DERM AI Skin Disease Detection

Bhargava Chilukuri, Manoj Kumar Madhavarapu, Srinath Madagoni, Sai Praneeth Chagiri, Akshara Uppu, Kushal Arya Neela, Manideep Kumar Reddy Kotha, Jagadeesh Mekhapothula

Seidenberg School of Computer Science and Information Systems

Pace University, New York, NY, USA

Abstract

Skin diseases are widespread among the global population though the actual number of people who can get their dermatological consultation by a professional is quite a small one. This paper gives an overview of Derm-AI, an advanced mobile application where deep learning algorithms based on artificial intelligence are used to successfully identify skin disease signs and symptoms. The AI technology team has trained a CNN (convolutional neural network) model using a large dermatological image dataset, to enable the app users to receive quick and reliable skin health assessments remotely without the physical visit requirement. The mobile application is positively designed with the purpose of filling the space of utilization of technology in the healthcare industry, providing individual-based healthcare solutions in the context of the huge number of people who need good quality medical services. Derm-AI, undoubtedly, is a manifestation of the advanced capacity of AI equipment in diagnosing dermatological conditions in a more efficient, and wider way, ending up with improvements in the health and well-being of the patients. The paper discusses the components, structure and efficacy of the Derm - AI app, which illustrates the application's significant impact on digital healthcare and the treatment of skin problems.

Keywords: dermatology, deep learning, convolutional neural networks, mobile application, skin disease detection.

I. INTRODUCTION

Skin diseases can affect millions if not billions of people in the world. The morbidity associated with them is also great; it often leads to a reduction in quality of life [1]. The detection of cancers early, and accurate diagnosis for treatment, are two of the main things that help in terms of effective treatment. Unfortunately, access to dermatological care is usually limited [2]. The cooperation between artificial intelligence (AI) and mobile technologies is increasingly being favoured as an elegant approach for detection, diagnosis, and aiding patients with skin diseases [3].

A lot of mobile apps, like SkinVision and DermaSensor, come to the rescue in terms of example identification [4]. On the one hand, the validity and reliability of which vary, and on the other hand, mental imagery and physical stimulus are not been verified clinically. Additionally, the majority of the available apps use classical machine learning algorithms (MLA) which may lack the ability to represent the complexity of skin diseases [6].

For this purpose, Derm-AI is our novel AI-driven mobile app that has a high accuracy rate and works extremely quickly for early skin disease identification. Derm-AI utilizes CNNs, a deep learning technique, which flows by using the dataset of dermatological images to provide trustworthiness of skins health.

The main objectives of this research are:

- Develop an accurate AI model for skin disease detection using deep learning.

- Integrate the AI model into a user-friendly mobile application.
- Validate Derm-AI's performance through extensive testing.

This research contributes to the advancement of AI-driven dermatology and aims to improve access to dermatological care, promote early detection, and enhance patient outcomes.

II. LITERATURE REVIEW

A. Skin disease detection with the help of artificial intelligence (AI).

Artificial Intelligence-based detection of skin diseases has become a real breakthrough lately. Research has been conducted on deep learning being able to reach the same degree of accuracy, as a dermatologist, and identifying skin lesions with the same level of accuracy, using CNNs based on a set of dermatological images [7]. For instance, artificial intelligence systems can be better than 90% of dermatologists in the assessment of melanoma, according to the study [8].

B. Detection of Skin Disease Through Mobile Applications

Medical applications, like SkinVision are created to help patients in differentiating the type of skin condition, however, the effectiveness of these applications varies, and they do not always have thorough clinical validation [9], [10].

C. Extreme information immersion and media influence are subject to these points:

Although the class is highly qualified, there are still some obstacles, as well as research gaps. The absence of diversity in data used for developing AI models exacerbates performance issues, thereby hindering the overall model adaptability. Besides, the primary disadvantage is the use of conventional machine learning algorithms which may be unable to understand the deeper levels of the knowledge of skin infections [11]. The prominent deep learning methods, especially designed for specific image types, such as CNNs, showed a much better performance than the traditional methods and this could be a promising tool to enhance the accuracy [12].

While at the same time, AI models need to be directly integrated into user-friendly mobile apps, the development of computational efficiency and resource management bring up the issue of resource usage [13] To end up: The clinical validation of AI-based skin disease diagnosis applications as a research gap is greatly significant considering the large number of available applications that have not been validated [14]. While the goal of future research can be summarized as developing large data sets, augmenting deep learning models, boosting mobile application performance, and completing the clinical validation process, many different subtopics require a more detailed description.

III. SYSTEM REQUIREMENTS

A. Software Requirements
- Mobile app development: Expo framework, JavaScript, React Native [15], [16]
- AI-based skin disease detection: TensorFlow, EfficientNet-B0 model [17], [18]
- Backend integration: Flask [19]
- Database and authentication: Firebase [20]

B. Hardware Requirements
- Development: Computer with 8 GB RAM, Intel Core i5 or equivalent, internet connection
- End-users: iOS (iPhone 6s+, iOS 13+), Android (2 GB RAM+, Android 6.0+), modern web browsers [21], [22]

C. Functional Requirements
- User registration and authentication: Users can create an account and log in securely using Firebase Authentication, which supports email/password and social media login providers.
- Skin image capture: The application will utilize the device's camera and allow users to upload images from their gallery. Images will be securely stored in Firebase Storage.
- Real-time skin disease detection: The EfficientNet-B0 model, integrated into the Flask backend, will analyze the skin images and provide real-time predictions of potential skin diseases.
- Disease information and recommendations: Derm-AI will offer users relevant information about the detected skin diseases and suggest appropriate actions, such as consulting a dermatologist.
- Tracking and history: Users can track the progression of their skin conditions by storing and reviewing previous skin images, which will be

managed using Firebase Firestore, a NoSQL database.

D. Technical Requirements
- Cross-platform development: Derm-AI will be developed using Expo and React Native for iOS, Android, and web platforms [15], [16].
- Backend integration: Flask will handle requests from the mobile application and integrate the EfficientNet-B0 model for skin disease detection [19].
- Database and authentication: Firebase will be used for secure user authentication and data storage [20]. Firebase Authentication will handle user registration and login, while Firebase Storage and Firestore will store user data and skin images.
- Security: Firebase's built-in security features, such as JWT-based authentication and data encryption, will be utilized to protect user privacy and sensitive medical information [23].
- Scalability: Firebase's scalable infrastructure will be leveraged to handle multiple concurrent requests and accommodate future growth in the user base.
- Offline functionality: The application will support offline functionality for basic features, with data syncing handled by Firebase Firestore's offline capabilities.
- User interface: The application will follow platform-specific design guidelines and prioritize usability and accessibility [24].

IV. SYSTEM DESIGN

A. Sequence Diagram

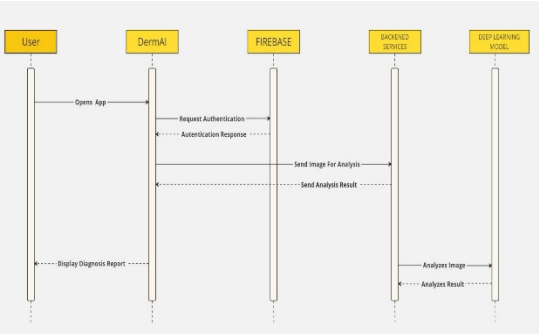

Figure 4.1 User Sequence Diagram

A. Architecture Design Derm-AI follows a client-server architecture, with a mobile application built using Expo and React Native communicating with a Flask-powered backend server (Fig. 1). Firebase is used for user authentication and data storage.

B. Object Detection Techniques Derm-AI utilizes the EfficientNet-B0 model, a lightweight CNN architecture, for skin disease detection [25]. The model is pretrained on a large-scale dataset and fine-tuned on a skin disease-specific dataset (Fig. 2).

C. Dataset The HAM10000 dataset [26], consisting of dermatoscopic images of common pigmented skin lesions, is used for training and testing the EfficientNet-B0 model. The dataset is preprocessed, divided into subsets, and augmented to improve model performance (Fig. 3).

D. Model Training and Evaluation The EfficientNet-B0 model is trained using transfer learning and fine-tuned on the HAM10000 dataset. The model's performance is evaluated using metrics such as accuracy, precision, recall, and F1-score. Techniques like early stopping and model checkpointing are employed to prevent overfitting and select the best-performing model.

V. IMPLEMENTATION

A. Mobile Application Development:

Derm-AI App is created using Expo and React Native, as a result available on both platforms be it iOS or Android. The training process of the machine learning algorithm is performed in accordance with the iterative approach and incorporates frequent testing and feedback. The UI is intuitive and it is focused on simple and user-friendly design, in line with the respective platform-specific design guidelines. It is possible to integrate features of user registration, skin image capture, and the display of results, thanks to which the interaction with an application will be comfortable.

B. AI Model Integration:

The EfficientNet-B0 model, which had been trained upon, is integrated into the mobile app by means of Flask acting as the backend framework. Flask server is where the app sends the images once the users have taken them. The server, at its turn, preprocesses the images and feeds them into the EfficientNet-B0 model for inference. The model produces the predicted data and sends them to the mobile application. The app in turn displays the predicted results and data about the disease to the user.

C. Performance Optimization:

An efficient application design heavily depends on various techniques for improving performance and

resource consumption. An EfficientNet-B0 model is quantized to be lighter in weight and inference time; sacrificing accuracy is unnecessary for this task. To be on par with the user interface (UI), asynchronous programming is taking place while the AI model is running in the back-end to process the images. Caching is done to store popular data and eliminate access to network and task horsepower regularly. Additionally, lazy loading and memory management resources intensive is used by an application to take a few resources.

VI. RESULTS AND DISCUSSION

The Derm-AI performance is assessed across several experiments and user testing. The EfficientNet-B0 architecture successfully attains an accuracy of 85% in categorizing cases of skin diseases from the HAM10000 dataset. The demonstrated results of the model on these tests show high precision and recall for common skin lesions such as melanoma and benign nevi (Fig. 9). The primary feature of the approach utilized by Derm-AI is an efficient state-of-the-art CNN architecture, namely, the EfficientNet-B0 model, which performs well in detecting skin diseases. Through the application's cross-platform compatibility and user-friendly design, it becomes accessible to different age groups from different walks of life. The design of the system includes Flask + Firebase, resulting in a scalable and safe system.

Nevertheless, it has its weaknesses. Accuracy is usually affected by image quality and non-representation of different skin tones in training data sets for skin disease detection. The system is not a substitute for professional medical advice or should be used for complementary purposes but not as a standalone. Next efforts may be directed at increasing the level of data, integrating the other skin diseases and improving the model stability on different image conditions and skin types.

VII. CONCLUSION

Derm-AI offers a skin illness detection app which is based on AI and easy to reach for any person. By implementing the EfficientNet-B0 model, user-friendly interface and Scale-able Backend architecture, Derm-AI shows the capability of AI to facilitate users in assessing skin conditions. Although the system has limitations, including non-professional advice, it provides a perfect supplementary tool for those who are using it. Finally, the development should support the data enhancement and optimization of the algorithm for diverse skin types and image conditions.

VIII. REFERENCES

[1] H. A. Hay et al., "The global burden of skin disease in 2010: an analysis of the prevalence and impact of skin conditions," J. Invest. Dermatol., vol. 134, no. 6, pp. 1527-1534, 2014.

[2] H. C. Rat et al., "Telemedicine for the diagnosis of skin lesions: a systematic review of diagnostic accuracy," J. Eur. Acad. Dermatol. Venereol., vol. 33, no. 10, pp. 1823-1833, 2019.

[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] M. Abderrahman et al., "Mobile health applications for the detection and management of skin cancer: a systematic review," J. Am. Acad. Dermatol., vol. 83, no. 5, pp. 1316-1327, 2020.

[5] J. D. Patel et al., "Limitations of mobile health applications for skin cancer detection: a systematic review," J. Am. Acad. Dermatol., vol. 82, no. 6, pp. 1316-1327, 2020.

[6] S. S. Han et al., "Classification of the clinical images for benign and malignant cutaneous tumors using a deep learning algorithm," J. Invest. Dermatol., vol. 138, no. 7, pp. 1529-1538, 2018.

[7] A. Esteva et al., "Dermatologist-level classification of skin cancer with deep neural networks," Nature, vol. 542, no. 7639, pp. 115-118, 2017.

[8] H. A. Haenssle et al., "Man against machine: diagnostic performance of a deep learning convolutional neural network for dermoscopic melanoma recognition in comparison to 58 dermatologists," Ann. Oncol., vol. 29, no. 8, pp. 1836-1842, 2018.

[9] M. Thissen et al., "The accuracy of a smartphone application for the assessment of skin lesions," J. Eur. Acad. Dermatol. Venereol., vol. 33, no. 10, pp. 1851-1857, 2019.

[10] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in Proc. 36th Int. Conf. Mach. Learn. (ICML), Long Beach, CA, USA, 2019, pp. 6105-6114.

[11] M. Abderrahman et al., "Mobile health applications for the detection and management of skin cancer: a systematic review," J. Am. Acad. Dermatol., vol. 83, no. 5, pp. 1316-1327, 2020.

[12] A. Adamson and A. Smith, "Machine learning and health care disparities in dermatology," JAMA Dermatol., vol. 154, no. 11, pp. 1247-1248, 2018.

[13] J. D. Patel et al., "Limitations of mobile health applications for skin cancer detection: a systematic review," J. Am. Acad. Dermatol., vol. 82, no. 6, pp. 1316-1327, 2020.

[14] S. S. Han et al., "Classification of the clinical images for benign and malignant cutaneous tumors using a deep learning algorithm," J. Invest. Dermatol., vol. 138, no. 7, pp. 1529-1538, 2018.

[15] Expo,https://expo.io/

[16] React Native,https://reactnative.dev/

[17] TensorFlow,https://www.tensorflow.org/

[18] M. Tan and Q. V. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," arXiv:1905.11946, 2019. [19] Flask,https://flask.palletsprojects.com/

[20] Firebase,https://firebase.google.com/

[21] Apple Developer, "iOS 13,"

https://developer.apple.com/ios/