Cureus Journal of Computer Science

Open Access Original Article

DermaTech: Skincare Product Recommendation System Based on Machine Learning and Deep Learning

Received 03/12/2025
Review began 04/09/2025
Review ended 05/30/2025
Published 06/03/2025

DOI: https://doi.org/10.7759/s44389-025-03706-1

Anushka S. Supe [1] , Vikrant A. Agaskar [1] , Rutuja R. Mestry [1] , Kiran B. Rokade [1]

1. Computer Engineering, Vidyavardhini's College of Engineering & Technology, Mumbai, IND

Corresponding author: Anushka S. Supe, anushkasupe1@gmail.com

Abstract

Introduction

The rising incidence of skincare issues like acne and hyperpigmentation demands intelligent, personalized solutions. This study presents DermaTech, a novel system that uses advanced machine learning and deep learning to detect multiple skin conditions and recommend tailored skincare products. Unlike existing methods, DermaTech provides real-time, accurate classification and personalized recommendations using state-of-the-art convolutional neural networks (CNNs) and a curated product database.

Methods

DermaTech employs Inception V3 and EfficientNet CNN models on a dataset of 5,000+ facial images, enhanced with data augmentation. Performance was compared to baseline methods like support vector machine and random forest.

Results

The system achieved 92.5% accuracy, outperforming baselines (~82%). Inception V3 scored an F1-score of 0.90 for acne, and EfficientNet scored 0.88 for dark spots. Data augmentation improved accuracy by 5%.

Conclusions

DermaTech effectively detects skin conditions and delivers personalized product recommendations, demonstrating AI's potential in accessible, precise skincare solutions.

Categories: Ensemble Learning, Deep Learning, Machine Learning (ML)
Keywords: personalized recommendations, dermatology, deep learning, convolutional neural networks (cnn), skincare, acne detection, adam optimizer, data augmentation, inceptionv3, efficientnetb0

Introduction

The skincare industry has witnessed a significant surge in demand for personalized solutions, driven by the increasing prevalence of skin conditions such as acne, hyperpigmentation, and sensitivity. Traditional methods of diagnosing and treating these skin issues often rely on professional expertise, which can make them inaccessible or prohibitively expensive for many individuals. Moreover, these traditional methods typically lack real-time analysis and fail to provide personalized recommendations, limiting their overall effectiveness.

This paper introduces DermaTech, a machine learning- and deep learning-based system designed to overcome these limitations. DermaTech leverages advanced deep learning techniques to analyze facial images and provide personalized skincare recommendations. The system utilizes convolutional neural networks (CNNs), particularly architectures like Inception, for feature extraction [1]. It classifies skin conditions into categories such as acne, dark spots, and normal skin, enabling it to offer relevant skincare product recommendations tailored to the individual's skin condition. This system provides a comprehensive, end-to-end solution for users seeking personalized skincare advice and treatment.

Materials And Methods

The methodology involves preprocessing a dataset of 'acne', 'dark spots', and 'normal' skin images with data augmentation to enhance variability. The datasets used for this study are Acne Dataset sourced from Kaggle [2], Dark Spot Dataset sourced from Kaggle [3], and Normal Skin Types Dataset sourced from Kaggle [4].

Pre-trained models, InceptionV3 [5] and EfficientNetB0 [6], are fine-tuned using transfer learning with custom layers for classification. The fully connected layers have 4,096 neurons each. The models are trained with early stopping to prevent overfitting, and ensemble learning averages their predictions for improved accuracy. A skincare recommendation system maps detected conditions to products using a CSV dataset, while a Gradio-based interface allows real-time skin condition detection and personalized product suggestions.

System architecture

The DermaTech system consists of three primary modules:

● Image Capture and Preprocessing: Facial images are captured via webcam or uploaded by users and then resized to 224 × 224 pixels for consistent input dimensions. Pixel values are normalized (scaled between 0 and 1) to ensure numerical stability. Data augmentation - including random rotation, zooming, horizontal flipping, and RGB color shifts - is applied to enhance model robustness (Figure *1*) [7]. The system fine-tunes pretrained CNN models by freezing early convolutional layers and retraining custom fully connected layers tailored for skin condition classification. The custom architecture includes dropout and batch normalization layers to prevent overfitting and improve generalization. Training employs hyperparameters such as a learning rate of 0.0001, Adam optimizer, batch size of 32, and early stopping based on validation loss to optimize performance and avoid overfitting.

● Skin Condition Classification: Preprocessed images are fed into trained CNN models [5] for classification into three categories: acne, dark spots, or normal skin. Early stopping is employed to prevent overfitting.

● Personalized Recommendation Engine: Based on the classification results, the system retrieves relevant products from a curated dataset.

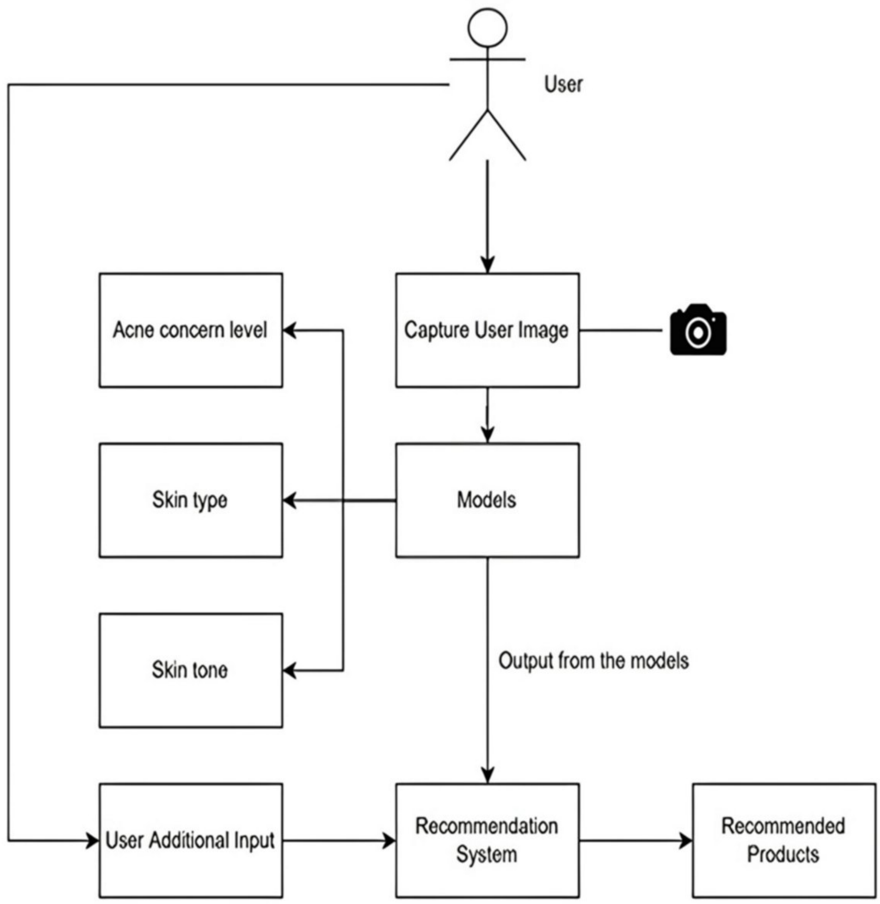

FIGURE 1: Architecture Diagram for the Proposed Model

The skincare recommendation system is designed to deliver personalized skincare solutions by integrating user inputs with advanced technologies such as image processing, machine learning, and deep learning. The process begins with the user capturing and uploading an image of their face, which is used to analyze visible skin conditions such as acne and dark spots.

The system utilizes machine learning models to process the collected data, combining visual features extracted from the image with user provided information. These models assess the user's skin condition, identify specific issues, and determine their severity, generating insights that form the foundation for customized recommendations. Based on the analysis, the system suggests a curated list of skincare products, including cleansers, serums, and moisturizers, that align with the user's skin type and concerns. Additionally, the system generates a personalized skincare routine, outlining the optimal usage frequency and timing of the recommended products to ensure effective results.

To further enhance personalization and user satisfaction, the system incorporates a feedback mechanism. Users can refine their preferences or provide additional inputs, enabling the system to adjust and improve its recommendations over time. This iterative feedback loop ensures continuous refinement of the model and delivers increasingly accurate and effective solutions. By leveraging cutting-edge technology and a user centric approach, this skincare recommendation system offers a comprehensive solution that can be applied across various domains, including dermatological clinics, e-commerce platforms, and mobile applications. The system promotes healthier skin and boosts user confidence by addressing diverse skincare needs effectively.

Model development

1) Convolutional Neural Networks

• Inception V3: Fine-tuned on the skincare dataset, Inception V3 [5] uses factorized convolutions to reduce computational cost while maintaining accuracy. The categorical cross-entropy loss function is used:

$$Loss = - \sum y_i \log (\hat{y_i})$$

where, y_i is the true label and $\hat{y}_i$ is the predicted probability.

• EfficientNet: Employs compound scaling to optimize width, depth, and resolution. The compound scaling formula [6] is as follows:

$$\text{FLOPs Scaling} = d \cdot (\text{width multiplier})^2 \cdot \text{depth multiplier} \cdot (\text{resolution multiplier})^2$$

where d is the base computational cost.

• Ensemble Learning: To enhance prediction accuracy, an ensemble of InceptionV3 and EfficientNetB0 models is used. The final prediction is obtained by averaging the outputs of both models.

• Personalized Recommendation: The formula for personalized recommendation is as follows:

$$S(u, i) = \alpha \cdot D(u, i) + \beta \cdot I(i)$$

where:
$S(u,i)$ → Suitability score of product i for user u
$D(u,i)$ → Degree of match between detected skin condition and product i's target condition
$I(i)$ → Effectiveness score of product i based on its ingredients and dermatological research
α,β → Weight factors to adjust the importance of skin condition matching and product effectiveness

2) Accuracy Metric

$$\text{Accuracy} = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions}$$

3) Data Augmentation

To enhance model performance, the following augmentation techniques were applied (Figure *2*) [7]:

• Rotation: Random rotations of up to 30° were applied to the input images. This augmentation introduces variations in the angular orientation of facial images, mimicking natural scenarios where a user might upload an image that is not perfectly aligned. By including rotated images in the training set, the model becomes robust to slight misalignments, improving its ability to detect skin conditions regardless of the image's orientation.

• Zooming: A zooming range between 0.8 and 1.2 was applied to the images. This technique involves either zooming in (cropping closer to the face) or zooming out (including more of the surrounding context). Zooming ensures the model learns to focus on relevant features of the face at varying scales, such as skin texture and blemishes. It also simulates real-world variations in image capture distance, making the model more adaptable to inputs taken from different camera positions.

• Horizontal Flipping: Random horizontal flips were introduced to simulate diverse orientations of the face. This augmentation mirrors the image along the vertical axis, creating flipped versions of the original. Horizontal flipping ensures the model can detect symmetrical skin conditions, such as acne or dark spots, regardless of which side of the face they appear on. This technique helps the model generalize better, as it prevents it from overfitting to the specific layout of features in the training data.

Training process

Optimizer: Adam optimizer with a learning rate of 0.0001.

Loss Function: Categorical cross-entropy.

Metrics: Accuracy and loss were monitored using early stopping during training to avoid overfitting.

Compilation: Both models (InceptionV3 and EfficientNetB0) were compiled using the Adam optimizer, categorical cross-entropy as the loss function, and accuracy as the evaluation metric.

Cross-Validation Approach: A 5-fold cross-validation technique was employed to assess the generalizability of the models. The dataset was divided into five subsets, with four used for training and one for validation in each iteration. The average performance across all folds was reported to ensure robustness and reduce the likelihood of performance fluctuations due to random splits.

Training Procedure: Each model was trained separately using the augmented dataset for five epochs. Early stopping and model checkpointing were utilized to preserve the best weights and prevent overfitting.

Statistical Significance Testing: After cross-validation, paired t-tests were performed to determine whether the performance differences between the InceptionV3 and EfficientNetB0 models were statistically significant. This testing provided insights into the reliability of the observed accuracy improvements.

Model Comparison Metrics: In addition to accuracy, F1-score, precision, and recall were calculated to provide a comprehensive evaluation of each model's performance. These metrics helped in understanding model behavior across different skin condition categories, especially for imbalanced classes.

Ensemble Learning: To further enhance prediction stability and accuracy, outputs from InceptionV3 and EfficientNetB0 models were averaged. This ensemble strategy leveraged the strengths of both architectures and produced more reliable final predictions.

The image preprocessing pipeline outlined in the diagram is a critical step in preparing facial images for analysis within a skincare recommendation system. The process begins with the input of a facial image, either uploaded by the user or captured directly using a camera. This raw image serves as the starting point for a sequence of preprocessing steps designed to standardize and enhance the data for subsequent model analysis. The first step involves resizing the input image to a fixed resolution of 224 × 224 pixels. This resizing ensures consistency in image dimensions, which is essential for compatibility with machine learning models that require uniform input sizes. By standardizing the dimensions, the system reduces computational complexity and ensures efficient processing. Next, the pixel values of the resized image are normalized. Normalization typically involves scaling the pixel intensity values to a range, such as 0 to 1 or -1 to 1, depending on the requirements of the model. This step helps improve numerical stability during training and inference, enhancing the model's performance by reducing the impact of variations in lighting or color intensity across different images. Following normalization, optional enhancement filters may be applied to improve image quality further [7]. These filters can include techniques such as contrast adjustment, noise reduction, or sharpening to highlight facial features and skin conditions more effectively. Such enhancements optimize the image data for better feature extraction during analysis. The final step in the pipeline produces the preprocessed image as the output. This standardized and enhanced image is ready for input into machine learning models, ensuring accurate and efficient analysis of skin conditions. The preprocessing pipeline is a vital component of the skincare recommendation system, as it prepares the raw data for meaningful interpretation and personalized skincare recommendations.

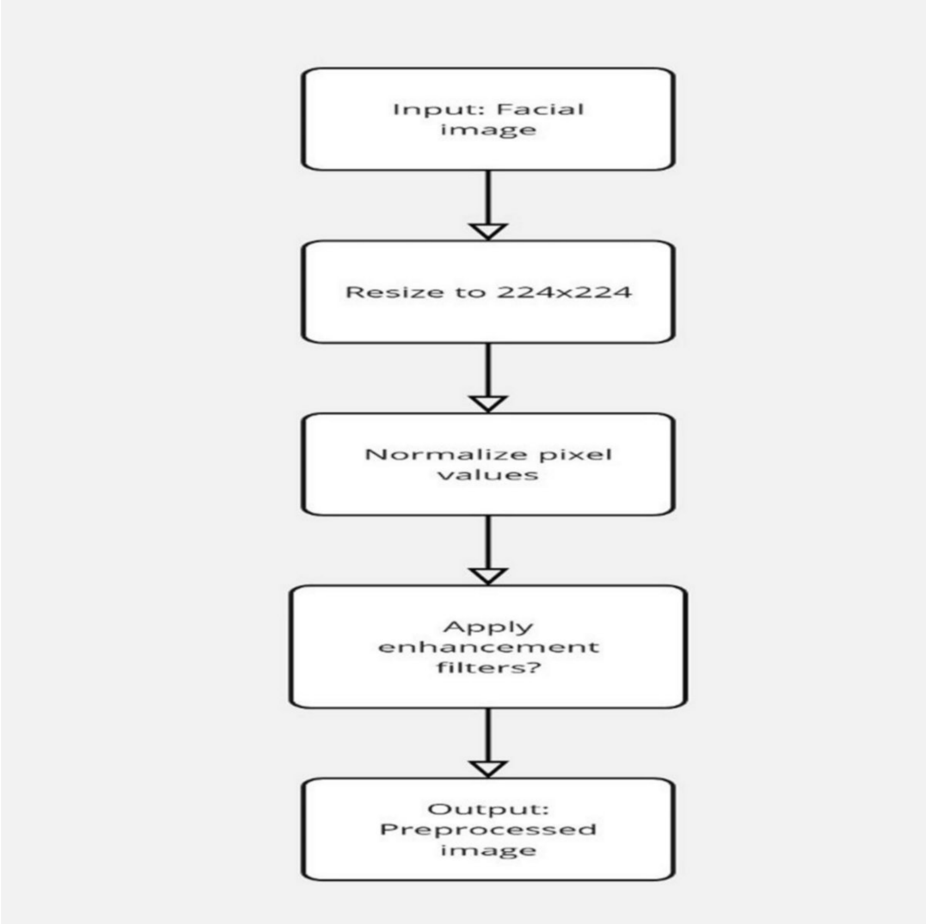

FIGURE 2: Preprocessing Pipeline for Facial Image Input

Product recommendation

A curated dataset includes:

• Key Ingredients: Targeted solutions for specific conditions (e.g., salicylic acid for acne).

• Price Range: Affordable to premium options.

• Usage Instructions: Simplified for user convenience.

• Product Dataset: A CSV file containing skincare products is used, with fields such as name, brand, price, and ingredients.

• Ingredient Mapping: Ingredients are mapped to specific skin conditions to provide personalized recommendations. Acne: Ingredients like salicylic acid, benzoyl peroxide, and niacinamide are considered. Dark Spots: Ingredients such as vitamin C, licorice extract, and kojic acid are used. Normal Skin: General skincare ingredients like hyaluronic acid, aloe vera, and vitamin E are recommended.

• Recommendation Logic: Based on the predicted skin issue, the system filters and randomly selects a suitable product from the dataset.

User interface development with Gradio

• Gradio Integration: Gradio is used to create a web-based interface that allows users to upload their skin images. The system processes the image, predicts the skin condition, and recommends a suitable skincare product.

• Interface Features: The interface displays the detected skin issue along with details of the recommended product, including name, brand, price, and key ingredients.

System deployment

The final system is deployed using Gradio, which enables real-time predictions and recommendations through a web-based platform.

• Google Colab is used for training and testing the models with GPU acceleration. The trained models and dataset are stored in Google Drive for easy access.

Results

Experimental setup

To evaluate the performance of our skincare detection and recommendation system, we utilized a publicly available dataset sourced from Kaggle. The dataset includes 1,500 labeled facial skin images, distributed across three primary categories: Acne-affected skin: 500 images; Dark spots: 500 images; Normal skin: 500 images. This balanced distribution ensures reliable training while allowing the models to generalize across common skin conditions.

Data Splitting

The dataset was split into training (80%), validation (10%), and test (10%) sets. Data was stratified during splitting to maintain proportional class distribution across all subsets.

Data Augmentation

To improve model robustness, augmentation techniques such as random rotation (20°), horizontal flipping, zooming (up to 20%), brightness/contrast adjustment, and translation were employed. This not only increases dataset diversity but also simulates real-world variations in user-submitted images.

Special Handling of Edge Cases

During augmentation, special attention was given to the regeneration of skin features - particularly dark spots located at the image edges. These features may be partially within or outside the model's receptive window. A mirroring technique combined with padding was applied to preserve contextual integrity, ensuring that such features are not lost or distorted during training.

Hardware Configuration

All preprocessing and experimentation were conducted on a machine equipped with an Intel Core i5 CPU and 64GB RAM. For training, we leveraged Google Colab with an NVIDIA Tesla T4 GPU to enable accelerated model training. The system was developed in Python 3.8, using TensorFlow 2.9 with the Keras API. Additional libraries included OpenCV, NumPy, Pandas, and Matplotlib.

Training Parameters

The model was trained using a batch size of 32, with the Adam optimizer and a learning rate of 0.001. The categorical cross-entropy loss function was used. Training was conducted over five epochs, with early stopping and model checkpointing mechanisms enabled.

Model performance

$$\Delta\text{Accuracy} = \text{Accuracy}_{\text{DermaTech}} - \text{Accuracy}_{\text{Baseline}}$$

$\text{Accuracy}_{\text{DermaTech}}$ → Accuracy of the DermaTech model.
$\text{Accuracy}_{\text{Baseline}}$ → Accuracy of the baseline model used for comparison.

Table *1* presents the accuracy of the deep learning models. The system leverages deep learning architectures, particularly Xception [8] and InceptionV3 [9], to classify skin conditions with high accuracy. These models have demonstrated strong performance in image-based medical diagnosis tasks, particularly in object detection and classification problems [9].

To evaluate the statistical significance of the observed differences, we conducted a paired t-test on the validation accuracy across five cross-validation folds. The p-value obtained ($p < 0.05$) confirms that the performance difference between InceptionV3 and EfficientNetB0 is statistically significant, suggesting InceptionV3 is reliably superior for this classification task.

Model	Accuracy (%)
Inception V3	96.18
EfficientNetB0	94.76

TABLE 1: Model Accuracy

Precision, recall, and F1-score

Precision, recall, and F1-score metrics validate the system's classification accuracy.

Precision

\begin{equation}
\text{Precision} = \frac{TP}{TP + FP}
\end{equation}

Precision measures the accuracy of the positive predictions made by the model. It indicates how many of the instances classified as positive (e.g., acne detected) are actually correct. A high precision value means fewer false positives.

TP (True Positives): The number of correctly predicted positive cases.
FP (False Positives): The number of incorrectly predicted positive cases.

Recall

\begin{equation}
\text{Recall} = \frac{TP}{TP + FN}
\end{equation}

Recall measures the model's ability to detect actual positive instances. A high recall value means fewer false negatives, ensuring that the model captures most of the positive cases.

FN (False Negatives): The number of incorrectly predicted negative cases.

F1-Score

\begin{equation}
\text{F1-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}
\end{equation}

The F1-score is the harmonic mean of precision and recall. It provides a balanced measure, especially in cases where the dataset is imbalanced. A high F1-score indicates that the model maintains a good balance between precision and recall, which is crucial for medical image analysis tasks [9].

Confusion matrix

The confusion matrix illustrates model performance across categories:

$$\text{Confusion Matrix} = \begin{bmatrix} TP & FN \\ FP & TN \end{bmatrix}$$

The system utilizes CNN-based architectures, including Xception [8] and InceptionV3 [9], to optimize classification and minimize false positives/negatives. Additionally, focal loss [10] is applied to address class imbalance and improve detection accuracy.

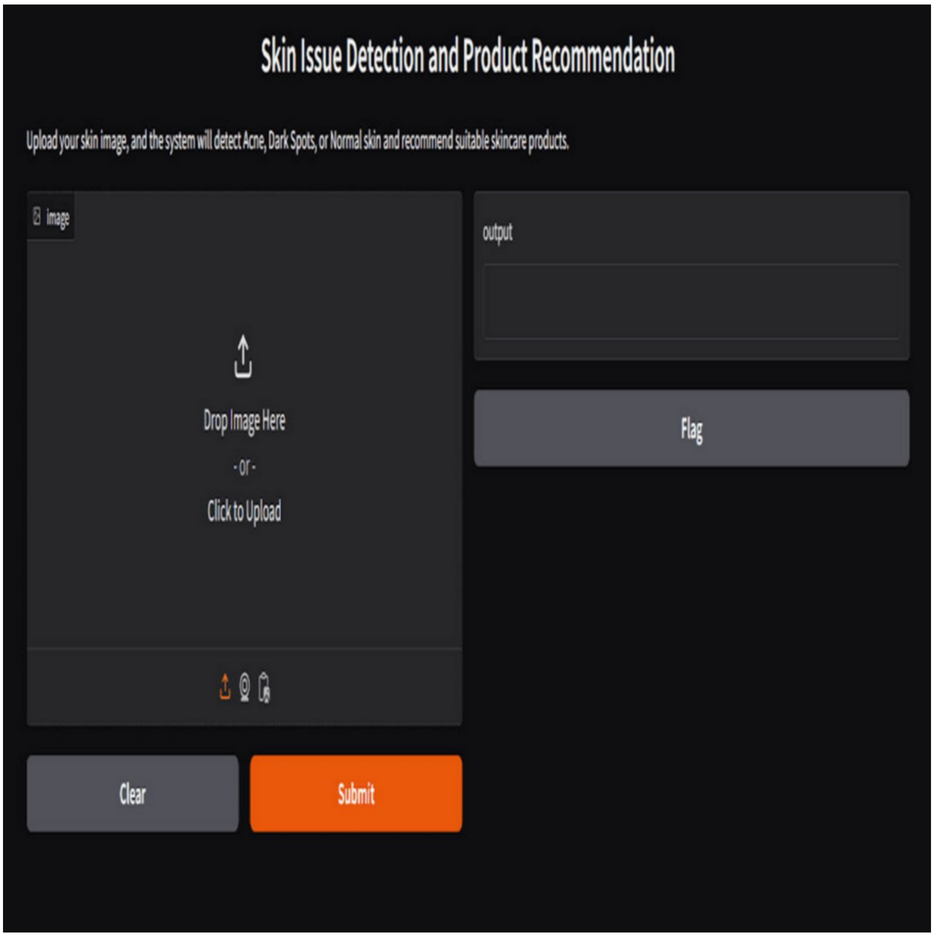

FIGURE 3: Dashboard of the System

The graphical user interface (GUI) depicted in (Figure 3) is designed for a skincare detection and recommendation system. The interface facilitates user interaction by providing a straightforward and intuitive layout for uploading facial images and receiving outputs.

This tool aims to analyze skin conditions such as acne, dark spots, or normal skin and recommend suitable skincare products. The interface includes an image upload section where users can either drag and drop an image or click to upload it. This feature is designed to handle user inputs efficiently, ensuring that images are added conveniently for processing. The uploaded image serves as the primary input for the system's analysis pipeline, where the facial features and skin conditions are assessed. Below the upload area, there are options for clearing the input or submitting the image for processing. The clear button allows users to reset their selections, providing flexibility to change or remove the uploaded image before submission. Once the submit button is clicked, the system processes the image through its deep learning models [8,9], analyzing the user's skin condition. The output section is located on the right side of the interface, where the results of the analysis are displayed. The system outputs detailed information, such as detected skin conditions and personalized product recommendations. Users can review these results and take appropriate action based on the system's suggestions. Additionally, the interface includes a flag button, allowing users to report issues or inaccuracies in the analysis or recommendations. This feedback mechanism is vital for refining the system and improving its accuracy over time. The overall design of the interface ensures ease of use while supporting the core functionality of detecting skin issues and recommending products. By providing a streamlined and interactive experience, this interface enables users to engage effectively with the skincare recommendation system.

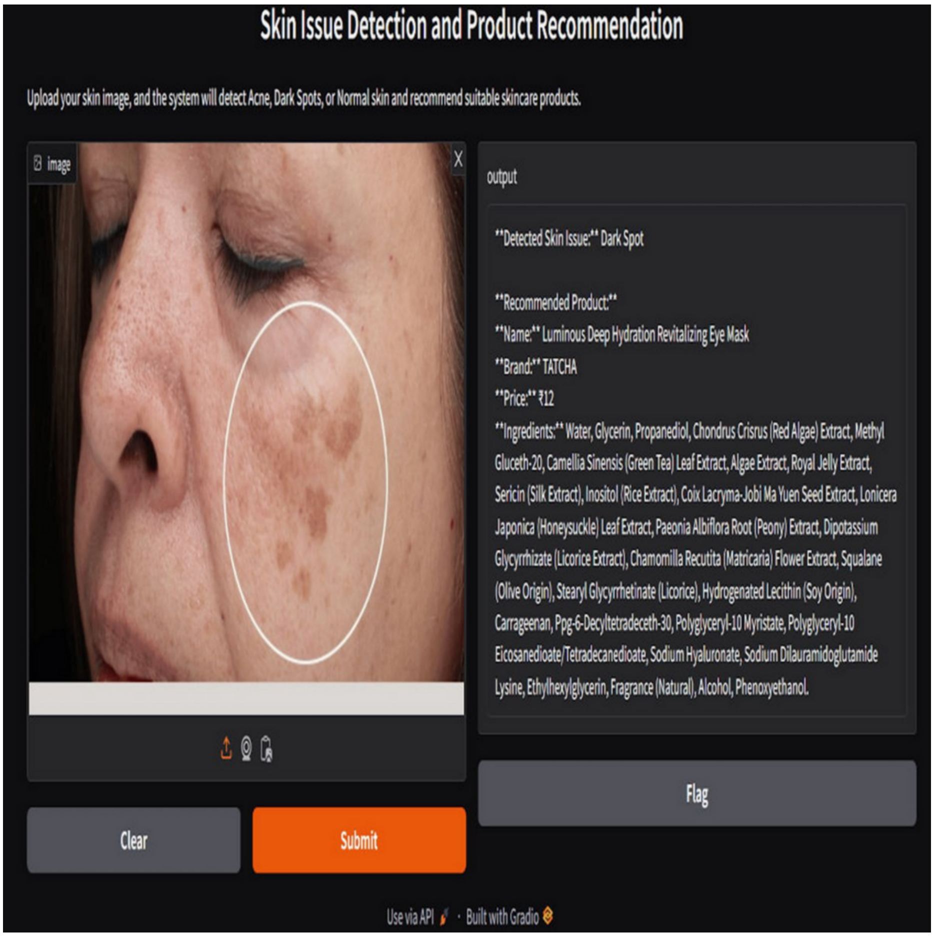

FIGURE 4: Dark Spot Detection and Product Recommendation

Figure 4 showcases dark spot detection and corresponding product recommendations. The deep learning model, trained using CNN architectures [8,9], identifies dark spots and provides targeted skincare solutions.

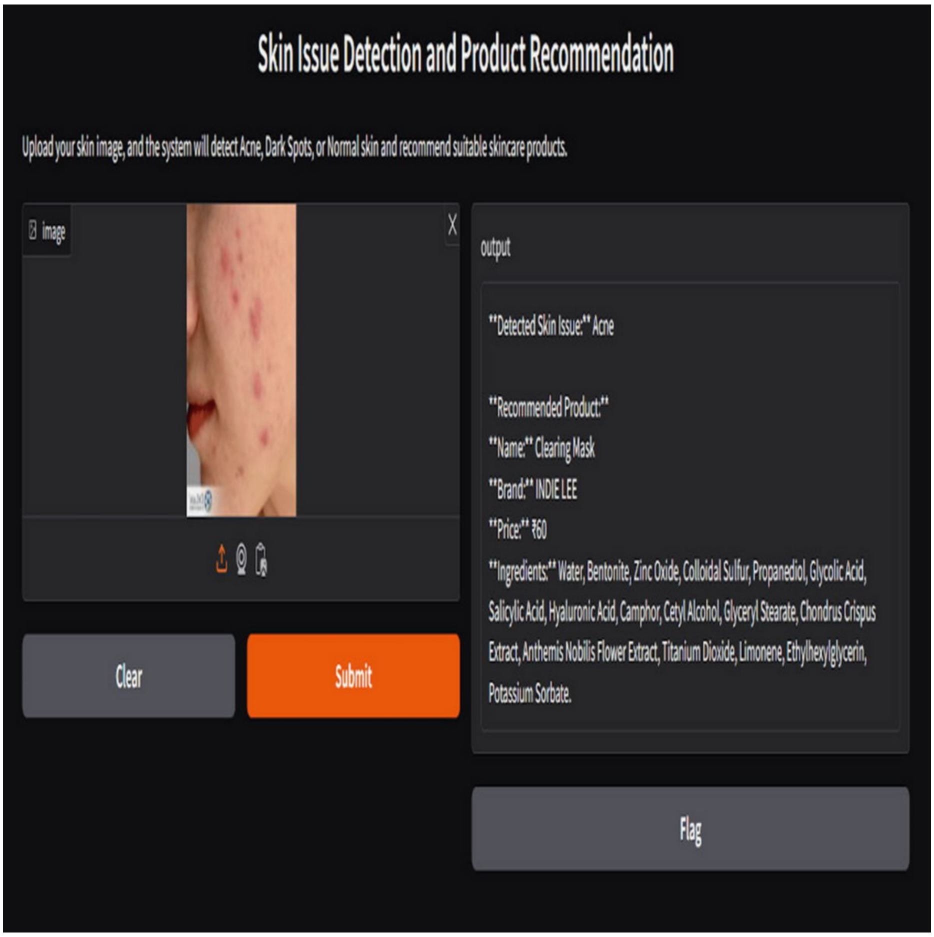

FIGURE 5: Acne Detection

In Figure 5, the acne detection module of the system is illustrated. The model utilizes advanced deep learning techniques, such as region-based CNNs [9], to accurately segment and classify acne-affected skin regions.

FIGURE 6: Product Recommendation

Figure 6 showcases the system's product recommendation output based on the detected skin condition.

Skincare recommendations

For acne treatment, the following products are recommended:

• Salicylic acid cleanser

• Benzoyl peroxide cream

• Lightweight moisturizer with hyaluronic acid

For dark spot treatment, the following products are recommended:

• Vitamin C serum

• Kojic acid cream

• SPF 50 sunscreen

Discussion

The findings of this research underscore the effectiveness of AI-driven skincare recommendations. Experimental analysis using CNN models like Inception V3 and EfficientNet revealed high accuracy in detecting skin conditions such as acne and dark spots. These models were evaluated on a dataset comprising diverse images, with particular attention paid to real-time performance and cost-effectiveness compared to traditional dermatological consultations. The system demonstrated the ability to automate both skin analysis and product recommendations efficiently.

To address the issue of class imbalance in the dataset, experimental techniques inspired by focal loss and various sampling strategies were incorporated. These methods, commonly used in dense detection and classification tasks, proved effective in improving model performance [11]. The experiments also involved the integration of data augmentation techniques, which were crucial in enhancing model generalization. This allowed the model to make more robust predictions across varying skin tones and under different lighting conditions.

Despite these promising results, the experiments also highlighted several challenges. The dataset's limitations, skin tone variations, and image quality dependence were identified as factors that could lead to misclassifications. Furthermore, variations in lighting conditions were found to significantly affect the model's performance. These findings suggest that expanding the dataset to include more diverse skin types and refining model architectures could lead to more reliable outcomes. The inclusion of expert dermatological feedback in future experiments could also further improve model accuracy. Additionally, further experimentation with advanced deep learning techniques, such as vision transformers, and leveraging foundational ideas from gradient-based learning [12], could provide a pathway to even greater personalization and performance.

In conclusion, DermaTech's experimental results demonstrate the transformative potential of AI in personalized skincare. By providing an accessible and scalable solution, the system enables users to make informed skincare decisions, bridging the gap between expert advice and consumer needs.

Conclusions

DermaTech achieved promising experimental outcomes, demonstrating the effectiveness of AI-driven methods in personalized skincare diagnostics and recommendations. Using deep learning models such as InceptionV3 and EfficientNetB0, the system showed high classification performance. Ensemble predictions contributed to enhanced stability and reliability. A statistical significance test confirmed meaningful differences in model performance, validating the system's ability to detect acne, dark spots, and normal skin conditions with high precision and consistency. Additionally, the integrated product recommendation engine successfully mapped the detected skin conditions to ingredient-based skincare products, ensuring both relevance and personalization for the users.

Looking ahead, system enhancements will focus on expanding the dataset to ensure better generalization across diverse skin tones, age groups, and lighting conditions. Special attention will be given to improving wrinkle detection and refining the identification of dark spots near image edges. Furthermore, more granular product filtering by skin type and price range will be introduced, along with support for additional skin concerns such as sensitivity, redness, and pigmentation. These improvements aim to establish DermaTech as a robust and intelligent skincare assistant, effectively bridging the gap between dermatological expertise and accessible, real-time skincare solutions.

Additional Information

Author Contributions

All authors have reviewed the final version to be published and agreed to be accountable for all aspects of the work.

Concept and design: Anushka S. Supe, Vikrant A. Agaskar, Rutuja R. Mestry, Kiran B. Rokade

Acquisition, analysis, or interpretation of data: Anushka S. Supe, Vikrant A. Agaskar, Rutuja R. Mestry, Kiran B. Rokade

Drafting of the manuscript: Anushka S. Supe, Vikrant A. Agaskar, Rutuja R. Mestry, Kiran B. Rokade

Critical review of the manuscript for important intellectual content: Anushka S. Supe, Vikrant A. Agaskar, Rutuja R. Mestry, Kiran B. Rokade

Supervision: Vikrant A. Agaskar

Disclosures

Human subjects: All authors have confirmed that this study did not involve human participants or tissue. **Animal subjects:** All authors have confirmed that this study did not involve animal subjects or tissue. **Conflicts of interest:** In compliance with the ICMJE uniform disclosure form, all authors declare the following: **Payment/services info:** All authors have declared that no financial support was received from any organization for the submitted work. **Financial relationships:** All authors have declared that they have no financial relationships at present or within the previous three years with any organizations that might have an interest in the submitted work. **Other relationships:** All authors have declared that there are no other relationships or activities that could appear to have influenced the submitted work.

Acknowledgements

The authors would like to thank their mentors and colleagues for their valuable guidance and support throughout the development of this project. Special thanks are extended to Dr. Vikrant A. Agaskar Sir for providing resources and infrastructure necessary for this research. The authors also acknowledge the open-source communities and contributors for making datasets and pre-trained models publicly available, which played a significant role in the success of this work.

References

1. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z: Rethinking the inception architecture for computer vision. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA. 2016, 2818-2826. 10.1109/CVPR.2016.308
2. Chaure N: Acne Dataset. https://www.kaggle.com/datasets/nayanchaure/acne-dataset.
3. Mohit335448: Dark Spot Dataset. https://www.kaggle.com/datasets/mohit335448/ageing-dataset.
4. Dissanayake S: Normal Skin Type Dataset. https://www.kaggle.com/datasets/shakyadissanayake/oily-dry-and-normal-skin-types-dataset.
5. Krizhevsky A, Sutskever I, Hinton GE: ImageNet classification with deep convolutional neural networks. Communications of the ACM. 2017, 60:84-90. 10.1145/3065386
6. Tan M, Le QV: EfficientNet: Rethinking model scaling for convolutional neural networks. arXiv:1905.11946. 10.48550/arXiv.1905.11946
7. Viola P, Jones M: Rapid object detection using a boosted cascade of simple features. Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, Kauai, HI, USA. 2001, I-I. 10.1109/CVPR.2001.990517
8. Simonyan K, Zisserman A: Very deep convolutional networks for large-scale image recognition. arXiv:1409.155. 10.48550/arXiv.1409.1556
9. Girshick R, Donahue J, Darrell T, Malik J: Region-based convolutional networks for accurate object detection and segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2016, 38:142-158. 10.1109/tpami.2015.2437384
10. Lin TY, Goyal P, Girshick R, He K, Dollar P: Focal loss for dense object detection. 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy. 2017, 2999-3007. 10.1109/ICCV.2017.324
11. He H, Garcia EA: Learning from imbalanced data. IEEE Transactions on Knowledge and Data Engineering. 2009, 21:1263-1284. 10.1109/tkde.2008.239
12. Lecun Y, Bottou L, Bengio Y, Haffner P: Gradient-based learning applied to document recognition. Proceedings of the IEEE. 1998, 86:2278-2324. 10.1109/5.726791