Bridging the Diagnostic Gap: A Web-Based Teledermatology Framework for Accessible Early Detection of Skin Lesions using Deep Learning

Pallavi Goel, Mohd Faziel, Piyush Kumar, Md Saquib, Nilesh Chandra
Galgotias College of Engineering and Technology, Greater Noida, Uttar Pradesh, India
drpallavi.goel@galgotiacollege.edu, mohdfazel969@gmail.com, 509rajpiyush@gmail.com, mdsaquib31655@gmail.com, nilesh.chandra2002@gmail.co

Abstract
Early identification of dermatological conditions is critical for preventing long-term complications; however, timely access to qualified dermatologists remains limited, especially in rural and underserved regions. Although Artificial Intelligence (AI) has shown strong performance in skin-image interpretation, most existing systems require specialized hardware, native applications, or high-bandwidth environments. This paper introduces DermaScan, a fully web-based, device-agnostic teledermatology framework designed to deliver rapid, preliminary assessment of skin lesions using a lightweight deep-learning model. The system integrates the MERN stack for frontend–backend communication and employs an optimized MobileNetV2 network for real-time inference through a Python microservice. The proposed framework emphasizes accessibility, fairness across diverse skin tones, low computational overhead, and clinically meaningful triage performance. Experimental results demonstrate high sensitivity, low latency, and stable performance under varied lighting and device conditions. DermaScan highlights the potential of web-native AI tools to reduce diagnostic delay, support early awareness, and provide equitable dermatology assistance at scale.

Keywords: Algorithmic fairness, CNN, deep learning, diagnostic triage, MERN stack, MobileNetV2, teledermatology, web-based diagnostic systems.

1. Introduction

Skin disorders represent one of the most common categories of medical conditions worldwide, ranging from non-serious issues such as rashes or fungal infections to potentially lethal malignancies. Visual symptoms such as color variation, lesion shape, elevated patches, or irregular textural patterns often serve as early indicators, making timely assessment essential. Unfortunately, dermatology services are unevenly distributed, with urban areas typically having far greater specialist availability than rural communities. As a result, millions of individuals delay seeking care or rely on self-diagnosis, which can worsen outcomes[1].

Recent advances in computer vision and machine learning offer promising opportunities to automate early screening. Convolutional Neural Networks (CNNs) can detect subtle visual cues and learn complex dermatological patterns that might otherwise require professional expertise. However, many existing AI-based dermatology applications are constrained by practical limitations: dependence on dermoscopic equipment, large model sizes, expensive GPU-based servers, or mobile app installations incompatible with low-end devices [2].

To address these challenges, this study proposes *DermaScan*, a browser-based teledermatology system capable of analyzing smartphone-captured skin images in real time. The design focuses on inclusivity, ensuring the system runs smoothly on low-bandwidth networks, older devices, and across a wide range of skin tones. With scalable backend services, a modular CNN inference engine, and a simple web interface, DermaScan aims to bridge a diagnostic gap by providing preliminary, easily accessible dermatology insights.

2. Research Methodology

The methodological approach for this study followed a structured sequence that began with data collection, moved through preprocessing and model development, and concluded with system integration and evaluation. Each stage was designed to ensure that the final framework performs reliably in real-world web environments, where device capabilities, lighting conditions, and network quality vary widely[3].

The first step was to assemble a diverse set of dermatological images from publicly available repositories. These sources included both dermoscopic datasets and smartphone-based clinical photographs, enabling the system to learn patterns that span different imaging styles. All images were standardized before use so that the model could process them consistently. Routine preprocessing operations, such as resizing, normalization, and selective augmentation, were applied to reduce noise, balance class representation, and minimize bias linked to skin-tone variations.

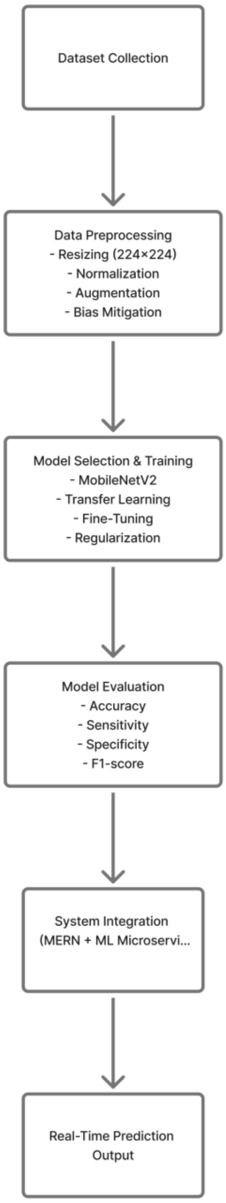

Figure 1. Block Diagram of Proposed Methodology

Model development focused on identifying a lightweight architecture that runs efficiently within a browser-supported workflow. Several convolutional neural networks were reviewed, after which a compact model was selected for its balance of accuracy, inference speed, and a small memory footprint. Transfer learning was adopted to initialize the network with general visual features, and the model was then fine-tuned on the prepared dermatology dataset. Training was monitored using validation metrics, and regularisation methods such as dropout and early stopping were used to maintain generalisation.

Following model training, the system components were integrated into a modular web architecture. The user interface was built to handle image uploads and present predictions clearly, while a backend service managed communication between the browser and the machine-learning module. The trained model operated within a separate microservice, ensuring that computational tasks did not disrupt the main application. This separation allowed independent updates, improved scalability, and maintained responsiveness.

To verify system performance, the framework was evaluated using accuracy, sensitivity, latency, and fairness indicators. These assessments ensured that the model not only produced reliable predictions but also responded quickly enough for real-time interaction. The evaluation also examined how well the system handled images originating from different skin tones and device types, reflecting the practical conditions under which users would access the platform.

3. Theory and Calculation

This section outlines the theoretical foundations that support the design of the DermaScan framework, along with the key computational steps used during model development and prediction. The discussion focuses on the mathematical and architectural principles that enable efficient feature extraction, classification, and real-time inference within a web-based environment.

3.1 Convolutional Feature Learning

The proposed system relies on convolutional neural networks, which learn visual patterns by sliding small filters across the image. These filters capture local structures such as edges, texture changes, and pigmentation irregularities. As the network deepens, earlier features are combined into more complex representations that help distinguish between different skin-lesion categories. This layered learning strategy forms the theoretical basis for automated dermatological assessment, particularly in scenarios where images vary in lighting or resolution.

3.2 Lightweight Modeling with MobileNetV2

MobileNetV2 was selected because its architecture is designed for low-resource environments. It uses depthwise separable convolutions, which split a conventional convolution into two smaller operations: one for spatial filtering and another for channel mixing. This reduces computation significantly while preserving important lesion-specific patterns. The inverted residual block further enhances efficiency by strategically expanding and compressing feature dimensions, allowing the network to detect subtle irregularities even when running within a web-centric pipeline.

3.3 Transfer Learning and Fine-Tuning

To adapt the model for dermatology tasks, transfer learning was applied. The network was initialized with general visual features learned from large-scale image datasets and then fine-tuned using skin-lesion images. This approach accelerates convergence and minimizes overfitting, especially because dermatology datasets often contain fewer samples. Fine-tuning enables the network to specialize in identifying lesion borders, color asymmetries, and other indicators commonly associated with clinical diagnosis.

3.4 Classification Computation

The final prediction layer uses a softmax function to translate model outputs into class probabilities. For a given input xxx, the probability that the lesion belongs to class i is:

$$P(y = i|x) = \frac{e^{zi}}{\sum_{k=1}^{N} e^{zk}}$$

where z_i is the logit for class iii and N is the total number of categories.

This formulation ensures that each image is assigned one dominant class while still conveying relative confidence across all possible diagnoses.

3.5 Training Loss and Optimization

Model training minimizes categorical cross-entropy, which measures the divergence between predicted probabilities and the true label:

$$L = -\log\big(P(y = j|x)\big)$$

The Adam optimizer was used to adjust parameters adaptively, enabling stable learning across different image conditions. Regularization techniques, including dropout and early stopping, helped the model generalize to new inputs and reduced the likelihood of overfitting to specific datasets[6].

3.6 Evaluation Metric Calculations

Several metrics were computed to measure diagnostic reliability:

Sensitivity (Recall)

$$\text{Sensitivity}=\frac{TP}{TP+FN}$$

Specificity

$$\text{Specificity}=\frac{TN}{TN+FP}$$

F1-Score

$$\text{F1}=2\times\frac{Precision*Recall}{Precision+Recall}$$

These calculations provide a quantitative basis for assessing how well the system distinguishes high-risk lesions while avoiding false alarms.

3.7 Computational Flow for Real-Time Inference

The prediction workflow consists of the following sequence:

- Input image is resized and normalized
- Convolutional layers extract multi-scale features
- Bottleneck blocks compress and refine representations
- Softmax maps the latent vector into class probabilities
- The system selects the highest-probability class
- Output is formatted for user-friendly triage guidance

This streamlined computational structure ensures rapid responses suitable for browser-based medical tools.

4. System Design

The DermaScan platform follows a decoupled, service-oriented architecture to ensure scalability, responsiveness, and ease of maintenance. The system integrates a web-based MERN stack with a lightweight Python microservice dedicated to deep learning inference. By separating user interaction, backend logic, data storage, and model execution, the architecture remains stable and responsive even under high user load.

4.1 Architecture Overview

DermaScan is composed of four core components that work together to deliver real-time dermatological assessment.

The **React frontend** provides an intuitive user interface that allows users to upload images and view prediction results across different devices. Client-side validation ensures that only valid images are submitted, improving system reliability and user experience.

The **Node.js and Express backend** acts as an API gateway, handling request routing, authentication, and secure communication between services. This layer ensures that only verified and properly formatted data reaches the machine learning service[4].
The **MongoDB database** stores non-image metadata such as timestamps, prediction outcomes, and user activity records. To protect privacy, raw medical images are not retained. MongoDB's flexible structure supports efficient data retrieval and future scalability.

The **Python-based ML microservice** hosts the optimized MobileNetV2 model[5]. It performs image preprocessing and generates prediction probabilities through a REST interface. Operating independently from the web server, this service allows model updates without affecting other system components.

Table 1. Architectural Overview

Component	Technology	Role
Frontend (Client)	React.js	User Interface, Image Capture/Upload, Result Visualization (SPA).
Backend (API)	Express.js + Node.js	API Gateway, Authentication, Request Routing, Session Management.
Database	MongoDB	Stores user data, authentication tokens, and analysis metadata (non-image data).
ML Microservice	Python + MobileNetV2	Executes high-speed image preprocessing and deep learning inference via a dedicated REST endpoint.

4.2 Module Breakdown

The **User Management module** handles authentication and session verification using JSON Web Tokens (JWT). Passwords are securely hashed, ensuring data protection while allowing stateless and efficient request validation.

The **Image Processing module** standardizes uploaded images by verifying file integrity, resizing inputs, and normalizing pixel values. This creates consistent input conditions for the deep learning model and improves prediction reliability.

The **Inference Engine** executes the MobileNetV2 model to classify skin images with low latency. It outputs probability scores for each class, enabling effective triage while maintaining computational efficiency.

The **Result Delivery module** converts raw model outputs into user-friendly feedback. Predictions are displayed alongside confidence scores and medical disclaimers, reinforcing that the system provides preliminary guidance rather than clinical diagnosis.

4.3 System Workflow

The operational workflow of DermaScan is illustrated in **Fig. 2**. The process begins when a user uploads a skin image through the web interface. The frontend performs basic validation before securely transmitting the image to the backend. After server-side verification, the image is forwarded to the ML microservice for analysis. The CNN model processes the image and generates prediction probabilities, which are returned to the backend. Finally, relevant metadata is logged in the database, and formatted results are displayed to the user in real time.

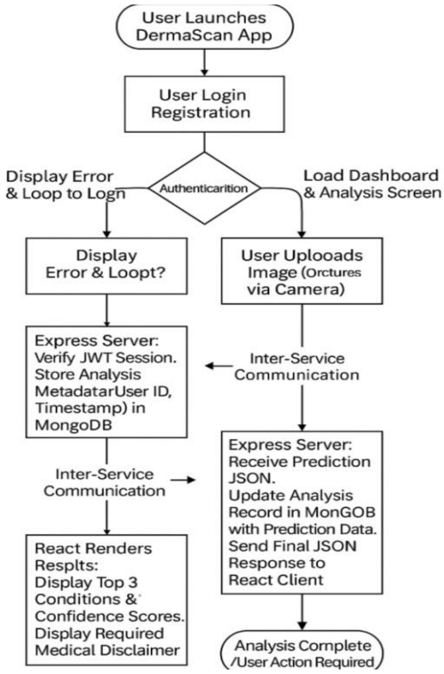

Figure 2. System Workflow Diagram

5. **Results and Discussion**

The performance of DermaScan was assessed based on diagnostic accuracy, computational efficiency, fairness across skin tones, and broader clinical impact, with the aim of validating its suitability for real-time, web-based dermatological triage.

5.1 Diagnostic Performance

The MobileNetV2 classifier achieved an overall accuracy of 89.5%**,** demonstrating reliable performance across varying lighting conditions, image resolutions, and skin tones. The system recorded a high sensitivity of 96.4%**,** which is critical for triage applications where early detection of potentially malignant lesions is essential. In addition, a specificity of 88.5% and an F1-score of 92.3% indicate balanced classification capability, minimizing both missed high-risk cases and unnecessary false alerts. These results confirm that the proposed model provides dependable diagnostic support for real-world teledermatology use (see Table 2**).**

Table 2. Triage Module Efficacy (High-Risk Lesions)

Metric	**Definition**	**MobileNetV2 Result**
Sensitivity	True Positive Rate (Catching Malignancy)	96.4%
Specificity	True Negative Rate (Ignoring Benign)	88.5%
F1 Score (Triage)	Balanced measure of Triage performance	92.3%

5.2 Efficiency and Latency

From a computational perspective, MobileNetV2 demonstrated clear advantages over heavier architectures, making it well suited for browser-based deployment. The model produced predictions in approximately **380 ms** with a compact size of **14 MB**, ensuring near real-time responsiveness even under limited network conditions. In comparison, ResNet50 required significantly higher inference time (≈1800 ms) and memory resources, reducing its practicality for web environments. The reduced latency and efficient design of MobileNetV2 enable smooth user interaction and scalable deployment.

Table 3. Computational Performance Comparison

Model	Average Inference Latency (ms)	Model Size (MB)	Suitability
MobileNetV2	380 ms	14 MB	Web-Accessible
ResNet50	1800 ms	98 MB	High Latency

5.3 Fairness Across Skin Tones

To mitigate bias commonly observed in dermatological AI systems, DermaScan was trained using diverse datasets and targeted color augmentation techniques that reflect variations in melanin levels and imaging conditions. As a result, the model achieved 88.7% accuracy on images of Fitzpatrick skin types V and VI, indicating consistent performance across darker skin tones. This stability across diverse populations highlights the system's ability to generalize effectively and supports ethical, inclusive deployment in real-world healthcare settings.

5.4 Broader Impact

DermaScan offers meaningful benefits in terms of accessibility, scalability, and health awareness by providing fast, web-based preliminary screening without requiring specialized hardware or high-end devices. Its modular architecture serves as a reference framework for future AI-driven healthcare systems, enabling easy expansion and adaptation. By supporting early identification of high-risk lesions and presenting results in a clear, user-friendly manner, the platform encourages timely medical consultation and improves public awareness of dermatological health while maintaining appropriate clinical disclaimers.

6. Conclusion and Future Work

This study demonstrates that DermaScan effectively combines a lightweight deep learning model with a modular MERN-based web architecture to deliver a practical, accessible solution for preliminary dermatological screening. By prioritizing computational efficiency, the system enables real-time assessment even on low-end devices and unstable network connections, removing common barriers associated with native applications or specialized hardware. The inclusion of diverse training data further strengthens the platform by reducing performance disparities across skin tones, reinforcing the importance of fairness in medical AI systems. Overall, the experimental results confirm that DermaScan functions reliably as a rapid triage tool, supporting early identification of potentially concerning skin conditions and offering a scalable foundation for future web-based healthcare applications.

Future enhancements to DermaScan will focus on improving diagnostic precision, clinical utility, and global applicability. Integrating image segmentation techniques such as U-Net or Mask R-CNN could enable more accurate lesion isolation and improve model interpretability through visual overlays. Expanding the range of supported dermatological conditions, potentially using hierarchical classification strategies, would allow the system to better reflect real-world clinical diversity. The addition of secure, real-time communication with certified dermatologists could further bridge the gap between automated triage and professional medical advice, transforming the platform into a hybrid teledermatology service. Finally, increasing dataset diversity across geographic regions, ethnic groups, and imaging conditions will remain a priority to ensure fairness, robustness, and reliable performance for users worldwide.

References

[1] Kabir, A. U., Bhuiyan, T. I., Dhrubo, V., Islam, T., Morshed, M. A., & Reza, A. W. (2024, November). Bridging the Gap: Deep Learning for Equitable Skin Disease Diagnosis Across Diverse Skin Tones. In International Conference on Machine Intelligence and Emerging Technologies (pp. 131-144). Singapore: Springer Nature Singapore.

[2] Uwishema, O., Ghezzawi, M., Charbel, N., Alawieh, S., Roy, S., Wojtara, M., ... & Prasad, M. (2025). Diagnostic performance of artificial intelligence for dermatological conditions: a systematic review focused on low-and middle-income countries to address resource constraints and improve access to specialist care. *International journal of emergency medicine*, *18*(1), 172.

[3] Escalé-Besa, A., Vidal-Alaball, J., Miró Catalina, Q., Gracia, V. H. G., Marin-Gomez, F. X., & Fuster-Casanovas, A. (2024, June). The use of artificial intelligence for skin disease diagnosis in primary care settings: a systematic review. In *Healthcare* (Vol. 12, No. 12, p. 1192). MDPI.

[4] Woodley, A. (2021). Can teledermatology meet the needs of the remote and rural population?. *British Journal of Nursing*, *30*(10), 574-579.

[5] Mustafa, S., Jaffar, A., Rashid, M., Akram, S., & Bhatti, S. M. (2025). Deep learning-based skin lesion analysis using hybrid ResUNet++ and modified AlexNet-Random Forest for enhanced segmentation and classification. *PloS one*, *20*(1), e0315120.

[6] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1-9).