www.ijarp.com ISSN 2456-9992 Page: 01-12

Research Article

International Journal Advanced Research Publications

Volume: 02
Issue: 10

DERMAVISION: AI-POWERED SKIN ANALYSIS AND PRODUCT RECOMMENDATION

V. Merlinfreeda*[1], [2]Aravindh A., [3]Priyan S., [4]Sasikumar K.

[1-3]Department of Computer Science and Engineering (Internet of Things)

[1-3]K.L.N. College of Engineering, Pottapalayam, Tamil Nadu, India

Article Received: 17 September 2026, Article Revised: 07 October 2026, Published on: 27 October 2026

***Corresponding Author: V. Merlinfreeda**

Department of Computer Science and Engineering (Internet of Things) K.L.N. College of Engineering, Pottapalayam, Tamil Nadu, India

Doi: https://doi-doi.org/101555/ijarp.3125

2. ABSTRACT

Choosing skincare and makeup products that suit an individual's skin usually relies on manual research, trial and error or generic advice, and questionnaire-based applications rarely analyse the face directly or consider several skin attributes together. This paper presents DermaVision, an AI-powered web application that estimates skin type, skin tone and acne concern level from a user's selfie and recommends skincare and makeup products matched to the resulting skin profile. The analysis pipeline uses face-api.js for browser-side face detection, OpenCV for image preprocessing, EfficientNet-B0 convolutional models for skin-type (Dry, Normal, Oily) and acne-severity classification, and a K-Nearest Neighbors (KNN) classifier on skin colour values for skin-tone classification. Products are ranked by cosine similarity between the user's skin profile and product feature vectors. The application is built with React.js, Node.js and Express, and Python AI services, with product recommendation data stored in CSV format and application data in MongoDB. On validation data, the project reports an accuracy of 80% (precision 82%, recall 80%, F1-score 80.5%) for skin type, 84% (precision 83%, recall 82%, F1-score 82.5%) for acne severity and 81% accuracy for skin tone. Dataset sizes, the validation protocol and independent test results are not fully specified, so these figures are indicative only. DermaVision supports product discovery and does not provide a medical diagnosis. Planned work covers e-commerce integration with price comparison, weekly routine tracking and skin progress monitoring.

3. KEYWORDS: DermaVision; Skin analysis; Computer vision; EfficientNet-B0; K-Nearest Neighbors; Cosine similarity; Personalized product recommendation

4. INTRODUCTION

Skincare and cosmetic products differ widely in formulation, intended skin type and target concern, which makes it hard for consumers to identify products that suit them. Selection often depends on manual research, advice from others or trial and error, and generic recommendations may not reflect an individual's skin. Deep learning applied to facial images offers a way to estimate skin attributes automatically. Earlier studies have classified skin type from facial images [4] and have combined facial skin condition analysis with cosmetic ingredient analysis for product recommendation [1].

This paper presents DermaVision, an AI-powered web application that analyses a user's selfie to estimate three attributes: skin type, skin tone and acne concern level. The three estimates are combined into a user skin profile, which the application matches against skincare and makeup product features using cosine similarity [5]. Before recommendations are generated, users can review and modify the detected attributes and add further skin concerns.

The contributions of this work are: (i) an integrated analysis pipeline that combines browser-side face detection, OpenCV preprocessing, two EfficientNet-B0 classifiers [2] and a KNN skin-tone classifier; (ii) a cosine-similarity recommendation approach that ranks products against the combined skin profile; and (iii) a transparent account of the project-reported evaluation results together with the details that the available material does not specify. DermaVision produces estimates to support product discovery. It is not a medical diagnostic tool. The remainder of the paper is organised as follows. Section 5 describes the existing system and the research motivation, Section 6 reviews related work, Section 7 states the objectives, Section 8 describes the proposed system, Section 9 explains the materials and methods, Section 10 lists the implementation technologies, Section 11 reports the model evaluation results, Section 12 discusses the findings, Section 13 covers limitations and responsible use, and Section 14 concludes and outlines future work.

5. EXISTING SYSTEM AND RESEARCH MOTIVATION

Traditional skincare product selection is mainly based on manual research, trial and error, or generic recommendations that may not suit individual skin needs. Most existing beauty applications depend on questionnaires completed by the user instead of automatically analysing

facial skin characteristics from images. Such systems also often fail to consider several skin attributes, such as skin tone, skin type and acne concern level, at the same time, which results in less personalised recommendations. Limited integration of artificial intelligence in existing platforms further reduces the accuracy with which products are matched to a user's skin profile.

These limitations motivate DermaVision. The system estimates skin attributes directly from a selfie, considers skin type, skin tone and acne concern jointly, and lets the user review and correct the estimates before products are ranked. In this way it aims to replace trial-and-error selection with a more informed, profile-based process.

6. RELATED WORK

Five studies identified in the project presentation inform the design of DermaVision. Table 1 summarises the methodology, merits and limitations of each study as given in the project material.

Table 1. Literature survey of the studies informing Derma Vision.

Study	Methodology	Merits and relevance	Limitations
[1] Lee et al., Journal of Cosmetic Dermatology, 2024, vol. 23, no. 6, pp. 2066–2077	Deep-learning facial skin image analysis module combined with an ingredient-efficacy scoring model; products are matched to users through content-based filtering.	Considers skin condition and ingredient safety and efficacy together, which reflects real skincare decisions. It is the base paper of this project.	Presented as a research prototype with no explainability, budget filtering, routine generation or progress tracking.
[2] Tan and Le, ICML, 2019, pp. 6105–6114	Compound scaling combined with neural architecture search to build the EfficientNet family (B0–B7).	High accuracy with far fewer parameters and lower computation than earlier CNNs. DermaVision fine-tunes EfficientNet-B0 as the backbone.	Not specific to skin analysis; needs task-specific fine-tuning.
[3] Mühler, GitHub repository, 2020	Wraps pre-trained models (SSD MobileNet, Tiny Face Detector) into a JavaScript API for client-side inference with TensorFlow.js.	Fast, server-independent face detection in the browser. DermaVision uses it for face detection and image-quality validation.	Not designed for skin-condition analysis; depends on lighting and image quality.
[4] Kim et al., Applied Sciences, 2022, vol. 12	CNN-based model trained on labelled facial skin images to classify oily, dry, combination and normal skin types.	Good classification accuracy with a simple end-to-end pipeline. DermaVision reuses the skin-type idea.	Considers skin type only, with no acne, ingredient or recommendation modelling.
[5] Hybrid recommendation system using cosine similarity, Int. J. Adv. Comput. Sci. Appl., vol. 8, 2021	Combines content-based and collaborative filtering and ranks products by cosine similarity between user preference and product feature vectors.	Simple, scalable and effective for personalised recommendation. It inspires the DermaVision recommendation engine.	Limited when product metadata is sparse.

Taken together, these studies show that skin-type classification [4], efficient CNN backbones [2], browser-based face detection [3] and cosine-similarity recommendation [5] are individually established, while the study closest to this work [1] combines skin condition analysis with product recommendation but remains a research prototype. DermaVision combines skin-type, acne-severity and skin-tone estimation with cosine-similarity ranking within one web application.

7. SYSTEM OBJECTIVES

The objectives of DermaVision, as defined in the project presentation, are: (i) to develop an AI-powered web application that provides personalised skincare and makeup recommendations; (ii) to detect facial skin attributes, namely skin tone, skin type and acne concern level, automatically from a user's selfie; (iii) to implement computer vision and deep learning models for skin analysis and classification; (iv) to use face detection to ensure that the captured image is of sufficient quality for reliable assessment; (v) to recommend suitable skincare and makeup products based on the user's skin characteristics and concerns; (vi) to use cosine similarity to match user skin attributes with the most relevant products; and (vii) to provide an interactive interface in which users can review and modify the detected attributes before receiving recommendations.

8. PROPOSED SYSTEM AND ARCHITECTURE

8.1 Overview

DermaVision is a web application in which a user supplies a selfie, receives estimates of skin tone, skin type and acne concern level, and obtains a ranked list of skincare and makeup products. The system uses OpenCV, face-api.js and EfficientNet models for facial skin analysis, allows the user to review, modify and extend the detected attributes, and applies a cosine-similarity engine to match the resulting profile with product features.

8.2 Architectural Layers

The project architecture is organised in seven layers: user, frontend, backend, AI, database, external services and cloud. Table 2 lists the components named for each layer. The architecture is described here in text and tabular form only.

Table 2. System layers and the components named in the project architecture.

Layer	Components named in the project	Role
User	User, web browser, login and registration, image upload, dashboard, community forum, AI chat assistant	Account access, image submission and viewing of results
Frontend (React)	React UI, authentication, image-processing UI, recommendation screen, AR makeup viewer, forum interface	Browser-side interface and client-side processing
Backend	Node.js API, Express server, forum service, chatbot routes, Python AI services	Request handling and access to the AI models
AI	Skin analysis, acne detection model, skin-tone analysis, recommendation engine, virtual makeup effects, AI chatbot	Skin attribute estimation and product matching
Database	MongoDB, recommendation CSV, skin-tone dataset, image dataset	Application data, product data and training data
External services	REST APIs, image-processing libraries, chatbot APIs	Supporting services and libraries
Cloud	Cloud deployment, GitHub repository, model storage	Deployment, version control and model storage

Not all of these components belong to the scope of this paper. The AR makeup viewer, community forum and AI chat assistant appear in the project architecture but are not evaluated here. The paper concentrates on the skin-analysis and recommendation components, which are the subject of the model development reported in the project.

8.3 Processing Sequence

The main processing sequence proceeds as follows. The user registers and signs in, then uploads a photograph or captures a selfie through the webcam. In the browser, face-api.js detects the face and checks that the image is suitable for analysis. The image is then preprocessed with OpenCV. Three predictors operate on the processed input: an EfficientNet-B0 model estimates skin type (Dry, Normal or Oily), a second EfficientNet-B0 model estimates acne severity, and a KNN classifier estimates skin tone from skin colour values. The three outputs are combined into a user skin profile, which the user may review and modify. The recommendation engine then computes the cosine similarity between the skin profile and the feature vector of each product and returns the highest-ranked skincare and makeup products. Analysis results can be stored so that they are available later in the user's dashboard.

9. MATERIALS AND METHODS

9.1 Dataset Collection

Three datasets support the models, as summarised in Table 3. Model development was organised in phases, and dataset collection formed the first stage of the skin-analysis pipeline.

Table 3. Datasets used for model development, as described in the project.

Dataset	Source and format	Content and use
Skin type	Kaggle; folder "Skin_type/train"; JPG/PNG images	Facial skin images categorised as Dry, Normal and Oily; used to train the EfficientNet-B0 skin-type model.
Acne severity	Kaggle; training and validation folders; JPG/JPEG images	Acne images grouped by severity level; used to train the EfficientNet-B0 acne-severity model.
Skin tone	Kaggle; file "skin_tone_dataset.csv"; CSV format	Custom dataset of skin colour values; used to train the KNN skin-tone classifier.

The number of images and records, the class distribution and the sizes of the training and validation sets are not stated in the project material. [Authors to state the exact dataset titles, licences, class-wise counts and train/validation split sizes.]

9.2 Image Preprocessing

Labelled images were organised into class folders and into training and validation sets. Unusable or mislabelled samples were removed before training. Images were resized to the input dimensions required by the models; the project workflow lists 128×128 and 224×224 pixels without specifying which model uses which size. Pixel values were rescaled by a factor of 1/255 so that they are normalised for stable training, and data augmentation was applied to the training images to improve generalisation. The specific augmentation operations are not reported. At inference time, OpenCV is used for image-quality checks and preprocessing of the captured selfie.

9.3 Face Detection

Face detection is performed in the browser with face-api.js [3], which runs pre-trained models through TensorFlow.js. The project uses the Tiny Face Detector model, which executes entirely on the client side, and the library also provides facial landmarks. In DermaVision, face detection serves as the initial stage of the pipeline and checks that a face is present and that the image quality is adequate before skin analysis. Because the library is not designed for skin-condition analysis and depends on lighting and image quality [3], it is used only for this validation stage.

9.4 Skin-Type Classification

Skin type is classified with EfficientNet-B0 [2], selected as the convolutional backbone. The model was trained on labelled facial skin images with transfer learning and fine-tuning to adapt it to the skin-type dataset. The classes are Dry, Normal and Oily. Training used the Adam optimiser with a cross-entropy loss, and performance was monitored on a held-out validation split. The trained model was exported as a TensorFlow SavedModel for use by the application.

9.5 Acne-Severity Classification

Acne severity is classified with a second EfficientNet-B0 model trained on facial acne images organised by severity level. The project workflow describes the output as a graded level of Low, Moderate or Severe. Training and validation datasets were used to evaluate the model, and the trained model was also exported as a TensorFlow SavedModel. The predicted acne concern level is combined with the skin-type and skin-tone estimates to form the user skin profile. The project material also mentions a separate lesion-detection model that is present in the code repository but not yet linked to the application programming interface; it is not part of the evaluation reported here.

9.6 Skin-Tone Classification

Skin tone is classified with a K-Nearest Neighbors (KNN) classifier trained on a custom CSV dataset of skin colour values. KNN is a lightweight approach for structured colour-value data, and the predicted skin tone is used to improve cosmetic and makeup recommendations. The tone categories, the colour features, the number of neighbours and the distance measure are not specified in the project material.

9.7 Cosine-Similarity Product Recommendation

The recommendation engine represents the user's skin profile, comprising the estimated skin type, skin tone and acne concern level together with any additional concerns added by the user, as a feature vector. Each product in the recommendation data, which is stored in CSV format, is represented by a feature vector of the same kind. Products are ranked by the cosine similarity between the two vectors [5], defined as

$$sim(u, p) = (u \cdot p) / (\|u\| \, \|p\|) = \Sigma \, u_i \, p_i / (\sqrt{\Sigma} \, u_i^2 \times \sqrt{\Sigma} \, p_i^2) \quad (1)$$

where u is the user profile vector and p is a product feature vector. Products with the highest similarity values are returned as the recommended skincare and makeup items. The feature encoding, any weighting of attributes and the size of the product catalogue are not specified in the supplied material. The project workflow also describes a check of product ingredients against allergens selected by the user; the implementation status of this check is not established by the material and it is not evaluated here.

9.8 Application Integration

The trained models were saved for integration with the application. According to the project workflow, the models were loaded into a Flask application that exposes an upload endpoint to the rest of the system. The integration chain runs from the React.js interface through face-api.js and OpenCV preprocessing to model inference and finally to the recommendation engine. The project reports that the AI prediction pipeline has been prepared and is ready for application testing and for integration of the recommendation engine. Results of end-to-end application testing are not reported.

10. IMPLEMENTATION TECHNOLOGIES

Table 4 lists the technologies identified in the project presentation and the role each plays in DermaVision.

Table 4. Implementation technologies and their roles.

Technology	Role in DermaVision
React.js	Single-page application framework; used to build the Homepage, Login, Signup, Forum, Profile and AR pages
React Router	Client-side navigation across 12 views
Redux and Redux-Thunk	Shared application state
face-api.js and TensorFlow.js	Browser-side face detection with the Tiny Face Detector model and facial landmark detection; runs entirely on the client side
Node.js and Express	Backend API and server layer
Python AI services	Hosting of the trained models for inference; the project workflow refers to a Flask application with an upload endpoint
OpenCV	Image-quality checks and preprocessing
TensorFlow	Model training and export of trained models as SavedModel files
EfficientNet-B0	CNN backbone for skin-type and acne-severity classification
K-Nearest Neighbors (KNN)	Skin-tone classification from skin colour values
Cosine similarity	Ranking of products against the user skin profile
MongoDB	Database layer of the architecture
CSV files	Product recommendation data and the skin-tone dataset
VS Code and Git	Development environment and version control for the frontend

11. MODEL EVALUATION AND RESULTS

11.1 Project-Reported Metrics

The trained models were evaluated on validation data by comparing predicted classes with the actual labels. Accuracy, precision, recall and F1-score were used. Table 5 gives the values reported in the project presentation; they are project-reported results and have not been independently reproduced for this paper.

Table 5. Project-reported classification results on validation data.

Model	Accuracy	Precision	Recall	F1-score
Skin-type classification using EfficientNet-B0	80%	82%	80%	80.5%
Acne-severity classification using EfficientNet-B0	84%	83%	82%	82.5%
Skin-tone classification using KNN	81%	Not reported	Not reported	Not reported

The acne-severity model reports the highest accuracy (84%), followed by the skin-tone classifier (81%) and the skin-type model (80%). The three figures belong to different tasks, datasets and classifiers and should not be read as a ranking of the methods. The project also reports that the skin-type, acne-severity and skin-tone models were trained, validated and saved for application integration.

11.2 Interpretation and Reporting Gaps

The supplied material does not specify the dataset sizes, the class distributions, the exact validation protocol, the way precision, recall and F1-score were averaged across classes, or any independent test results. The presentation notes that a confusion matrix can be used to analyse class-wise predictions, but no confusion matrices are reported. In addition, the recommendation engine has no reported evaluation of relevance or user satisfaction. One internal point also needs checking: for the skin-type model, the harmonic mean of the reported precision (82%) and recall (80%) is approximately 81.0%, whereas the reported F1-score is 80.5%. This difference may arise from the averaging method used, as the average of per-class F1-scores need not equal the harmonic mean of the average precision and recall, but the figures are reproduced here exactly as supplied. [Authors to confirm the averaging method and metric values.] The results therefore indicate that the models learned useful distinctions on the validation data, and they do not establish clinical validity or performance on unseen users.

12. DISCUSSION

DermaVision joins skin-analysis models and a recommendation method in one workflow. Each component has a distinct role. Face detection acts as a gatekeeper that checks image suitability before analysis, the two EfficientNet-B0 models estimate skin type and acne severity from image content, and the KNN classifier estimates skin tone from colour values, a lighter task for which a simple classifier on structured data is appropriate. The cosine-similarity engine then uses the

three estimates jointly, which addresses the limitation of existing systems that consider these attributes separately or rely on questionnaires.

Allowing users to review and modify the detected attributes, and to add further concerns, is a practical safeguard because the classifiers are imperfect: a wrong estimate can be corrected before it affects the ranking. Compared with the related work, DermaVision extends the skin-type idea of Kim et al. [4] with acne-severity and skin-tone estimation and with a product-recommendation stage, and it follows the base paper [1] in linking skin condition analysis to product matching. Several capabilities that the base paper lacks, such as routine generation and progress tracking, are planned for a later phase of DermaVision and are not yet implemented.

The main value of the system lies in product discovery: it narrows a large product space to items that match an estimated skin profile and does so quickly in a web interface. The quality of the ranking, however, depends on the accuracy of the three estimates and on how well product features are described. Because the recommendation stage has not been evaluated quantitatively, claims about its effectiveness cannot be made from the present material.

13. LIMITATIONS AND RESPONSIBLE USE

13.1 Limitations

Several limitations follow from the project material. First, the evaluation details are incomplete: dataset sizes, class distributions, the validation protocol and independent test results are not reported, and the metrics were computed on validation data. Second, the system depends on image quality; face-api.js is sensitive to lighting and image quality [3], and the same conditions can affect the skin-type, acne and skin-tone estimates. Third, the datasets come from Kaggle sources, and their label quality and their coverage of different skin tones, ages and imaging conditions are not documented, so performance across user groups has not been assessed. Fourth, the skin-tone classifier works on colour values whose extraction method is not detailed, and colour values are themselves influenced by illumination. Fifth, the recommendation stage has not been evaluated for relevance or user satisfaction, and the encoding of product features is not specified. Sixth, a lesion-detection model exists in the project repository but is not yet linked to the application. Finally, the architecture and technology listing name MongoDB as the database, while the workflow description also mentions Firebase services for authentication and data storage. [Authors to confirm the data-management configuration of the final system.]

13.2 Responsible Use

The outputs of DermaVision are model estimates intended to support product selection. They are not medical diagnoses, and the system does not replace the assessment of a qualified

dermatologist. Users with persistent, severe or painful skin conditions should seek professional advice rather than rely on an acne-severity estimate. The facility to modify detected attributes should be kept so that users stay in control of the profile used for recommendations. Facial images are sensitive personal data. Detecting the face in the browser keeps that step on the client side, but the supplied material does not describe how uploaded images and stored results are protected, and any deployment should define and document its data-protection practices.

14. CONCLUSION

This paper presented DermaVision, a web application that estimates skin type, skin tone and acne concern level from a selfie and recommends skincare and makeup products by cosine similarity. The analysis pipeline combines browser-side face detection with face-api.js, OpenCV preprocessing, two EfficientNet-B0 classifiers and a KNN skin-tone classifier. On validation data the project reports accuracies of 80% for skin type, 84% for acne severity and 81% for skin tone. The contribution is an integrated workflow that considers several skin attributes jointly and lets users review them before recommendations are made. Because dataset sizes, the validation protocol and independent test results are not fully specified, and the recommendation stage has not been evaluated, the reported figures should be treated as preliminary and as estimates rather than diagnoses.

14.1 Future Work

Table 6 distinguishes the capabilities that the project material reports as completed from those that are planned or whose status is not established.

Table 6. Completed and planned capabilities according to the project material.

Capability	Status reported in the project material
Skin-type and acne-severity models (EfficientNet-B0)	Trained, validated on validation data and saved
Skin-tone model (KNN)	Trained, validated on validation data and saved
AI prediction pipeline	Prepared; reported as ready for recommendation-engine integration and application testing
Cosine-similarity recommendation engine	Designed; end-to-end evaluation not reported
Lesion-detection model	Present in the repository; not yet linked to the application
E-commerce product integration and price comparison	Planned (Phase 3)
Weekly skincare routine tracking	Planned (Phase 3)
Skin progress monitoring	Planned (Phase 3)
Ingredient-allergen check, AR makeup viewer, community forum, AI chat assistant	Described in the architecture or workflow; status not established and outside the scope of this paper

The planned Phase 3 work comprises three features. E-commerce integration will provide product options and shopping links from Amazon, Myntra and Nykaa and will help users compare prices for similar skincare products. Weekly routine tracking will record morning and evening routines and the products used each day and will show them as a date-wise history. Skin progress monitoring will compare earlier and recent skin-analysis results to identify changes over time and will summarise routine consistency. In addition to these features, further work should evaluate the models on an independent test set with confusion matrices and class-wise metrics, document the datasets, assess performance across diverse skin tones and imaging conditions, measure the relevance of the recommendations with users, and complete and test the integration of the models with the application.

15. ACKNOWLEDGEMENTS

The authors thank K.L.N. College of Engineering and the Department of Computer Science and Engineering (Internet of Things) for their support. The authors are grateful to their project guide, Ms. V. Merlinfreeda, for her guidance throughout the DermaVision project, and acknowledge the publicly available datasets and the open-source tools that supported the development.

16. REFERENCES

1. Lee, J., Yoon, H., Kim, S., Lee, C., Lee, J., and Yoo, S., “Deep Learning-based Skin Care Product Recommendation: A Focus on Cosmetic Ingredient Analysis and Facial Skin Conditions,” Journal of Cosmetic Dermatology, vol. 23, no. 6, pp. 2066–2077, 2024. (Base Paper)
2. Tan, M., and Le, Q. V., “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks,” Proceedings of the 36th International Conference on Machine Learning (ICML), pp. 6105–6114, 2019.
3. Mühler, V., “face-api.js: JavaScript API for Face Detection and Face Recognition in the Browser,” GitHub Repository, 2020. [Repository URL to be added.]
4. Kim, J., et al., “Image-Based Skin Type Classification Using Deep Learning,” Applied Sciences, vol. 12, 2022. [Full author list, article number and DOI to be verified.]
5. “A Hybrid Recommendation System Using Cosine Similarity for Personalized Product Recommendation,” International Journal of Advanced Computer Science and Applications, vol. 8, 2021. [Authors, issue and page numbers not supplied in the source; to be verified.]