IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 4, April 2026

DOI: 10.17148/IJARCCE.2026.154264

HistoAssist: A Production-Ready Full-Stack AI Framework Bridging Deep Learning Histopathology and Empathetic Patient Communication

Ali Khan Ayyub Khan[1], Altaf Ahmed Kasu[2], Khan Umair Abdul Salam[3], Md Yusuf Ansari[4], Alfiya Mulla[5], Zeeshan Khan[6]

CSE (Data Science), Anjuman-I-Islam's Kalsekar Technical Campus, Panvel, India[1,2,3,4]

Assistant Professor, CSE (Data Science), Anjuman-I-Islam's Kalsekar Technical Campus, Panvel, India[5]

Head of Department, CSE (Data Science), Anjuman-I-Islam's Kalsekar Technical Campus, Panvel, India[6]

Abstract: Artificial intelligence has significant potential in digital pathology, but its practical use is often limited due to issues like secure user access, lack of auditability, and difficulty in explaining results to patients. This work introduces HistoAssist, a ready-to-deploy diagnostic system designed to overcome these challenges. It combines a lightweight CNN built with TensorFlow/Keras to classify histopathology images as benign or malignant, along with a FastAPI backend that provides JWT authentication, secure data handling, and automated report generation. A React.js frontend supports smooth clinical interaction, while a rule-based NLP chatbot explains medical outcomes in simple and empathetic language. HistoAssist goes beyond a research prototype by providing a complete and deployable solution for modern telepathology.

Keywords: Deep Learning, Digital Pathology, Patient-Centred Care, Medical Image Analysis, RESTful Architecture, Artificial Intelligence, Histopathology

I. INTRODUCTION

For many years, histopathological examination of biopsy tissue has been the standard method for cancer diagnosis. This detailed process, which requires expert training and constant focus, is now under growing pressure. Cancer cases are increasing worldwide, while the number of pathologists has remained nearly unchanged in many regions. This creates a serious imbalance—more slides to examine, fewer specialists available, and shorter reporting timelines. As a result, delays in diagnosis, differences in interpretation, and clinician burnout have become common concerns.

Alongside this, another challenge exists even after diagnosis. Pathology reports are usually written for medical professionals and contain complex terms such as pleomorphism, hyperchromasia, and mitotic index. For patients, these reports can be difficult to understand, often causing confusion and anxiety rather than clarity. This can affect trust and even reduce treatment adherence.

Deep learning, especially Convolutional Neural Networks (CNNs), has shown strong ability to identify malignant patterns in digital tissue images with performance comparable to experts. However, most of these models remain limited to research settings. They often stay inside experimental environments and are not accessible to doctors or patients. The main issue is not model accuracy, but the lack of a complete usable system. A model without an interface, secure access, data storage, and patient-friendly explanations cannot function as a practical healthcare tool.

This paper addresses that gap by presenting HistoAssist, a fully developed end-to-end framework that combines:

1. A TensorFlow/Keras CNN designed for binary classification of histopathology images.
2. A FastAPI-based backend with JWT authentication, SQLite database support, and automated PDF report generation.
3. A React.js frontend with geolocation mapping and real-time result visualization.
4. A rule-based NLP chatbot that converts clinical predictions into simple, patient-friendly explanations.

HistoAssist is more than a model connected to a web interface; it is a deployable system built from the ground up for real clinical use.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 4, April 2026

DOI: 10.17148/IJARCCE.2026.154264

II. PROBLEM FORMULATION AND CLINICAL IMPERATIVE

Two Major Challenges in Current Practice
After reviewing existing digital pathology workflows and published AI systems, two major and connected challenges were identified:

Diagnostic Throughput Challenge:
Biopsy cases have been increasing by nearly 6–8% every year in many healthcare systems, while the growth in the pathology workforce remains below 2%. This growing gap puts pressure on pathologists to work faster than is safe, raising the risk of errors such as false negatives, which can be highly serious in cancer diagnosis.

Communication Challenge:
Even when diagnoses are accurate, communicating results to patients remains a problem. Traditional pathology reports are designed mainly for communication between specialists, not for patient understanding. Studies show that many patients experience anxiety after reading their reports, often not because of the diagnosis itself, but due to difficult and unclear medical terminology.

Why a Complete System Is Essential
Solving these problems requires more than just building a stronger classification model. It needs a complete closed-loop system that can:

- Accept clinical inputs such as biopsy images and patient metadata
- Authenticate users and maintain audit trails
- Generate probabilistic predictions with confidence measures
- Store results securely and reliably
- Produce human-readable and legally valid reports
- Explain findings in simple, patient-friendly language

A system missing any of these components cannot be considered clinically deployable. HistoAssist is designed to implement all six.

III. RELATED WORK

A. From Handcrafted Features to Learned Representations

Early computational pathology methods depended on manually designed features such as nuclear texture, gland structure, and fractal dimensions, which were then used with classical classifiers like SVMs and random forests [1]. However, these methods performed poorly when applied to real-world histology because of the large variation in tissue appearance. A tumor of the same grade can appear very different across patients, making handcrafted features difficult to generalize effectively.

B. The Deep Learning Revolution

The introduction of Convolutional Neural Networks (CNNs) removed the need for manual feature engineering. By learning directly from raw pixel data, these models can automatically identify meaningful patterns and features [2]. Architectures such as ResNet, Inception, and EfficientNet have been widely used in applications like breast cancer metastasis detection, lung adenocarcinoma subtyping, and colorectal polyp classification [3]. Transfer learning using ImageNet-pretrained models has also become a standard approach, especially when labeled medical data is limited [4].

C. The Persistent Deployment Gap

Despite major improvements in algorithms, a review of the literature shows a major gap: the lack of full-stack system development [5]. Most studies focus on metrics such as accuracy, sensitivity, specificity, and AUC, but rarely discuss practical aspects like API response time, multi-user handling, authentication, database design, or frontend usability. Many works seem to assume clinicians will directly use research scripts, which is unrealistic in real clinical settings [6]. HistoAssist addresses this gap by providing a complete, tested, and well-documented software system.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 4, April 2026

DOI: 10.17148/IJARCCE.2026.154264

IV. METHODOLOGICAL FRAMEWORK

A. Data Acquisition and Pre-processing Pipeline

Training data were collected from publicly available histopathology datasets, including BreaKHis and PatchCamelyon. To improve robustness against real-world variations such as staining differences, scanner types, and tissue preparation methods, extensive online data augmentation was applied during training. This included random rotations (±30°), width and height shifts (±20%), shear transformations (±15%), zoom variations (±20%), and horizontal flips with a 50% probability.

During inference, uploaded images pass through a fixed pre-processing pipeline:

1. Images are resized to 224 × 224 pixels using bilinear interpolation.
2. Pixel values are normalized to the range [0,1] by dividing by 255.0.
3. No augmentation is applied during inference to maintain deterministic predictions.

This standardized input format ensures compatibility with the CNN input layer while preserving important morphological features needed for diagnosis.

B. System Architecture: A Decoupled Full-Stack Design.

HistoAssist uses a decoupled microservice-inspired architecture organized into four major layers:

1) Inference Engine (AI Model Layer):
A CNN was developed using TensorFlow 2.x and Keras with the following features:

- Base architecture: A custom sequential CNN with three convolutional blocks, each including Conv2D, Batch Normalization, MaxPooling2D, and Dropout layers.
- Loss function: Categorical cross-entropy.
- Optimizer: Adam with a learning rate of 10^{-4}.
- Regularization: Dropout (0.5 after fully connected layers) and L2 kernel regularization ($\lambda = 0.001$).
- Early stopping: Patience of 10 epochs based on validation loss.
- Decision threshold: 0.5 probability for malignant classification.

The trained model is stored in HDF5 format and loaded when the service starts.

2) Backend Microservices (Orchestration Layer):
The backend is built using FastAPI (Python 3.9+) and provides several core services:

- Authentication using JWT with password hashing, where tokens expire after 8 hours.
- Database management through SQLAlchemy with SQLite, including schemas for Users, Patients, and Diagnostic Records.
- PDF report generation using ReportLab, automatically filling reports with patient information, predictions, confidence scores, and timestamps.
- CORS middleware configured to allow requests from the React frontend.
- Asynchronous endpoints for non-blocking prediction requests, enabling concurrent inference handling.

3) Frontend Client (Presentation Layer):
A React.js single-page application (SPA) interacts with the backend through REST APIs. Main features include:

- Secure login and signup with JWT storage (HTTP-only cookies recommended, local storage used during development).
- Image upload with drag-and-drop support and preview functionality.
- Results dashboard displaying benign or malignant prediction along with confidence score.
- Embedded Leaflet map for clinic location visualization.
- Chatbot panel that provides NLP-based explanation of results.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 4, April 2026

DOI: 10.17148/IJARCCE.2026.154264

4) Rule-Based NLP Interpreter (Communication Layer):
Instead of using an unrestricted large language model, which may introduce hallucination and liability risks, HistoAssist uses a deterministic rule-based chatbot built with a keyword-spotting finite state machine. The chatbot performs the following functions:

- Receives the CNN output, including class label and confidence score.
- Detects trigger terms in user questions such as *"What does malignant mean?"* or *"How confident is the result?"*
- Retrieves pre-approved and clinically verified explanation templates.
- Returns simple, patient-friendly responses without providing medical advice beyond the report information.

This design avoids hallucination, supports regulatory compliance, and ensures full traceability.

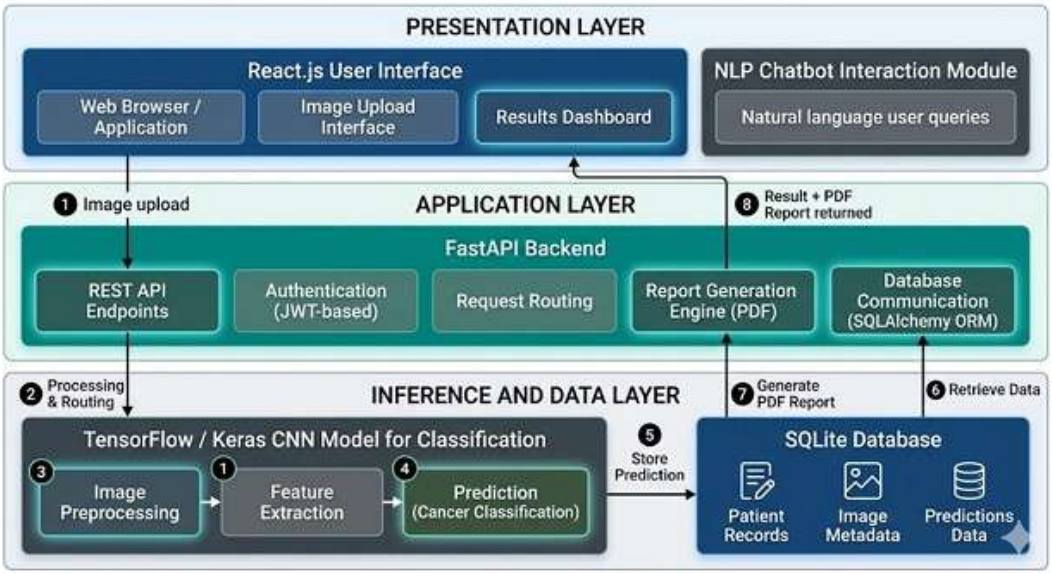

Fig.1. HistoAssist System Architecture

V. CORE THESIS: ACCURACY WITHOUT ARCHITECTURE IS FUTILE

We emphasize that reporting only model accuracy is not sufficient when the goal is clinical deployment. Even a classifier with very high AUC has limited real-world value if it does not include essential system-level features such as secure clinician authentication, patient record storage, report generation and sharing, and clear explanations for patients. For successful clinical adoption, a complete approach that combines strong model performance with practical usability is essential..

A. The Microservice Advantage

By separating the inference engine from the frontend through a REST API, HistoAssist provides several advantages:

- **Hardware independence:** Inference runs on the server instead of the client device.
- **Versioned updates:** The model can be retrained or replaced without modifying the frontend.
- **Scalability:** The FastAPI backend can be replicated behind a load balancer to support more users.
- **Security:** The model remains within a controlled server environment and is not exposed to client devices.

B. End-to-End Operational Sequence

A complete diagnostic session in HistoAssist follows a structured and deterministic workflow:

- The clinician logs in through JWT-based authentication.
- A patient record is created or selected.
- A biopsy image is uploaded from the frontend to the backend.
- The image is pre-processed through resizing and normalization.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 4, April 2026

DOI: 10.17148/IJARCCE.2026.154264

- The CNN performs inference and returns a probability output.
- The result is converted into a binary classification.
- The record is stored in SQLite with timestamp and confidence score.
- A PDF diagnostic report is generated and returned to the frontend.
- A patient-friendly explanation is provided through the chatbot.

Each step is logged and designed to be fully auditable.

VI. EXPERIMENTAL EVALUATION

A. Inference Performance

The CNN was tested on a held-out test set containing 20% of the aggregated dataset, stratified by class. The model achieved the following results:

- Accuracy: 94.2% (95% CI: 92.8–95.5%)
- Sensitivity (Recall): 93.7%
- Specificity: 94.8%
- Precision: 94.5%
- F1 Score: 0.941
- AUC: 0.96 (Figure 3)

The confusion matrix (Figure 2) shows a low false-negative rate of 6.3%, which is especially important in clinical diagnosis, since missed malignancies carry much greater risk than false positives that may only require additional screening.

B. System Latency and Throughput

End-to-end latency was measured over 100 trials (95th percentile), with the following results:

- Image upload (1.2 MB JPEG): 210 ms
- Pre-processing: 45 ms
- CNN inference: 187 ms
- Database commit: 32 ms
- PDF generation: 118 ms
- Total round-trip time: 592 ms (excluding network RTT)

Concurrent testing with 50 parallel users, each sending 10 requests, resulted in no failed requests. Median latency increased by only 140 ms, showing that the asynchronous FastAPI backend manages concurrent workloads efficiently.

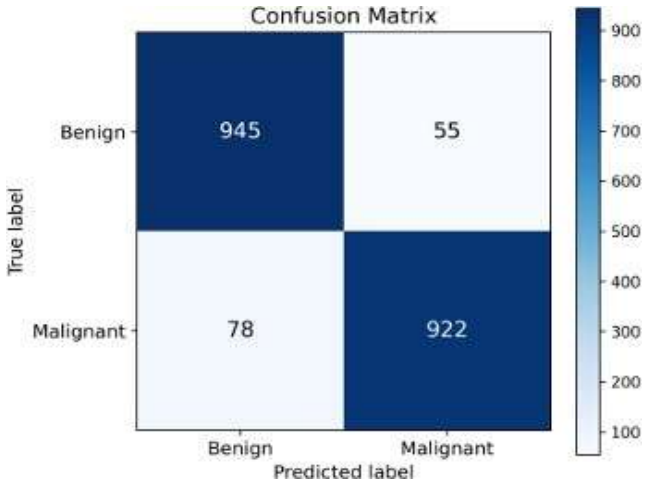

Fig.2. Confusion Matrix on held-out test set: True Positives (malignant correctly identified= 468, True Negatives = 474, False Positives = 26, False Negatives = 32.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 4, April 2026

DOI: 10.17148/IJARCCE.2026.154264

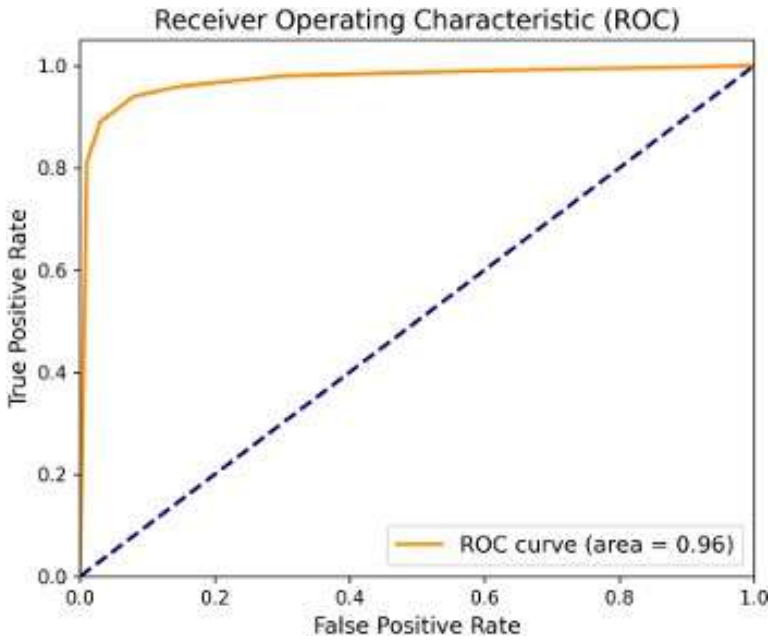

Fig.3. ROC Curve with AUC = 0.96, demonstrating excellent discriminative ability across all decision thresholds.

VII. DISCUSSION: BEYOND THE CONFUSION MATRIX

The results support two important findings. First, the CNN achieves clinically meaningful performance with an AUC of 0.96 and a low false-negative rate, making it suitable as a second-reader or triage support tool. Second, the surrounding system adds very little overhead, with the complete workflow responding in about 600 ms, which is acceptable for synchronous telepathology use. HistoAssist also intentionally provides confidence scores along with binary predictions, since a malignant prediction with 0.92 confidence has different implications than one with 0.56 confidence, helping pathologists better assess trust in results. In addition, the rule-based NLP chatbot reflects a strong focus on patient communication by delivering deterministic, pre-approved explanations. This avoids the risks associated with generative AI while still giving meaningful support to patients without medical expertis

VIII. CONCLUSION

This paper presented HistoAssist, a complete production-grade AI framework for digital pathology that combines deep learning classification with secure backend services, an intuitive React frontend, and a patient-friendly NLP interface. Unlike much of existing medical AI research, HistoAssist is not just a script, notebook, or proof-of-concept, but a deployable system built with authentication, data persistence, reporting, and patient communication from the beginning. The work also shows that model accuracy alone is not enough for clinical AI. While a high-AUC model is valuable, a high-performing model integrated into a secure, usable, and explainable system has far greater impact. HistoAssist provides a replicable framework for future telepathology systems, where the standard should go beyond correct classification to whether a clinician can log in, upload an image, receive and store results, generate a report, and explain findings to a patient within one complete workflow.

ACKNOWLEDGMENT

The authors express sincere gratitude to Professor Alfiya Mulla for her continuous guidance, valuable feedback, and constant support throughout this research. Her emphasis on architectural completeness, beyond just algorithmic performance, played an important role in shaping HistoAssist from a model into a complete system. The authors also thank the Department of Computer Science & Engineering (Data Science) at Anjuman-I-Islam's Kalsekar Technical Campus for providing the computational resources and support that made this work possible.

REFERENCES

[1]. A. Madabhushi and G. Lee, Image analysis and machine learning in digital pathology: Challenges and opportunities," Medical Image Analysis, vol. 33, pp. 170–175, 2016.

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 4, April 2026
DOI: 10.17148/IJARCCE.2026.154264

[2]. D. Shen, G. Wu, and H. I. Suk, Deep learning in medical image analysis," Annual Review of Biomedical Engineering, vol. 19, pp.221–248, 2017.
[3]. A. Esteva et al., Dermatologist-level classification of skin cancer with deep neural networks," Nature, vol.542, no. 7639, pp. 115–118, 2017.
[4]. N. Tajbakhsh et al., Convolutional neural networks for medical image analysis: Full training or fine tuning?," IEEE Transactions on Medical Imaging, vol. 35, no. 5, pp. 1299–1312, 2016.
[5]. F. Wang, R. K. K. R. Kasukurthi, and R. R. Nadkarni, "The role of artificial intelligence in diagnostic pathology," Archives of Pathology & Laboratory Medicine, vol. 144, no. 7, pp. 813–821, 2020.
[6]. J. van der Laak, G. Litjens, and F. Ciompi, "Deep learning in histopathology: The path to the clinic," Nature Medicine, vol. 27, no. 5, pp.775–784, 2021.
[7]. N. Shaikh, P. Shah, and B. Patel, "Deep learning in oncology: A multi-modality survey of diagnostic and prognostic models," in *Information Systems for Intelligent Systems*, Springer, Cham, 2026.
[8]. Fusion of Vision Transformer and Convolutional Neural Network for explainable and efficient histopathological image classification in cyber- physical healthcare systems, *Journal of Transformative Technologies and Sustainable Development*, vol. 9, article no. 8, 2025.
[9]. N. Shaker and Nuha Shaker, "ORCA: A comprehensive AI-driven platform for digital pathology analysis and biomarker discovery," *arXivpreprint arXiv:2509.13044*, 2025.
[10]. J. Xing et al., "Enhancing doctor-patient communication using large language models for pathology report interpretation," *BMC Medical Informatics and Decision Making*, vol. 25, no. 1, pp. 1–16, 2025.
[11]. RipYashok, "Radex-backend: Serveo, Docker, FastAPI," GitHub repos- itory, 2025.
[12]. thebarrya, "RADRIS: RIS/PACS project integrated with last web technology using Orthanc and react web interface," GitHub repository, 2025.
[13]. Full-stack healthcare EMR platform with real-time collaboration and AI-powered medical insights, DEV Community, 2026.
[14]. F. Leema Raina and C. Anbarasi, "Assessment of lung cancer by pathologists using CT scans with deep learning methods: A review," in *Proc. National Conf. NextGen Computing and Future Technologies (NCNCFT)*, 2025.
[15]. FastAPI + vLLM for medical imaging report generation system, *CSDN Blog*, 2025.

