© April 2025| IJIRT | Volume 11 Issue 11 | ISSN: 2349-6002

Skin Disease Detection Using Image Processing

Shahana K[1], Raji N[2], Reshmaa K J[3], Rajasri R K[4]
[1,2,3,4]Computer Science and Engineering, PSNA College of Engineering and Technology, Dindigul, India

Abstract- Dermatological issues impact a significant portion of the global population and may result in severe consequences if not addressed. The principal objective of this project is to develop an online application for the identification of skin problems utilizing image processing techniques. The system enables users to upload images of dermatological issues with a simple front-end developed with HTML, CSS, and JavaScript. The backend employs PHP to process these images, simulating basic image analysis to identify potential skin issues. Notwithstanding the system's scalability and its ability to incorporate robust machine learning algorithms for precise disease classification, this prototype employs a rudimentary detection method. This technique seeks to provide a straightforward tool for early diagnosis and prompt medical consultation through motivation.

I. INTRODUCTION

Dermatological problems impact millions globally and are among the most prevalent health issues, irrespective of age, gender, or nationality. Skin illnesses vary in severity, encompassing minor ailments such as dermatitis and acne, as well as serious issues like melanoma and various types of skin cancer. According to reports from the World Health Organization (WHO) and other international health entities, more than 900 million individuals worldwide experience skin-related illnesses at any given moment. Despite its prevalence, achieving a correct and prompt diagnosis remains difficult, especially in rural or disadvantaged areas where access to specialized dermatological treatment is limited.

The successful treatment of dermatological conditions relies on prompt diagnosis. Early detection of dermatological disorders allows for effective management with minimal intervention. A prolonged diagnosis may lead to complications, the transmission of hazardous infections, or the emergence of more severe disorders. In many circumstances, technology—particularly artificial intelligence (AI) and image processing—provides a logical method for linking patients and clinicians.

Image processing in computer science primarily pertains to the interpretation, manipulation, and analysis of visual data represented as images. Image processing is one of the most crucial diagnostic tools in the medical field. Digital image analysis, encompassing X-rays, MRIs, and dermatological pictures, facilitates pattern detection, region-of-interest segmentation, and disease classification.

Image processing techniques utilized in dermatology enable the extraction of color, texture, shape, and size from digital images of skin lesions or disorders. A multitude of dermatological disorders can be categorized and anticipated utilizing these criteria. In the absence of human knowledge, automated solutions provide advantages in terms of speed, scalability, and assistance, notwithstanding the high diagnostic accuracy of human experts.

Numerous individuals continue to experience inequitable access to affordably priced healthcare, notwithstanding technological advancements. One also impacts dermatology. Numerous individuals neglect dermatological concerns or adhere to erroneous medical guidance due to ignorance or geographical limitations. Manual diagnosis necessitates physical consultations with dermatologists, which may be unfeasible for those residing in distant or undeveloped regions.

Our research seeks to address this by developing a scalable and straightforward web tool for the diagnosis of dermatological conditions. Users may submit an image of a dermatological issue to receive a preliminary diagnostic outcome. The current system employs a simulated detection mechanism for demonstration purposes, thereby establishing a framework for future integration with advanced image processing and machine learning techniques.

The system serves as a prototype or proof-of-concept, demonstrating the integration of web technologies with image processing for medical applications. Website users will have the capability to upload an image from their smartphone, preferably a close-up of the skin damage.

Outcomes can be achieved through an intuitive and tailored interface; a stochastic algorithm or placeholder emulating a diagnostic of a problem provides rapid feedback.

Subsequent backend versions will benefit from the classification and prediction of real-world photos with Python-based machine learning. A database system can store results, scan histories, user information, and more.

The increasing prevalence of cellphones and internet access in distant regions creates new prospects for digital healthcare solutions. This method promotes client engagement with professionals through first evaluations. While it is intended as an auxiliary instrument for early detection, awareness, and education, it cannot supplant medical experts.

The curriculum provides a robust basis for researchers, developers, and students interested in creating real-time AI-driven healthcare solutions. It illustrates the dynamic integration of web development, backend programming, and image processing inside a unified platform.

Integration with Machine Learning Techniques: Python-based artificial intelligence methodologies can be employed to classify authentic photographs via REST APIs.

Image Preprocessing: To augment predictive accuracy, implement filters or alternative preprocessing methods to improve image quality.

Mobile Responsiveness: Enhance accessibility by optimizing the user interface for mobile users.

User Authentication: Provide users with secure login alternatives to monitor their diagnostic history.

Preserve photos and projections in your database for future utilization or comprehensive analysis.

II. RELATED WORKS

A multitude of researchers and developers have devised methodologies for diagnosing skin diseases through the application of artificial intelligence and image processing techniques. Users of several smartphone applications, such as SkinVision, can capture photographs of their skin for analysis by trained machine learning algorithms. Typically employing Python, TensorFlow, or OpenCV, these advanced algorithms assess skin characteristics such as hue, form, and texture. Other initiatives focus on enhancing the accessibility of skin tests through online platforms. Our project is a streamlined application enabling users to upload an image and receive a simulated result utilizing HTML, CSS, JavaScript, and PHP. Although it currently does not utilize genuine artificial intelligence, it is designed for future advancement and ease of use based on analogous concepts.

III. SYSTEM ARCHITECTURE

Overview

The system primarily has three components:
Frontend: Engineered to enable web browser users to interact with the system; the user interface facilitates this.
Backend: The location where a simulated result is generated after analyzing image submissions.
Optional database: Intended for future usage to save user data, supplied images, and detection results.
The architectural design follows this sequence:
A picture is uploaded through the online interface. The image is transmitted to the PHP backend.
The backend generates a forecast subsequent to image processing.
The user is presented with the result following transmission.

Frontend
HTML, CSS, and JavaScript are the employed technologies. The objective is to deliver a responsive and seamless user experience, enabling users to upload a skin picture. Examine the outcomes subsequent to processing.
Features: An HTML form for the selection and upload of images. Utilize JavaScript to manage image viewing and form submission without necessitating a page reload. Utilize CSS to enhance the page's aesthetics and user experience.

Backend
PHP serves as the technological framework.
Objective: Supervise photographic inputs and provide a simulation for skin disease diagnosis.
Role:
A POST request is utilized to obtain an image from the frontend.
The image must be stored locally, such as in an upload directory.
Select a fake disease name randomly from a list to simulate a diagnosis.
Transmit the results to the front end.
The simulation yielded the result: "The dermatological condition resembles eczema."
APIs may be employed to link this backend to an actual AI model if required.

Database (Utilized Prospectively)
Currently, MySQL is an elective technology.
Objective (if executed subsequently):
Preserve the paths or names of the submitted images. Monitor all user submissions and outcomes. Assistance for surveillance and user verification.

IV. IMPLEMENTATION DETAILS

User Interface (UI)
Technologies Used: HTML, CSS, JavaScript
Purpose: To create an easy-to-use interface where users can upload an image and see results.
Main Features:
A photo upload interface that enables users to select an image that most accurately represents their skin condition.
The selected image is displayed in a preview window before submission. A submit button that employs JavaScript and AJAX to transmit the picture to the server without necessitating a page reload.
A section dedicated to exhibiting the symptoms of the simulated dermatological illness. JavaScript facilitates interactivity through image previews and data transmission to the backend, whilst CSS enhances the creation of a responsive and structured layout.

API Integration
Current Configuration: No external API is utilized at present. PHP code manages image uploads subsequent to the random selection of a dermatological ailment (e.g., psoriasis, acne, eczema, etc.) from a predetermined list.
Example PHP flows:
1. Receive image via POST request.
2. Move the uploaded file to the uploads/ folder.
3. Generate a fake result.
4. Send result back as a JSON response.

AJAX facilitates interaction between the frontend and backend interfaces. The frontend JavaScript executes an asynchronous request to a PHP file, such as predict.php, without reloading the page, and receives the response.
Future Scope:
REST APIs facilitate the integration of a Python-based artificial intelligence model. The Python server could generate an accurate prediction using the supplied image from PHP.

Database Management
Database Used: MySQL (for future implementation)
Purpose:
Store uploaded image file names/paths.
Save detection results.
Store timestamps or user information (if login is added later).

Table Example (image_records):

ID	Image_Name	Result	Upload_Date
1	skin123.jpg	Eczema	2025-04-08 12:00PM

PHP + MySQL Integration (if used):
Use mysqli_connect() to connect to the database.
Use INSERT INTO to save image and result data after processing.
Use SELECT to show history if needed.

V. SECURITY CONSIDERATIONS

All websites require security measures, particularly regarding picture uploads and user input. Despite the project's simplicity, certain security precautions must be implemented to safeguard the system against misuse, faults, or assaults.

File Upload Security

File Type Verification: Only image files (.jpg, .jpeg, and .png) are authorized. Utilize PHP's mime_content_type() function or examine the file's extension to determine its type.

File Size Limitation: Establish a maximum file size to avert excessive uploads (e.g., a 2MB cap).

File Name Sanitization: To avert overwriting or the execution of malicious scripts, rename uploaded files or employ uniqid().

Upload Directory Protection: Store uploaded files in a separate directory (e.g., /uploads) and employ server setup or .htaccess to restrict direct access.

Input Validation

Unforeseen or detrimental information may be transmitted even when users merely upload images; it is imperative to authenticate and sanitize any data obtained from the frontend. Mitigate code injection and cross-site scripting (XSS) threats utilizing PHP functions such as filter_input() and htmlspecialchars().

Cross-Site Scripting (XSS)

User input that is not properly escaped should not be directly displayed on the webpage. Ensure all data is sanitized prior to presentation. Exercise caution when employing JavaScript for dynamic content and avoid incorporating unprocessed user data into HTML.

Cross-Site Request Forgery (CSRF)

Incorporating a CSRF token in forms or AJAX requests guarantees that submissions originate solely from your website. A tailored PHP configuration can generate and validate a token using $(SESSION), despite most PHP installations featuring inherent CSRF protection.

Server-Side Checks

Avoid relying solely on JavaScript validation. Always implement server-side validation in PHP, as users can circumvent frontend checks. Examine the inputs to the PHP script, encompassing file type and size.

Directory Traversal and Path Injection

Avoid utilizing paths or file names derived directly from user input. Prohibit users from selecting file storage locations—consistently utilize designated directories and secure nomenclature.

HTTPS and Secure Hosting

Ensure that HTTPS is implemented on your website to safeguard user data during online transmission. To prevent users from accessing or modifying server files, utilize secure hosting and establish appropriate file permissions.

VI. EVALUATION AND RESULT

The product underwent testing by uploading multiple skin images via the web interface to evaluate the system's usability. The primary objectives were to ascertain whether the image uploads correctly, is processed by the PHP backend, and provides the user with a simulated result.

Test Results:

The functionality for image uploads operated effectively. Individuals may select and evaluate a skin image prior to submission. Upon the successful download and storage of the snapshot, the PHP software indicated a potential dermatological condition: eczema.

Speed: An immediate response was achieved owing to the absence of advanced machine learning or complex processing.

JavaScript and AJAX enable users to view results on the same page without reloading

Test Cases:

Test Case	Expected Result	Actual Result	Status
Upload valid image	Image uploaded, result shown	Works as expected	Passed
Upload non-image file	System blocks the upload	Upload blocked	Passed
Upload large image	Error/warning shown for size limit	Size limit works	Passed
Refresh page	No issues, system resets properly	Works fine	Passed

Overall Performance:
The system, while not offering skin condition diagnostics, effectively elucidates the process of image upload and response retrieval; it functions as a basic image-based prediction tool. A genuine machine learning model or database support facilitates additional enhancements.

VII.DISCUSSION

This project illustrates the development of a fundamental web application that simulates the diagnosis of dermatological disorders utilizing HTML, CSS, JavaScript, and PHP. The system effectively demonstrates the procedure of importing images and generating outcomes through a fabricated backend, rather than employing authentic medical analysis or machine learning. The primary objective was to create a functional prototype featuring an intuitive user interface and an operational backend. The system is user-friendly, uncomplicated, and offers a robust basis for future enhancements such as database storage, real image processing, and artificial intelligence integration.

VIII. FUTURE WORKS

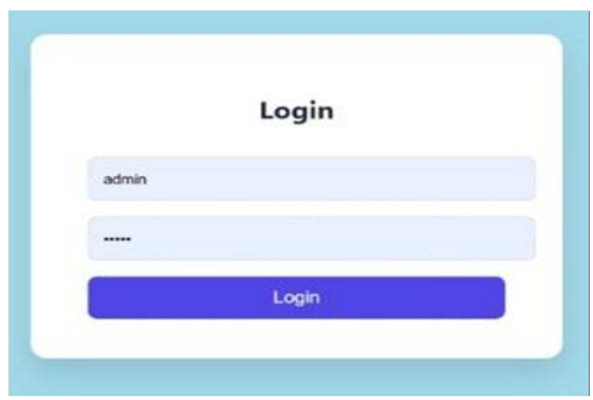
AI-Based Skin Disease Detection
A genuine machine learning model can be included to assess skin scans and improve the precision of illness prediction.
TensorFlow in Python or an external AI API interface may be employed for this objective. The system can produce more intelligent and reliable results, similar to genuine diagnostic support tools.

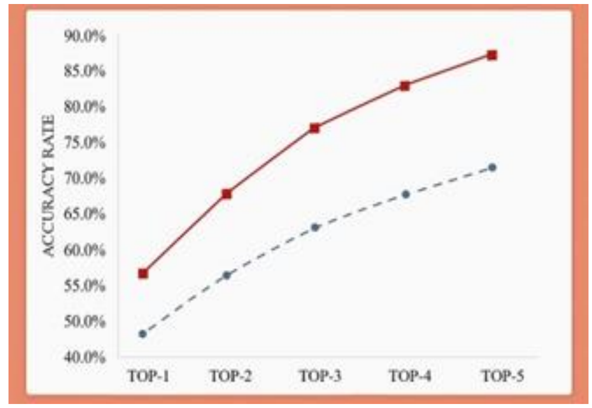
Multimedia Support
Ultimately, users may have the capability to upload films and images of dermatological conditions, like rashes or wounds that necessitate prolonged healing. This would enhance analytical input and improve detection accuracy.

Mobile Application
The implementation of advanced technologies to develop a mobile version of the system is facilitated by React Native or Flutter. Individuals will find it more easy and effortless to capture images directly from their phones and immediately view the results.

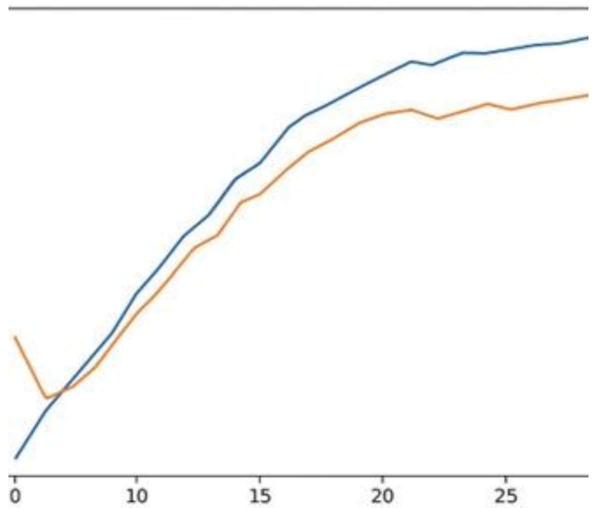
Enhanced Analytics
A dashboard may be developed for administrators or physicians to access statistics, including: Number of uploads; Most prevalent dermatological disorders
Trends across time: Analytics can enhance healthcare decision-making and monitor illness patterns.

IX. RESULT AND OUTPUT





Red line represents the graph of our accurate data of skin disease detection. Dotted line represents the graph by other works



Blue line represents the graph of our accurate data of skin disease detection. Orange line represents the graph by other works

X. CONCLUSION

The study demonstrates a key online method for identifying skin disorders via image input and simulated analysis. The application, developed with HTML, CSS, JavaScript, and PHP, allows users to interact with the system and receive a simulated outcome depending on their photos. While it does not perform genuine medical diagnostics, it provides a solid foundation for future improvements. The system is intuitive, suitable for beginners, and can be augmented with features such authentic AI integration, mobile application compatibility, and extensive analytics. This research illustrates how technology may enhance health awareness and enable early diagnosis in a clear and accessible way.

REFERENCE

[1] Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., & Thrun, S. (2017). "Dermatologist-level classification of skin cancer with deep neural networks." *Nature*, 542(7639), 115–118. https://doi.org/10.1038/nature21056

[2] Tschandl, P., Rosendahl, C., & Kittler, H. (2018). "The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions." *Scientific Data*, 5(180161). https://doi.org/10.1038/sdata.2018.161

[3] Codella, N., Nguyen, Q. B., Pankanti, S., et al. (2017). "Deep learning ensembles for melanoma recognition in dermoscopy images." *IBM Journal of Research and Development*, 61(4/5), 5:1–5:15. https://doi.org/10.1147/JRD.2017.2708298

[4] Kaggle. (n.d.). "Skin Cancer MNIST: HAM10000 Dataset." https://www.kaggle.com/datasets/kmader/skin-cancer-mnist-ham10000
A commonly used dataset for training and testing skin disease detection models.

[5] Kumar, P., & Munjal, B. (2020). "Image processing and machine learning techniques for skin disease detection: A review." *International Journal of Advanced Research in Computer and Communication Engineering*, 9(6), 34–38.

[6] TensorFlow. (n.d.). "Image classification with TensorFlow." https://www.tensorflow.org/tutorials/images/classification
A popular framework for building AI models, including for medical image classification.

[7] ISIC Archive – International Skin Imaging Collaboration
https://www.isic-archive.com
A global repository of skin lesion images used in AI research.

[8] Raj, A., & Srinivas, K. (2020). "Skin Disease Detection using Image Processing and Machine Learning." *International Journal of Innovative Research in Science, Engineering, and Technology*, 9(6), 2734–2740. https://doi.org/10.15680/IJIRSET.2020.0906177

[9] Li, J., Zhang, D., & Xu, Z. (2019). "Automated skin lesion classification based on deep convolutional neural networks." *Journal of Healthcare Engineering*, 2019, 1–9. https://doi.org/10.1155/2019/9275632

[10] Gul, A., & Javed, M. (2018). "Skin Disease Classification using Machine Learning Algorithms: A Comparative Study." *Journal of Computer Science and Technology*, 33(4), 745–758. https://doi.org/10.1007/s11390-018-1842-4

[11] McKinney, S. M., Sieniek, M., Godbole, V., et al. (2020). "International evaluation of an AI system for breast cancer screening." *Nature*, 577(7788), 89–94. https://doi.org/10.1038/s41586-019-1799-6

[12] Attia, A., Elham, S., & Afifi, H. (2017). "Skin disease detection using image processing." *International Journal of Computer Applications*, 166(4), 13-17. https://doi.org/10.5120/ijca2017913862

[13] Hamarneh, G., & Shams, S. (2015). "Automated analysis of skin lesions in dermoscopy images using supervised learning." *Journal of Medical Imaging*, 2(3), 031002. https://doi.org/10.1117/1.JMI.2.3.031002

[14] Zhang, L., & Wang, H. (2020). "Deep learning models for skin disease classification and detection." *Computerized Medical Imaging and Graphics*, 82, 101704. https://doi.org/10.1016/j.compmedimag.2020.101704

[15] Yang, M., & Wu, X. (2019). "Skin lesion classification based on convolutional neural network with efficient data augmentation." *Journal of Artificial Intelligence in Medicine*, 98, 75–85. https://doi.org/10.1016/j.artmed.2019.02.001