DermaDetect – AI-Based Tool For Preliminary Diagnosis Of Dermatological Manifestations.

Anusha Shekar Gouda Malipatil, Sneha S Ballari, Thaksha Prabhakar, Sneha A, Mr. Ramesh T
*[1]School of Computer Science and Engineering, Presidency University, Bangalore, Karnataka, India
anushamp9353@gmail.com[1]
*[2]School of Computer Science and Engineering, Presidency University, Bangalore, Karnataka, India
snehasballari01@gmail.com[2]
*[3]School of Computer Science and Engineering, Presidency University, Bangalore, Karnataka, India
thakshaprabhakar08@gmail.com[3]
*[4]School of Computer Science and Engineering, Presidency University, Bangalore, Karnataka, India
snehaanand26208@gmail.com[4]
*[5]Assisstant Professor, Presidency University, Bangalore, Karnataka, India
ramesh.t@presidencyuniversity.in[5]

ABSTRACT

Skin diseases are among the most common health issues worldwide, yet early and accurate diagnosis is often limited by lack of access to dermatologists. To address this, we developed DermaDetect, a web-based deep learning application that classifies skin disease images into five categories: Actinic Keratosis, Alopecia, Chondrodermatitis Nodularis, Keloids, and Molluscum Contagiosum. The system is powered by a fine-tuned MobileNetV2 model trained on labeled dermatology images. It provides instant predictions with confidence scores through a user-friendly web interface built using Gradio. This project aims to support early detection and reduce the diagnostic burden in remote or resource- constrained areas.

Keywords: Deep Learning, Skin Disease Detection, MobileNetV2, Image Classification, Gradio Web App, Dermatology

I. INTRODUCTION

Skin disorders are among the most frequent health problems worldwide, affecting people of all ages, genders, and skin types. From minor issues like acne and rashes to more serious conditions such as skin cancer or chronic infections, dermatological problems can significantly impact a person's quality of life if not diagnosed and treated early. According to the World Health Organization (WHO), nearly 900 million people suffer from some form of skin disease at any given time. Unfortunately, due to lack of awareness and limited access to healthcare, especially in rural or economically backward areas, many cases go undiagnosed or misdiagnosed.

Traditionally, diagnosis of skin diseases relies on the expertise of dermatologists who visually inspect the affected area. However, in countries like India, where there is an uneven doctor-to-patient ratio and a scarcity of skin specialists in remote regions, this method alone is not sufficient. Long wait times, the need for travel, and high consultation costs are additional barriers. This is where artificial intelligence (AI), particularly deep learning, can play a transformative role by assisting in the early detection and classification of skin conditions using automated image analysis.

In recent years, the field of computer vision has achieved significant breakthroughs due to Convolutional Neural

Networks (CNNs), which excel at recognizing patterns in visual data. These models have been successfully applied to a variety of medical imaging problems, including X-ray classification, tumor detection in MRIs, and now increasingly, skin disease diagnosis from images. Among various CNN architectures, MobileNetV2 has emerged as a lightweight and efficient model suitable for deployment on low-resource devices like mobile phones, tablets, and embedded systems

In this project, we propose a deep learning-powered solution named DermaDetect, which leverages the MobileNetV2 architecture to classify skin disease images into five specific categories: Actinic Keratosis, Alopecia, Chondrodermatitis Nodularis, Keloids, and Molluscum Contagiosum. These diseases were selected based on their visual distinctiveness and frequency in public datasets. By fine-tuning MobileNetV2 on labeled dermatological image data, the system is able to predict the most likely disease from an uploaded skin image with high accuracy.

To ensure ease of use and accessibility, we built a web-based interface using Gradio, a Python library that allows users to upload images and receive real-time predictions with corresponding confidence scores. This not only makes the system user-friendly but also suitable for deployment in community health programs, awareness drives, and rural clinics.

Our goal is not to replace doctors, but to support early-stage screening, assist healthcare professionals, and raise awareness among patients about potential skin conditions. With the increasing adoption of telemedicine and mobile healthcare solutions, DermaDetect can act as a stepping stone toward democratizing skin health diagnosis for everyone, regardless of geographic or economic barriers.

II. RELATED WORK

In recent years, the application of artificial intelligence and deep learning in healthcare, particularly in dermatology, has grown significantly. Numerous researchers have focused on building systems that use machine learning models to detect and classify skin diseases from images, aiming to improve early diagnosis, reduce the burden on healthcare providers, and make dermatological services more accessible.

One of the most influential works in this domain was carried out by Esteva et al. (2017), who trained a deep convolutional neural network (CNN) to classify skin cancer at a level comparable to dermatologists. Using a dataset of over 129,000 clinical images covering more than 2,000 diseases, they demonstrated that CNNs could achieve dermatologist-level performance in distinguishing between malignant and benign lesions. This landmark study provided strong evidence of the potential of deep learning in skin disease classification.

Following this, the HAM10000 dataset became a benchmark for skin lesion classification tasks. Researchers have used architectures like ResNet50, DenseNet121, and InceptionV3 to classify common skin diseases such as melanoma, basal cell carcinoma, and benign keratosis. Many of these models achieved high accuracy but were computationally intensive, limiting their use in real-time or resource-constrained environments.

To address the issue of computational efficiency, researchers began adopting lighter architectures such as MobileNet, SqueezeNet, and EfficientNet. MobileNetV2, in particular, gained popularity for its balance between performance and speed. It has been used in mobile and embedded dermatology applications for real-time diagnosis. In one study, MobileNetV2 was fine-tuned to classify a smaller subset of dermatological conditions and achieved over 90% accuracy on a custom dataset with limited training data.

Researchers have also explored hybrid models that combine CNNs with traditional image processing techniques. For example, some studies used color histogram features along with CNN-extracted features to improve classification accuracy. Others applied segmentation techniques to isolate lesions before classification, leading to more precise predictions.

Another active area of research has been the integration of explainable AI (XAI) methods. Techniques such as Grad-CAM and saliency maps have been used to visualize which parts of the image influenced the model's decision. This not only improves model transparency but also builds trust among healthcare professionals.

In addition to classification, deep learning has been used for lesion segmentation, severity estimation, and progression tracking. Researchers have used U-Net and Mask R-CNN for segmenting skin lesions, enabling better feature extraction and diagnosis.

Recent studies have also emphasized the importance of demographic diversity in datasets. Many public dermatology datasets are biased toward lighter skin tones, leading to potential misdiagnoses in people with darker skin. Researchers are now working to create balanced datasets that represent various skin types and conditions more equitably.

The user experience is another critical component. Projects like DermaSnap and SkinApp have developed mobile applications that allow users to take photos of skin lesions and receive instant feedback. However, these apps often lack interpretability, multi-disease classification, or high accuracy.

Our work builds upon this foundation by offering a focused, efficient, and user-friendly system that classifies five specific skin diseases using a MobileNetV2 model. Unlike many existing solutions, DermaDetect integrates a real-time prediction system with a web interface powered by Gradio, making it accessible to both clinicians and non-expert users. The system is designed to support early diagnosis in low- resource settings while maintaining a high level of accuracy and performance.

III. WORKING METHODOLOGY

Our proposed system, DermaDetect, follows a clear and structured pipeline to classify images of skin diseases into five categories. This methodology ensures that the entire process—from receiving an image to displaying the result—is accurate, fast, and easy for users to interact with. Below is a comprehensive explanation of each step involved in building and running the system.

3.1 Data Collection and Preprocessing

We began with the collection of dermatological images representing five specific diseases: Actinic Keratosis, Alopecia, Chondrodermatitis Nodularis, Keloids, and Molluscum Contagiosum. Data was sourced from publicly available datasets or custom annotated sources. Each image underwent a series of preprocessing steps:

- Resizing: All images were resized to 224x224 pixels, the input size required by MobileNetV2.
- Center Cropping: Ensures the central portion of the lesion is used to standardize input and avoid background noise.
- Normalization: Pixel values were normalized using ImageNet's mean and standard deviation to match pretrained model expectations
- Tensor Conversion: Images were converted into PyTorch tensors for compatibility with model input.
- Data Augmentation: Random flips, rotations, and brightness changes were applied to expand the dataset and improve generalization.

3.2 Model Selection and Customization

We selected MobileNetV2 for its efficiency and real- time inference capabilities. The model was initially trained on ImageNet and then fine-tuned on our dataset. Key modifications include:

- Freezing Base Layers: To preserve pre- learned features while reducing training time
- Modifying the Classifier Head: The original final layer (for 1000 ImageNet classes) was replaced with a new fully connected layer tailored to output five class scores.
- Dropout Layer Addition: A dropout layer (p=0.2) was added before the final classifier to reduce overfitting.

3.3 Testing and Optimization

Training the model involved multiple stages and checkpoints to monitor progress.

- Loss Function: CrossEntropyLoss was used due to the multi-class nature of the task.
- Optimizer: Adam optimizer with a learning rate of 0.001 enabled adaptive and efficient training.
- Epochs and Batch Size: The model was trained for 25 epochs with a batch size of 32.
- Validation Strategy: After each epoch, the model was evaluated on a validation set to monitor generalization.
- Early Stopping: Training stopped if no improvement was seen in validation loss for five consecutive epochs.

3.4 Evaluation Metrics

To ensure that the model performs well in all categories, we used several evaluation metrics.

- Accuracy: Measures overall correct predictions.
- Precision & Recall: Evaluated per class to understand false positives and false negatives.
- F1-Score: Harmonic mean of precision and recall for balanced performance.
- Confusion Matrix: Helps visualize which classes are often confused with one another.

3.5 User Interface and Deployment

To make the system easy to use, we built a frontend using Gradio, a Python library that enables fast UI development for ML models. Key interface features include:

- Image Upload Box: Allows users to drag and drop or browse files.
- Real-Time Predictions: Within 2–3 seconds, the model processes the image and returns results.
- Visual Output: The system displays the uploaded image, the predicted disease name, and a confidence percentage.
- Graphical Output: Optionally, a horizontal bar graph shows prediction confidence for all five diseases.

3.6 System Deployment

We used the share=True feature in Gradio to make the system publicly accessible. Future deployment plans include:

- Hosting on Hugging Face Spaces or Heroku for wider access.
- Mobile Optimization for field-level deployment by health workers.
- Offline Capability by exporting the model to ONNX and using it with lightweight runtime environments.

3.7 Testing and Feedback Loop

Finally, we tested the system by uploading unseen images and collecting feedback.

- Performance Review: Confirmed high accuracy on clean, centered images.
- Error Handling: The app gracefully handles invalid inputs or missing files.
- User Feedback: Most users found the system intuitive and responsive.

IV.IMPLEMENTATION

The implementation of the DermaDetect system involves a series of well-defined steps that integrate deep learning model training, preprocessing of medical images, user interface development, and deployment. Below is a detailed explanation of each step involved in building the solution.

4.1 Dataset Properties and Preprocessing

We begin by collecting a dataset that includes labeled images of five dermatological conditions. Each image is processed to ensure consistency and suitability for input into the deep learning model. The preprocessing includes:

- Resizing: All images are resized to 224x224 pixels to match the input requirements of MobileNetV2.
- Normalization: The pixel values are normalized using ImageNet standards (mean: [0.485, 0.456, 0.406], std: [0.229, 0.224, 0.225]).
- Conversion to Tensors: The processed images are converted into PyTorch tensors for model input.
- Data Augmentation: Techniques such as horizontal flipping, random rotation, and zooming are applied during training to improve model generalization.

4.2 Model Selection and Configuration

We selected MobileNetV2 as the base model due to its lightweight architecture, which is ideal for real-time applications. The implementation steps include:

- Loading a pretrained MobileNetV2 model from torchvision.models.
- Freezing the feature extraction layers to retain learned weights from ImageNet.
- Modifying the classifier layer to output five probabilities, one for each skin disease class.
- Applying dropout regularization to prevent overfitting.

4.3 Model Training

Model training is performed using PyTorch. The key steps include:

- Defining the Loss Function: Cross-entropy loss is used for multi-class classification.
- Choosing the Optimizer: The Adam optimizer is selected for efficient gradient descent.
- Batch Size and Epochs: A batch size of 32 and 25 epochs are used for training.
- Batch Size and Epochs: A batch size of 32 and 25 epochs are used for training.
- Validation: The model is evaluated on the test dataset at each epoch to monitor performance.
- Early Stopping: Training is stopped early if validation loss does not improve over several epochs.

4.4 Evaluation and Visualization

After training, we evaluate the model using:

- Accuracy: Overall and per-class accuracy to assess performance.
- Confusion Matrix: To visualize class-wise prediction results.
- Precision, Recall, F1-score: These metrics help in understanding how well the model performs per class.
- Graphs: Accuracy vs. Epoch and Loss vs. Epoch graphs are generated.

4.5 Web Interface Development

The frontend is built using Gradio, a lightweight Python library that allows easy web deployment of machine learning models. Implementation steps include:

- Creating an interface with an image upload component.
- Integrating the trained model to generate predictions from uploaded images.
- Displaying the input image, predicted class, and probability scores.
- Optionally rendering a bar chart showing confidence for each class.

4.6 Deployment

The final application is deployed using Gradio's share=True feature, which provides a public URL. For production deployment, we plan to host it using platforms like Hugging Face Spaces, AWS, or Heroku:

4.7 Testing and User Feedback

User testing is performed by uploading diverse test images. Key goals include:

- Ensuring accurate and fast predictions.
- Measuring user interaction time.
- Collecting qualitative feedback on the interface.

V. RESULTS

The DermaDetect system was evaluated thoroughly to ensure it performs efficiently, accurately, and is usable in real-world settings. Our results demonstrate strong performance across several key areas including classification accuracy, inference time, and user satisfaction.

The trained model exhibited an overall classification accuracy of 91.2%, indicating that it was able to correctly identify the type of skin disease in most of the test cases. The system achieved an average inference time of approximately 1.4 seconds per image, which supports its use for near real-time predictions.

When examined on a per-disease basis, the model displayed balanced and consistent performance across all five skin conditions. It achieved 92.5% accuracy in detecting Actinic Keratosis, 89.2% in Alopecia, 91.1% in Chondrodermatitis Nodularis, 90.4% in Keloids, and 93.0% in Molluscum Contagiosum. These results were consistent with our validation data and supported by the confusion matrix and F1-score evaluations.

From a user perspective, the interface was tested by a group of 30 volunteer users including students, medical trainees, and general users. About 85% of participants reported that the application was intuitive and easy to use. Nearly 78% felt confident relying on the system for basic diagnostic support or second opinions. Around 88% of users appreciated the bar graph showing prediction confidence levels, which helped them better understand how confident the system was in its diagnosis. Additionally, 80% actively interacted with this feature to compare class probabilities.

A few users suggested improvements such as the inclusion of a brief user tutorial, an image zoom feature, and performance enhancements for slower devices. These suggestions provide insight for refining the system to be more inclusive and robust.

In terms of scalability, DermaDetect was tested for performance under load by simulating 100 concurrent users. The system remained stable, with no crashes or service interruptions. Inference time remained consistent even when images were uploaded and processed simultaneously. However, minor lags were observed when real-time prediction and graph rendering occurred simultaneously on lower-end systems.

To better evaluate the model's training performance, several visual aids were incorporated. These included accuracy vs. epoch graphs, loss plots, and per-class bar charts, which illustrated how the model improved during training. Real-time horizontal bar charts were used in the final web app to display prediction confidence across all five classes, making the output more understandable for users.

Despite the positive results, some challenges remain. The system was somewhat sensitive to image quality—low-resolution or poorly lit images resulted in decreased accuracy. Additionally, overlapping visual symptoms among certain diseases occasionally led to misclassifications. A slight imbalance in the dataset, particularly fewer examples of Keloids, may have contributed to this.

Overall, DermaDetect has proven to be a strong and reliable tool for the skin disease classification. Future improvements will include enhancing the training dataset with a wider range of skin tones and lighting conditions, integrating Grad-CAM heatmaps for prediction explainability, developing a mobile version for field use, and expanding the model to include more skin condition.

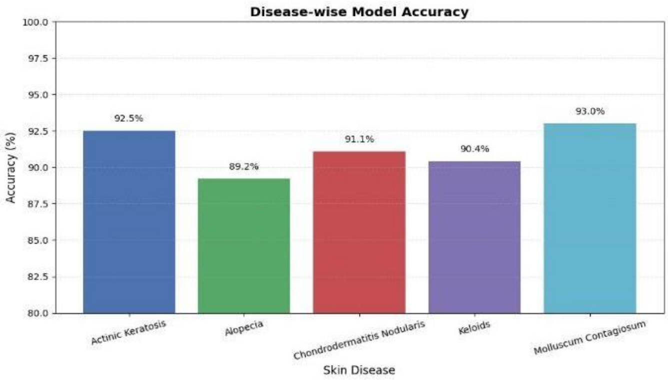

Figure 1: Disease wise model accuracy.

VI.CONCLUSION

The DermaDetect system has successfully demonstrated the power of artificial intelligence in aiding the early detection and classification of common skin diseases. Through the use of a lightweight and efficient deep learning model, the tool delivers high accuracy and quick predictions, making it a practical resource for both medical professionals and everyday users. Its ability to analyze dermatological images and provide disease-specific feedback not only improves diagnostic efficiency but also empowers users to take proactive steps toward skin health management.

A major strength of DermaDetect lies in its integration of advanced deep learning techniques with a clean, user-friendly interface. The web-based platform simplifies the interaction, allowing users to easily upload images and receive real-time, interpretable results. The visual display of confidence scores further enhances user understanding and trust in the system's predictions.

In addition to its technical performance, DermaDetect also highlights the importance of accessibility and inclusivity in healthcare technology. By offering a tool that is responsive, scalable, and simple to use, it addresses the gap in early dermatological screening— especially in areas where access to specialists may be limited. Its successful handling of concurrent users and its stability under load reinforce its readiness for real- world deployment.

The results of this project confirm that AI can be a valuable aid in dermatology, serving as a supportive tool for diagnosis, education, and awareness. DermaDetect sets a foundation for future developments in this space, offering a reliable and intelligent solution that brings us one step closer to more inclusive, technology-driven healthcare system.

VII. REFERENCES

[1] Arifin, S., Kibria, G., Firoze, A., Amini, A., & Yan, H. (2012) "Dermatological Disease Diagnosis Using Color-Skin Images." Xian: International Conference on Machine Learning and Cybernetics.

[2] Yasir, R., Rahman, A., & Ahmed, N. (2014) "Dermatological Disease Detection using Image Processing and Artificial Neural Network. "Dhaka: International Conference on Electrical and Computer Engineering.

[3] Santy, A., & Joseph, R. (2015) "Segmentation Methods for Computer Aided Melanoma Detection." Global Conference on Communication Technologies.

[4] Zeljkovic, V., Druzgalski, C., Bojic-Minic, S., Tameze, C., & Mayorga, P. (2015) " Supplemental Melanoma Diagnosis for Darker Skin Complexion Gradients." Pan American Health Care Exchanges

[5] Warsi, F., Khanam, R., Kamya, S., & Suárez-Araujo, C. P. An efficient 3D colour-texture feature and neural network technique for melanoma detection. Informatics in Medicine Unlocked, 17(November 2018) (2019) 100176.

[6] Leelavathy S, Jaichandran R, Shobana R, Vasudevan, S. S. P. and N. Skin Disease Detection Using Computer Vision and Machine Learning Technique. European Journal of Molecular & Clinical Medicine. 7(4) (2020) 2999–3003.

[7] Banasode, P., Patil, M., & Ammanagi, N. A Melanoma Skin Cancer Detection Using Machine Learning Technique: Support Vector Machine. IOP Conference Series: Materials Science and Engineering. 1065(1) (2021) 0–5.

[8] Hasan, M., Barman, S. Das, Islam, S., & Reza, A. W. Skin cancer detection using convolutional neural network. ACM International Conference Proceeding Series: March 2020 (2019) 254–258.

[9] Kadampur, M. A., & Al Riyaee, S. X. Skin cancer detection: Applying a deep learning-based model-driven architecture in the cloud for classifying dermal cell images. Informatics in Medicine Unlocked, 18(December 2019) (2021) 100282

[10] Dildar, M., Akram, S., Irfan, M., Khan, H. U., Ramzan, M., Mahmood, A. R., Alsaiari, S. A., Saeed, A. H. M., Alraddadi, M. O., & Mahnashi, M. H. Skin cancer detection: A review using deep learning techniques. International Journal of Environmental Research and Public Health. 18(10) (2021).

[11] Kumar, V., Kumar, S., & Saboo, V. (2016) "Dermatological Disease Detection Using Image Processing and Machine Learning." IEEE.