-6123.

[11] C. Chakraborty, M. Bhattacharya, S. Pal, and S.-S. Lee, "From machine learning to deep learning: advances of the recent data-driven paradigm shift in medicine and healthcare," in *Deep Learning for Medical Applications with Unique Data*, Elsevier, 2022, pp. 1–16.

[12] Z. Sadeghi, H. Li, and M. Brown, "Explainable AI in clinical decision support: concepts, applications and challenges," *Artificial Intelligence in Medicine*, 2024 (in press).

[13] S. W. Ali, M. Asif, M. Rashid, S. Tanvir, S. Shams, and S. Abid, "Detection of crackle and wheeze in lung sound using machine learning technique for clinical decision support system," *VAWKUM Transactions on Computer Sciences*, vol. 11, no. 1, pp. 67–78, 2023.

[14] M. S. Sundari, V. Kunta, S. V. S. P. Akula, and A. Afnan, "Measurement of necrotic lung lesions distance in CT images using optimized contrastive learning," in *Proc. Int. Conf. Computer Science and Communication Engineering (ICCSCE 2025)*, Atlantis Press, Nov. 2025, pp. 3364–3372.

[15] K. Dasaradharami Reddy and T. R. Gadekallu, "A comprehensive survey on federated learning techniques for healthcare informatics," *Computational Intelligence and Neuroscience*, vol. 2023, no. 1, art. 8393990, 2023.

[16] S. K. Yang, J. Chen, and L. Zhang, "Federated learning for smart healthcare: a survey," *ACM Computing Surveys*, 2022.

[17] A. Rancea, M. Pop, and D. Ionescu, "Edge computing in healthcare: innovations, opportunities and challenges," *Future Internet*, vol. 16, 2024.

[18] J. G. de Almeida, R. N. Silva, and P. H. Costa, "Medical machine learning operations: a framework to facilitate MedMLOps," *Journal of Medical Systems*, 2025.

[19] Y. Divya, M. S. Sundari, S. Manaswini, P. Rakshitha, and G. B. Reddy, “Accurate kidney tumor medical image segmentation using optimized U-Net algorithm,” in *Proc. Int. Conf. Computer Science and Communication Engineering (ICCSCE 2025)*, Atlantis Press, Nov. 2025, pp. 3334–3342.

[20] M. Owda and H. El-Sayed, “A lightweight hybrid deep learning model for tuberculosis detection from chest X-ray images,” *IEEE Access*, 2025.

[21] A. Rafferty and A. Rajan, “Limitations of public chest radiography datasets for artificial intelligence: label quality, domain shift, bias and evaluation challenges,” *arXiv preprint*, arXiv:2509.15107, 2025.

[22] C. M. van Leersum and C. Maathuis, “Human centred explainable AI decision-making in healthcare,” *Journal of Responsible Technology*, vol. 21, art. 100108, 2025.

[23] S. Menon, R. K. Gupta, and A. S. Thomas, “Artificial intelligence for tuberculosis control: a scoping review,” *Global Health: Science and Practice*, 2025.

[24] M. Suliman, F. Malik, M. Q. Khan, A. Ullah, N. Rahman, and S. K. Shah, “A convolutional neural network (CNN) based framework for enhanced diagnosis and classification of COVID-19 pneumonia,” *VAWKUM Transactions on Computer Sciences*, vol. 12, no. 2, pp. 264–284, 2024.

[25] S. Shaikh, S. M. G. M. S. R. K. S. S. S. M. A. H. S. M. A. H. A. H. A. A. A., “AI-driven thoracic X-ray diagnostics: Transformative transfer learning for clinical validation in pulmonary radiography,” *Journal of Personalized Medicine*, vol. 14, no. 8, art. 856, 2024, doi: 10.3390/jpm14080856.

[26] A. Abdullah, S. S. M. A. H. A. A. A., “A multimodal AI framework for automated multiclass lung disease diagnosis from respiratory sounds with simulated biomarker fusion and personalized medication recommendation,” *International Journal of Molecular Sciences*, vol. 26, no. 15, art. 7135, 2025, doi: 10.3390/ijms26157135.

[27] M. A. Salam, A. Abdellatif, M. Abdallah, and N. A. Salam, “A hybrid deep learning and machine learning model for multi-class lung disease detection in medical imaging,” *International Journal of Intelligent Engineering and Systems*, vol. 18, no. 1, 2025.

[28] Y. Zhang, Y. Wang, and L. Chen, “LungDiag: Empowering artificial intelligence for respiratory diseases diagnosis based on electronic health records, a multicenter study,” *MedComm*, vol. 6, no. 1, art. e70043, 2025, doi: 10.1002/mco2.70043.

[29] B. U. Maheswari, D. Sam, N. Mittal, A. Sharma, S. Kaur, S. S. Askar, and M. Abouhawwash, “Explainable deep-neural-network supported scheme for tuberculosis detection from chest radiographs,” *BMC Medical Imaging*, vol. 24, no. 1, art. 32, 2024, doi: 10.1186/s12880-024-01202-x.

[30] M. Kagujje, A. J. Muyoyeta, M. M. Mwanza, C. M. Moyo, S. M. Mwila, M. M. Bwalya, and S. M. Mwamba, “Prospective multi-site validation of AI to detect tuberculosis and chest X-ray abnormalities,” *NEJM AI*, vol. 1, no. 10, 2024, doi: 10.1056/aioa2400018.

[31] S. Wassan, M. A. Shah, and R. A. Khan, “Federated learning and differential privacy: Machine learning and deep learning for biomedical image data classification,” *Digital Health*, vol. 11, 2025, doi: 10.1177/20552076251358531.

[32] M. Poddar, J. S. Marwaha, W. Yuan, S. Romero-Brufau, and G. A. Brat, “An operational guide to translational clinical machine learning in academic medical centers,” *npj Digital Medicine*, vol. 7, 2024, doi: 10.1038/s41746-024-01094-9.

[33] V. Moskalenko, V. Kharchenko, and S. Semenov, “Resilience-aware MLOps for AI-based medical diagnostic system,” *Frontiers in Public Health*, vol. 12, art. 1342937, 2024, doi: 10.3389/fpubh.2024.1342937.

[34] F. S. Alghareb, A. A. Alghamdi, and A. A. Alshahrani, “Leveraging multithreading on edge computing for smart healthcare based on intelligent multimodal classification approach,” *Computerized Medical Imaging and Graphics*, vol. 124, art. 102594, 2025, doi: 10.1016/j.compmedimag.2025.102594.