International Journal of Creative and Open Research in Engineering and Management
ISSN: 3108-1754 (Online)
Volume 02 Issue 10 October-2026 | Impact Factor: 3.5



AI-Based Skin Disease Prediction System Using ConvNeXt-Tiny with Explainable Visualization and Phase Guidance

Balkrishna
1RF23CS044
Dept. of Computer Science and Engineering
RV Institute of Technology and Management
Bengaluru, India

Arun M
1RF23CS038
Dept. of Computer Science and Engineering
RV Institute of Technology and Management
Bengaluru, India

Adarsh
1RF23CS006
Dept. of Computer Science and Engineering
RV Institute of Technology and Management
Bengaluru, India

Arban Reyaz Khan
1RF23CS033
Dept. of Computer Science and Engineering
RV Institute of Technology and Management
Bengaluru, India

Dr. Syeda Ayesha Unisa
Assistant Professor
Dept. of Computer Science and Engineering
RV Institute of Technology and Management
Bengaluru, India

How to Cite this Article:
Balkrishna, , M, A., Adarsh, & Khan, A. R. (2026). AI-Based Skin Disease Prediction System Using ConvNeXt-Tiny with Explainable Visualization and Phase Guidance. International Journal of Creative and Open Research in Engineering and Management, 02(10), 1-9. https://doi.org/10.55041/ijcope.v2i10.195

License:
This article is published under the terms of the Creative Commons Attribution 4.0 International License (CC BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author(s) and the source are credited.
© The Author(s). Published by International Journal of Creative and Open Research in Engineering and Management.







https://doi.org/10.55041/ijcope.v2i10.195

***Abstract*—Skin diseases are common, but seeing a dermatologist is not easy for everyone, especially in under-served regions. Many deep-learning classifiers are heavy, hard to interpret and stop at a class label. We present MediCare AI, a lightweight screening system that sorts skin images into seven disease categories by fully fine-tuning an ImageNet-pretrained ConvNeXt-Tiny with AdamW, cosine learning-rate decay, label smoothing and early stopping. Gradient-weighted Class Activation Mapping shows which parts of the image drove each prediction. A Streamlit app accepts uploaded photos and live camera frames, reports the confidence and a rough three-level visible phase, and gives treatment suggestions and preventive advice through a generative service, with a local offline fallback. A single network reached roughly 80%–90% validation accuracy. The tool is meant for education and early screening, not as a replacement for professional diagnosis.**

***Index Terms*—skin disease classification, ConvNeXt, transfer learning, Grad-CAM, explainable artificial intelligence, Streamlit.**

I. INTRODUCTION

Skin problems are everyday health complaints. Acne, eczema, psoriasis and fungal, bacterial or viral infections send huge numbers of people to the clinic, and in most cases the first step is simply a dermatologist looking at the skin. The trouble is that dermatologists are not equally available everywhere. In rural and under-served areas a patient may travel a long way for a short consultation, and the wait often lets a treatable condition get worse. Deep learning has already shown that an image classifier can match specialists on well-defined lesion tasks [1], and recent surveys, including one from 2026, agree that convolutional networks, transfer learning and explainable AI now sit at the centre of this field [2], [3].

Many of the most accurate systems, though, depend on ensembles or attention-heavy designs that are costly to run and awkward to deploy [4], [5], [6]. Others cover only a handful of diseases or train on small datasets [7], [8]. Very few show why the model decided what it did, and almost none go beyond a class label to tell the user how advanced the condition looks or what to do next [2]. We built MediCare AI to close these gaps. In this paper we make the following contributions:

- a single-network classifier for seven common skin disease categories, made by fully fine-tuning an ImageNet-pretrained ConvNeXt-Tiny [9];
- visual explanations of each prediction using Gradient-weighted Class Activation Mapping (Grad-CAM) [10];
- an indicative three-level estimate of how advanced the visible condition is, plus generated health information with a local offline fallback; and
- a Streamlit web application that takes uploaded photos or a live camera feed, with model export and a ready-to-run container.

II. RELATED WORK

The first computer-aided systems relied on hand-crafted colour, texture and shape features fed to shallow classifiers. They worked on tidy data but struggled when lighting or cameras changed [11], [12]. Convolutional networks removed the need to design features by hand, and because medical datasets are small, most people start from a network pretrained on a large photo collection [13]. Fine-tuned VGG, ResNet, DenseNet, MobileNet and EfficientNet models are all common in the skin-disease literature [14], [15], [7], [16], [17], and some studies add special loss functions or dermoscopic images to handle many classes [18], [19].

To push accuracy higher, researchers have turned to ensembles and feature fusion [20], [6], [21], chains of specialised models [4], and attention, capsule or transformer networks [22], [5], [23], [24]. These help, but each step adds computation and engineering effort. On the explanation side, gradient-based class activation maps [10] have been paired with strong classifiers so that clinicians can see what the network looked at [24]. Our approach is deliberately simpler: one efficient backbone, with explanation, a rough severity indication and guidance built around it in a single application.

III. PROPOSED SYSTEM

A. Architecture

Fig. 1 gives the overall picture. The user uploads a photo or points a camera at the skin. The image is resized and normalised, and the ConvNeXt-Tiny classifier returns a disease class with a confidence score. A second stage then labels the condition as phase 1 (mild), 2 (moderate) or 3 (severe), a generative service writes out medical information, treatment suggestions and preventive advice, and the results screen shows all of this next to a Grad-CAM heatmap.

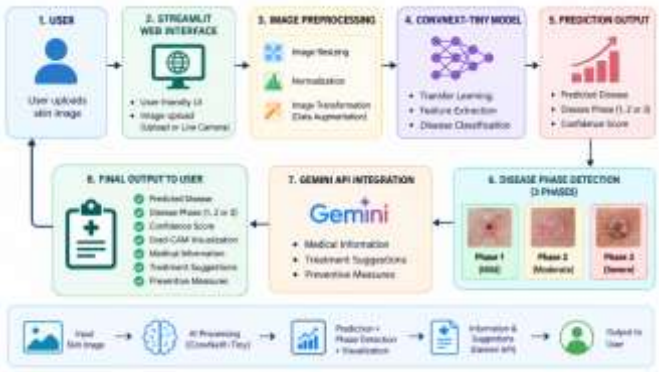


Fig. 1. System architecture of the proposed skin disease prediction system.

B. Dataset and Pre-processing

We arranged the labelled images in the standard ImageFolder layout, one folder per class, for seven categories: acne and rosacea; cellulitis, impetigo and other bacterial infections; eczema; psoriasis, lichen planus and related diseases; tinea, ringworm, candidiasis and other fungal infections; urticaria; and warts, molluscum and other viral infections. Blurred and noisy pictures were removed, and the rest was split roughly 80% for training and 20% for validation. All images were resized to 224 × 224 pixels and normalised. Training images were also randomly flipped, rotated by up to 15° and colour-jittered so the model would not depend on pose or lighting.

C. Model

ConvNeXt-Tiny takes a standard residual network and updates it with ideas from vision transformers: a patchify stem, 7 × 7 depthwise convolutions, an inverted bottleneck, layer normalisation and GELU activations [9]. As Fig. 2 shows, it has four stages with 3, 3, 9 and 3 blocks and 96, 192, 384 and 768 channels, which comes to about 28 million parameters. We swapped the original 1000-class head for a linear layer with C = 7 outputs. Writing z_k for the raw score (logit) of class k, the class probabilities are given by softmax:

$$p_k = \frac{\exp(z_k)}{\sum_{j=1}^{C} \exp(z_j)} \tag{1}$$

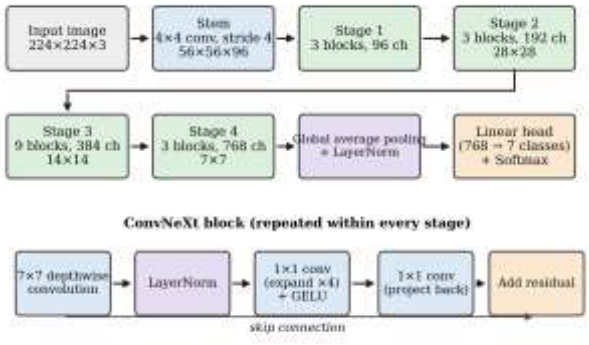


Fig. 2. ConvNeXt-Tiny structure with the modified seven-class head.

D. Training

Every layer was fine-tuned. We used AdamW, which applies weight decay separately from the gradient update [25], together with a cosine learning-rate schedule [26] and label smoothing [27] to keep the model from becoming overconfident. With smoothing factor ε and C classes, the training target for an image with true class y is $q_k = (1-\varepsilon)\delta_{k,y} + \frac{\varepsilon}{C}$, where $\delta_{k,y}$ is 1 when k = y and 0 otherwise. The loss is

$$\mathcal{L} = -\sum_{k=1}^{C} q_k \log p_k \tag{2}$$

Training stopped once validation performance had not improved for seven epochs, and we kept the best checkpoint. The full settings are in Table I.

TABLE I
TRAINING CONFIGURATION

Parameter	Value
Backbone	ConvNeXt-Tiny (ImageNet-1K)
Input size	224 × 224
Optimiser	AdamW, lr 1e-4, decay 1e-2
Scheduler	Cosine, min lr 1e-6
Loss	Cross-entropy, smoothing 0.1
Batch / epochs	32 / up to 30
Early stopping	Patience 7

E. Explainability and Phase Guidance

To see what drove a prediction, Grad-CAM takes the gradient of the pre-softmax score y^c of the predicted class c with respect to the feature maps A^k of the last stage, averages it over the Z spatial positions (i, j) to get a weight α_k^c for each map, and combines the maps into a heatmap L^c [10]:

$$\alpha_k^c = \frac{1}{Z}\sum_i \sum_j \frac{\partial y^c}{\partial A_{ij}^k} \tag{3}$$

$$L^c = \mathrm{ReLU}\left(\sum_k \alpha_k^c A^k\right) \tag{4}$$

In (4), the ReLU keeps only the regions that raise the score of the predicted class. We upsample this map and lay it over the original photo. One thing worth being upfront about: the network is trained on disease labels only, so it cannot predict phase. The phase comes from a separate stage that gives a three-level visible-severity estimate, with a simple rule-based guide as an offline option. We always present it as a rough indication, not a medical staging. The health information comes from the Google Gemini service, and when there is no network the app reads it from a local JSON file instead.

IV. IMPLEMENTATION

The system is written in Python using PyTorch and Torchvision [28], OpenCV for the camera and heatmap overlay, and Streamlit for the interface. Training, evaluation (accuracy, precision, recall, F1-score and a confusion matrix), command-line inference, Grad-CAM and live-camera classification each live in their own module. In camera mode we average the probabilities over the last few frames so the label does not flicker. The interface shows the top prediction with a colour-coded confidence bar, the gap between the first and second guesses, a probability chart for all classes, the heatmap, the phase and the generated advice. The trained model can be exported to TorchScript and ONNX, and a Docker image serves the app on port 8501.

V. RESULTS AND DISCUSSION

A. Quantitative Performance

On the validation set the classifier reached an overall accuracy of roughly 80% to 90% across the seven categories, using one network and no ensemble. As you would expect, the mistakes cluster among conditions that look alike. Eczema, psoriasis and fungal infections all show up as red, scaly patches and are the classes most often mixed up.

B. Qualitative Results

Fig. 3 shows the web app analysing an urticaria photo. The top of the screen gives the predicted disease with a confidence bar, and below it the Gemini service lists medical treatments, home care and doctor tips. Table II shows the ranked output for



the same image. The top class beat the runner-up by 85.3 percentage points, so this was a confident call.

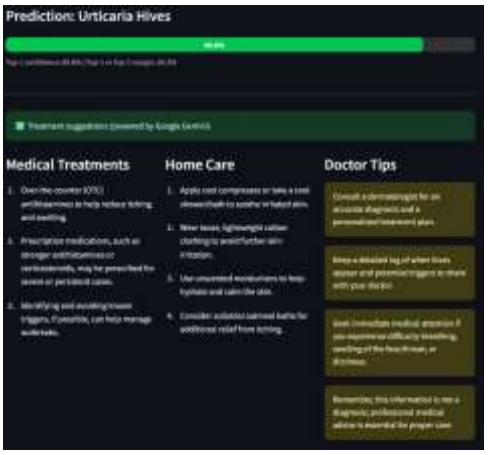


Fig. 3. Prediction, confidence and generated treatment suggestions in the web application.

TABLE II
RANKED PREDICTION FOR THE EXAMPLE IMAGE

Rank / category	Probability
1. Urticaria	88.8%
2. Fungal infections (tinea, ringworm, candidiasis)	3.5%
3. Psoriasis, lichen planus and related	1.8%

The Grad-CAM overlay in Fig. 4 puts the warm areas right on the raised wheals, which tells us the model was looking at the lesion and not at the background. Fig. 5 shows the probability chart across all seven classes, where one bar clearly dominates, so a user can see at a glance how sure the model is.

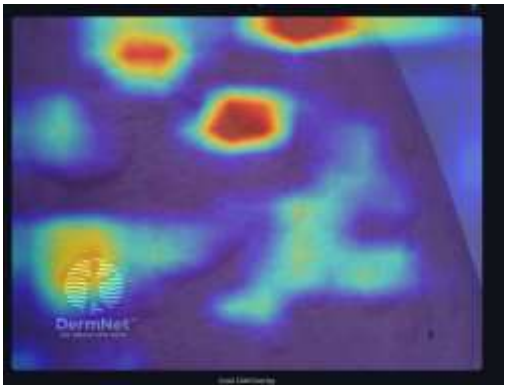


Fig. 4. Grad-CAM overlay highlighting the affected region for an urticaria image.

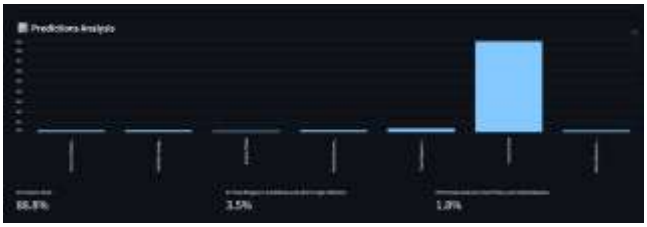


Fig. 5. Probability chart over the seven disease categories with the top three predictions.

Finally, Fig. 6 shows the phase panel, which described this image as phase 2, a clearly active condition with several prominent raised wheals.



Fig. 6. Visible phase estimate shown for the example image.

C. Discussion and Limitations

A fine-tuned ConvNeXt-Tiny gives useful accuracy without much computing cost, and the heatmap, the probability gap and the phase note together let a user judge how far to trust each result. There are clear limits. Results depend on how large, balanced and clean the dataset is. Look-alike classes still get confused. Malignant lesions are not covered. The phase is estimated, not learned, and generated text can contain mistakes. The tool is meant for education and early screening, and it should never replace a visit to a dermatologist.

VI. CONCLUSION AND FUTURE WORK

We presented a lightweight, explainable skin disease screening system that sorts photos into seven categories using a fine-tuned ConvNeXt-Tiny, shows its reasoning with Grad-CAM, gives a rough phase, and offers practical guidance through a web app with upload and live-camera modes. Next we plan to gather a larger and more varied dataset with more diseases and skin tones, train a model on phase labels, add calibrated uncertainty and a way to reject out-of-scope images, run inference on phones, and test the system with dermatologists.

REFERENCES

[1] A. Esteva *et al.*, “Dermatologist-level classification of skin cancer with deep neural networks,” *Nature*, vol. 542, no. 7639, pp. 115–118, Feb. 2017.

[2] Z. R. Abdulqader and A. R. Abrahim, “Deep learning-based skin disease detection and classification: A systematic literature review,” *Dasinya J. Eng. Informat.*, vol. 2, no. 1, 2026.

[3] A. Aboulmira, H. Hrimech, and M. Lachgar, “Skin diseases classification with machine learning and deep learning techniques: A systematic review,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 15, no. 10, 2024.

[4] M. Badr, A. Elkasaby, M. Alrahmawy, and S. El-Metwally, “A multi-model deep learning architecture for diagnosing multi-class skin diseases,” *J. Imaging Informat. Med.*, vol. 38, pp. 1776–1795, 2025.

[5] K. Behara, E. Bhero, and J. T. Agee, “An improved skin lesion classification using a hybrid approach with attention-guided capsule networks,” *Diagnostics*, vol. 14, no. 6, p. 636, 2024.

[6] Y. Yang *et al.*, “Skin disease classification using transfer learning model and fusion strategy,” *Intell.-Based Med.*, vol. 12, 2025.

[7] N. Tarisafitri and A. Aljabar, “Classification of skin diseases using digital image processing with MobileNetV2 architecture,” *Nusantara J. Artif. Intell. Inf. Syst.*, vol. 1, no. 1, pp. 35–44, 2025.

[8] F. Ahmmed *et al.*, “Skin disease detection and classification of actinic keratosis and psoriasis utilizing deep transfer learning,” arXiv:2501.13713, Jan. 2025.

[9] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, “A ConvNet for the 2020s,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, New Orleans, LA, USA, 2022, pp. 11976–11986.

[10] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, “Grad-CAM: Visual explanations from deep networks via gradient-based localization,” in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, 2017, pp. 618–626.

[11] A. M. Patel, P. Parmar, N. Thakkar, and H. M. Barot, “Skin disease classification using machine learning algorithms: A review,” 2023.

[12] M. Zahid, M. Rziza, and R. Alaoui, “Skin lesion classification using hybrid feature extraction based on classical and deep learning methods,” *BioMedInformatics*, vol. 5, no. 3, p. 41, 2025.

[13] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Miami, FL, USA, 2009, pp. 248–255.

[14] A. Kumar, V. Yadav, S. Kale, P. Satpute, and M. N. Vharkate, “Classification of various skin diseases by using deep learning,” *Int. Res. J. Adv. Eng. Hub*, vol. 3, no. 7, pp. 3265–3269, 2025.

[15] M. Y. Shams *et al.*, “Skin disease classification: A comparison of ResNet50, MobileNet, and EfficientNet-B0,” 2025.

[16] H. S. Khalifa and S. F. Abdullah, “Skin disease classification using CNN deep learning technique,” *Current Sci. Int.*, vol. 14, no. 2, pp. 121–128, 2025.

[17] B. Durgabhavani, B. Chandana, S. Sandhyarani, G. Lavanya, G. Srikanth, and N. Bhaskar, “Classification of skin disease using CNN,” in *Proc. Int. Conf. Evol. Algorithms Soft Comput. Techn. (EASCT)*, 2023.



[18] S. Zhi, Z. Li, X. Yang, K. Sun, and J. Wang, "A multiclassification model for skin diseases using dermatoscopy images with Inception-v2," *Appl. Sci.*, vol. 14, no. 22, p. 10197, 2024.

[19] S. G. Malik *et al.*, "High-precision skin disease diagnosis through deep learning on dermoscopic images," *Bioengineering*, vol. 11, no. 9, p. 867, 2024.

[20] V. Kumar and J. Rani, "Human skin disease detection and classification using ensemble learning," *J. Recent Innov. Comput. Sci. Technol.*, vol. 2, no. 2, 2025.

[21] S. R. Shankar *et al.*, "Skin disease classification using multi-model optimization and augmentation," *IARJSET*, vol. 12, no. 6, 2025.

[22] M. N. Noor, F. Haneef, I. Ashraf, and M. Masud, "Enhanced skin disease classification via dataset refinement and attention-based vision approach," *Bioengineering*, vol. 12, no. 275, 2025.

[23] S. A. Hanuma, A. Dey, and M. A. Kabir, "An attention-guided deep learning approach for classifying 39 skin lesion types," 2025.

[24] J. Mohan, A. Sivasubramanian, V. Sowmya, and V. Ravi, "Enhancing skin disease classification leveraging transformer-based deep learning architectures and explainable AI," *Comput. Biol. Med.*, 2025, doi: 10.1016/j.compbiomed.2025.110007.

[25] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, New Orleans, LA, USA, 2019.

[26] I. Loshchilov and F. Hutter, "SGDR: Stochastic gradient descent with warm restarts," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, Toulon, France, 2017.

[27] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, 2016, pp. 2818–2826.

[28] A. Paszke *et al.*, "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 32, Vancouver, Canada, 2019, pp. 8024–8035.