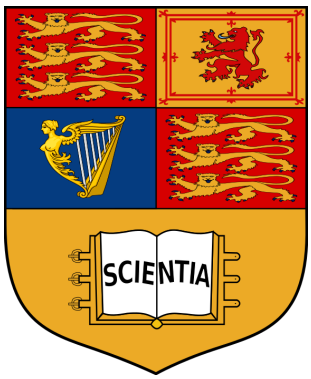


Developing a Machine Learning Model to Detect Eczema Herpeticum

Department of Bioengineering

Imperial College London

Supervisor:

Dr. Reiko Tanaka

Authors:

Donghyun Kim, Jiho Shin, Jessica Utomo, Elif Civelekoglu,
Yousuf Yaqub, Anika Ip, Yushu Gao

15 April 2025

Developing a Machine Learning Model to Detect Eczema Herpeticum

Donghyun Kim, Jiho Shin, Jessica Utomo, Elif Civelekoglu,

Yousuf Yaqub, Anika Ip, Yushu Gao

Abstract

Eczema Herpeticum (EH) is a rare and life-threatening skin infection that frequently affects individuals with preexisting dermatological conditions such as eczema. Early and accurate diagnosis is critical but remains challenging due to similarities in symptoms with other skin conditions and the limitations of current diagnosis methods with speed and precision. This study proposes machine learning-based image classification models for accurate and rapid EH detection.

Approaches to artificially expand the dataset were implemented to address the challenge of limited data availability, including data augmentation and Generative Adversarial Networks (GAN) to generate synthetic images. A custom-designed Convolutional Neural Network (CNN) and transfer learning-based pre-trained CNNs were trained on the dataset, with notable performance achieved. Our InceptionV3-based model achieved 98.1% accuracy and sensitivity, indicating strong potential for clinical application. Additionally, our custom CNN model achieved an accuracy of 97.7% and sensitivity of 96.2%, and demonstrated higher adaptability and compatibility with GAN-generated synthetic data shown by increased performance upon addition of synthetic train data. Apart from showing promising results for clinical integration, the findings of this study holds significant implications for future research and applications - particularly in the classification of rare skin diseases beyond EH where limited datasets can be expanded with synthetic images. Transfer learning models demonstrate strong potential with its higher accuracy when trained with limited real datasets, whereas the custom CNN may present a viable alternative when augmented with synthetic images.

1 Introduction

1.1 Project Motivation and Clinical Significance

Eczema Herpeticum (EH) is a viral skin infection caused by the herpes simplex virus. This occurs mainly in individuals with pre-existing skin conditions such as atopic dermatitis (eczema). EH manifests itself mainly as small, painful, and itchy blisters and erosions (Lin, 2023). The infection, though rare (occurs from less than 3% of atopic dermatitis patients (Leung, 2013)), is potentially life-threatening with its ability to rapidly worsen. Skin lesions are prone to bacterial infection and their ability to spread beyond the skin may cause severe damage, leading to organ failure and death (Xiao and Tsuchiya, 2021).

Current clinical diagnoses are limited to viral swabs, viral culture and antibody stains, as well as microscopy and culture, but it currently takes 14-21 days for results (Hull University Teaching Hospitals NHS Trust, 2022). Hospitalisation and subsequent monitoring can occur if a specialist suspects EH, but it is commonly confused with other skin infections such as chickenpox or impetigo, or a simple flare-up of eczema (Patient.info, 2023). Its rarity leads to severe under-diagnosis, and therefore clinicians are also less experienced in identifying and diagnosing it, increasing the chances of misdiagnosis even if they are consulted. These challenges highlight the need for automated and quick screening tools to aid in early detection of EH.

Our project aims to use Machine Learning (ML) to provide a telemedicine aid for EH screening. This is prompted by recent developments in computer vision and machine learning, which achieved expert-level performance in image-based disease classification through convolutional neural networks (CNNs). AI-driven eczema monitoring models already exist, such as EczemaNet (Attar et al., 2023), a CNN system that can detect eczema lesions and assess severity from photographs. However, current EczemaNet does not screen for EH, and our project aims to develop this missing EH screening module which can be integrated into existing systems. Ideally, the integration of this detector would help people performing routine eczema evaluations at home detect EH, enabling timely medical intervention.

1.2 Existing Literature

Previous studies have explored ML applications in dermatology, particularly using CNNs for skin disease classification (Han et al., 2020). They trained a model with images of 174 skin diseases and compared their results to 47 professionals. It was found that their model performed on par with dermatology residents but below specialists, showing that these models are feasible and relatively consistent with human diagnosis. Biases relevant to our project, such as those introduced by lower-quality images and those stemming from non-characteristic locations should be noted, especially as they may affect our results.

Shanthi, T. et al also used a CNN architecture for automatic diagnosis of skin diseases (Shanthi et al., 2020). It classified images into four categories: acne, seborrheic keratosis, eczema herpeticum, and urticaria. Their model achieved an impressive accuracy in distinguishing these skin conditions, indicating that even EH lesions can be recognized with high

accuracy by a dedicated CNN when given sufficient training data. It also greatly outperformed more conventional machine learning techniques and provided a comprehensive approach to health support by incorporating patient-reported symptoms, but found shortcomings such as dataset variability from their small test image set (69 images) and high processing needs, with their model needing 2-3 days to train.

There has also been efforts to develop practical eczema area detection systems. Rahman, A. et al. introduced EczemaNet2, a two-stage CNN pipeline for automated eczema assessment, focused on classifying eczema severity within an uploaded image (Attar et al., 2023). It introduced segmentation of the photograph at the pixel level, improving segmentation accuracy for eczema assessment and therefore the quality of the severity prediction. This proves that specialised models are definitely feasible. However, this current model focuses on eczema detection and severity scoring, and does not explicitly identify EH or other infections that may occur.

Other key challenges involved in developing and testing an EH classifier is the lack of large and labelled image datasets due to its rareness. This is particularly notable as previous studies using CNNs to identify skin disorders (Han et al., 2020) have used more than 220,000 images, greatly outnumbering our available data of 474 EH images. Many studies employ transfer learning to address this, wherein a CNN pretrained on a massive general image dataset is fine-tuned on the given medical images. (Shanthi et al., 2020) Sadik, R. et al. (Sadik et al., 2023) used popular CNN architectures with this transfer learning method for skin diseases diagnosis including eczema, and achieved an impressive average classification accuracy of 96% (MobileNet) and 97% (Xception), but again lacks classification of EH.

Robb, E. et al. (Robb et al., 2020) found that Few-Shot Generative Adversarial Networks (FSGANs) provide potential solutions to the challenge, investigating its ability to decrease dataset sizes needed for training by generating images based on the initially given (seed) set of images. This has seen promising and relatively consistent results, ranking either best or second-best when compared quantitatively with other methods such as SSGAN, TransferGAN and FreezeD. However, it should be noted that the use of FSGANs introduces new potential issues such as bias and diversity removal, as well as leaving the model inaccurate toward unseen lesion types.

1.3 Project Overview and Objectives

This project focuses on creating a model with high sensitivity and accuracy to effectively screen EH, a rare but potentially life-threatening condition. The model should generalize well to unseen images and be computationally efficient. Multiple machine learning approaches will be incorporated to find the optimal approach and given the small dataset (474 EH images), the effect of using synthetic images will be investigated. Therefore, the key objectives of our project are:

- Create an accurate and sensitive EH screening model which performs well on unseen data.
- Investigate model performance based on the use of synthetic images
- Build a computationally efficient EH screening model.
- Evaluate and compare different machine learning approaches.

2 Methods

2.1 Image Preprocessing

2.1.1 Data Distribution and Augmentation

Starting with 474 EH images, 20% is first allocated to the test dataset. To enhance model performance (Wang and Perez, 2017), remaining data is augmented by horizontal and vertical flips, as well as 90, 180 and 270 degree rotations. These five total image transformations create a five-fold increase in the dataset, and this is followed by a 90:10 split into the training and validation datasets, respectively. Finally, synthetic images are added to the train dataset in 1:2 and 1:5 ratios of real to synthetic images to observe its effect. Creating the non-EH datasets with corresponding number of images from the Skin Condition Image Network (SCIN) (Ward et al., 2024), created by Google and Stanford Medicine, allowed us to have a balanced dataset of EH and non-EH images.

2.1.2 Two Test Datasets

Two test datasets were used to evaluate model performances: the balanced test dataset and biased test dataset. The balanced test dataset consists of 94 EH images stemming from various public sources and 94 non-EH images from the SCIN image dataset. On the other hand, the biased test dataset contains 188 non-EH images from the Softened Water Eczema Trial (SWET) (Thomas et al., 2011) image dataset.

The best models were selected based on a balanced test dataset, which was further assessed using a biased test dataset. This was done to investigate overfitting on SCIN images and to evaluate model performance under more real-world conditions, as EH is a rare disease.

2.1.3 Few-Shot Generative Adversarial Network (FSGAN)

The risk of overfitting is especially high, given our limited training data. To mitigate this, we imple-

ment the FSGAN method, which introduces more images to train the model with. In general, General Adversarial Networks (GAN) consists of two neural networks, the generator the and discriminator. The generator produces synthetic images that attempt to mimic the distribution of the training data, while the discriminator evaluates both real and generated images, learning to distinguish between them. Through this adversarial process, the generator improves iteratively by attempting to fool the discriminator, resulting in increasingly realistic outputs.

FSGANs are specifically developed to be suitable for small datasets. NVIDIA Stylegan2-ADA-Pytorch was used as a FSGAN due to its proven effectiveness in data-constrained settings. This model incorporates Adaptive Discriminator Augmentation (ADA) (Karras et al., 2020), a technique that dynamically adjusts data augmentation strength based on the discriminator's feedback. This specific design allows dynamic augmentation which depends on the performance of the discriminator to prevent overfitting.

Three hyperparameters were adjusted and tuned during training through manual selection.

- Gamma - controls regularisation to prevent the discriminator from overfitting to real data
- Target - controls augmentation to increase dataset
- Learning Rate - controls magnitude of weight updates during training

The Frechet Inception Distance (FID) metric was used to assess the quality of developed GANs by measuring the similarity and diversity of generated synthetic images, evaluating their effectiveness for training image classification models (M. et al., 2017). A lower FID score indicates a closer resemblance to real images, signifying a more effective GAN.

Additionally, image selection criteria are used to filter out poor, unrealistic images in two ways: VGG16 feature vector similarity (Irfan Rasyid, 2023) and the GAN discriminator feedback (Fan, 2024). Using VGG16 (Simonyan and Zisserman, 2015), feature vectors are extracted from each of the synthetic images and a cosine similarity to the average of the real EH image feature vectors is calculated. This results in a measure of how similar the image is to the real images, in terms of features such as texture and colour. On the other hand, the discriminator feedback is a score that indicates how close the discriminator thinks the synthetic image is to a real image. Images with scores that met a minimum threshold were selected, where the threshold used was the mean of the scores calculated for the real images subtracted by two standard deviations. This meant that we chose images with any score that fell within the range of where >95% of real image scores would sit, assuming a normal distribution (Swinscow, 2021).

2.2 CNNs

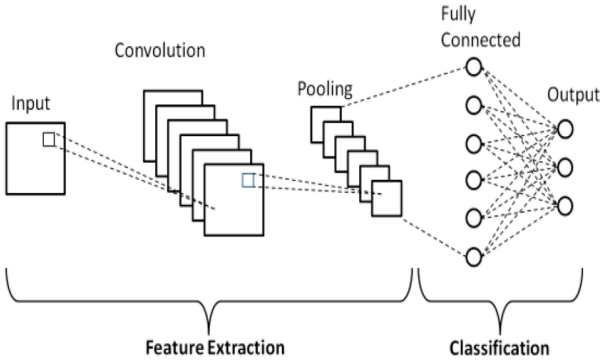


Figure 1: Schematic diagram of a basic CNN architecture (Phung and Rhee, 2018)

Our classification model is built on a Convolutional Neural Network (CNN) architecture (LeCun et al., 1998). CNNs are a deep-learning algorithm, commonly used to analyse visual data such as photos and videos, making it useful for the application of this project. The CNN works by learning hierarchical feature representation via convolutional and fully connected layers which is simply represented by Figure 1.

2.2.1 Convolutional and Pooling Layers

When an image is passed through a Convolutional Neural Network (CNN), its features are extracted using small learnable matrices known as kernels. These kernels slide across the input image, multiplying its values against the corresponding pixel values of the image, summing them together. This generates a feature map that highlights specific visual patterns present in the image. Max-pooling layers are used to reduce the spatial dimensions of the feature maps while preserving the most important information.

2.2.2 Fully-connected Layers

After the convolutional and pooling layers extract and condense features from the input data, the resulting feature maps are flattened into a one-dimensional feature vector and passed into the Fully Connected (FC) layers. These layers interpret the extracted features and perform the final classification.

2.2.3 Other Functional Layers

Rectified Linear Unit (ReLU) activation functions were applied after each convolutional layer to introduce non-linearity into the network. The ReLU function is defined as follows (Nair and Hinton, 2010):

$$\mathrm{ReLU}(x) = \max(0, x) \tag{1}$$

where it outputs the input x if $x > 0$, and 0 otherwise. It introduces non-linearity into the model and helps the network learn complex patterns efficiently.

Batch normalisation was implemented to prevent internal covariate shift by normalising layer inputs, which stabilises and accelerates training (Ioffe and Szegedy, 2015). The normalization is given by:

$$\hat{x} = \frac{x - \mu}{\sqrt{\sigma^2 + \epsilon}} \tag{2}$$

where μ and σ^2 are the batch mean and variance, and ϵ is a small constant to prevent division by zero.

Dropout layers were also included in the FC layers to reduce overfitting. During training, a fraction of the neurons are randomly deactivated, promoting redundancy and robustness in learned representations.

The final output layer uses a Softmax activation function to produce a probability distribution over the output classes (Bridle, 1990):

$$\text{Softmax}(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}} \tag{3}$$

In the case of binary classification, this effectively reduces to a single output neuron producing a probability score indicating the presence or absence of EH.

2.2.4 Hyperparameter tuning

To get the best possible performance for the CNN model, hyperparameter tuning strategies were implemented. This allowed us to select the optimal configuration of parameter settings which includes the learning rate, number of convolutional layers, fully connected layers, and the optimizer - RMSprop (Tieleman and Hinton, 2012), ADAM (Kingma and Ba, 2015), and SGD (Robbins and Monro, 1951) - which defines the magnitude and speed of the learning rate update.

Optuna (Optuna, 2018) is a hyperparameter optimisation framework that uses Bayesian optimisation, to efficiently search the hyperparameter space. Given an objective function, it suggests parameter sets, evaluates performance, and adapts over time to find the best configurations with fewer trials.
After finding optimal hyperparameter settings, the model undergoes training stage where the model weights are updated. As mentioned, the best model is selected based on the balanced test dataset and further tested on the biased test dataset.

2.3 Transfer Learning

Transfer Learning presents a computationally efficient approach to image classification by utilising pre-trained CNNs which are trained on large-scale datasets such as ImageNet, which consists of over 14 million images (Huh et al., 2016). This vast pre-training allows effective feature extraction which can be adapted to specific tasks.

Pre-trained CNNs are fine tuned by unfreezing trainable layers, allowing their weights to be updated during training to adapt to certain tasks (Kim et al., 2022). The number of unfrozen layers were manually adjusted based on model performance and additional trainable layers were appended to enhance the model's ability to adapt to EH image classification and convert extracted features into predictions. The learning rate, which controls the magnitude of weight updates to improve training efficiency, was also adjusted manually to find the optimum combination.

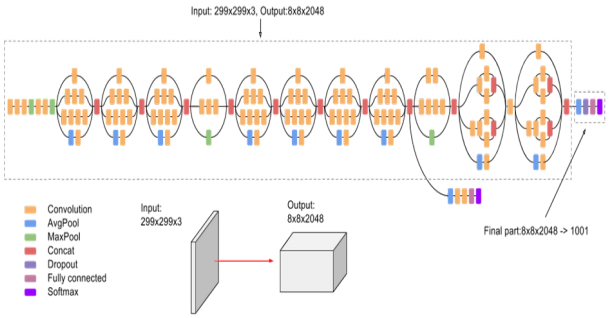


Figure 2: InceptionV3 Model Architecture (Szegedy et al., 2016)

Pre-trained CNNs, including InceptionV3, VGG16 and more, were used to train and evaluate classification performance. InceptionV3 was selected for its multi-scale feature extraction and efficient factorised convolutions as seen in Figure 2 which consists of 313 layers in total, while VGG16's sequential architecture with uniform 3×3 convolutions facilitated effective transfer learning, making it well-suited for small datasets (Simonyan and Zisserman, 2015). The best model was saved and further tested using the same procedure as the CNN model. After single-fold tests, four best-performed pre-trained CNNs were further evaluated by training with synthetic images.

2.4 k-Fold Cross Validation

k-fold Cross Validation, which is visualised in Figure 3, evaluates with k different test sets and which is implemented to further increase confidence to our evaluation results and ensure minimal overfitting has occurred. However, it increases the computational load by k times compared to the single-fold initial screening evaluation. Therefore, an initial screening using single-fold testing was conducted and two best=performing models were further evaluated with 5-fold cross validation to strengthen the reliability of the accuracy and reduce the risk of overfitting during training.

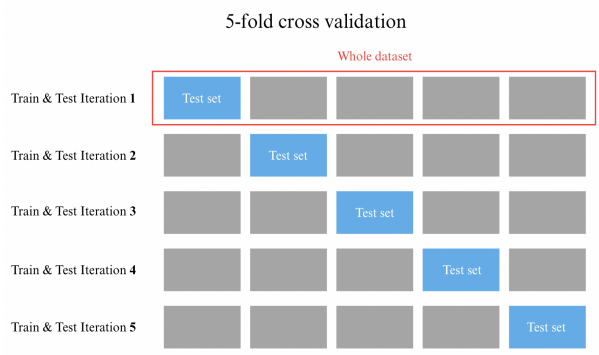


Figure 3: Illustration of 5-Fold Cross-Validation Process

2.5 Evaluation Metrics

To comprehensively evaluate the performance of our model, we rely on a variety of metrics derived from the confusion matrix — a summary of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) (Abdelmaksoud et al., 2023). Each metric offers a distinct perspective on the model's diagnostic performance.

Accuracy measures the proportion of total correct predictions. It provides a broad overview of performance, but can be misleading when applied to imbalanced datasets where one class dominates.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

Sensitivity (also known as **Recall**) quantifies the ability of the model to correctly identify positive cases. It is particularly important in reducing false negatives, which is critical in medical applications including EH detection. Sensitivity is considered to be the most important metric given the deadliness of EH.

$$\text{Sensitivity} = \frac{TP}{TP + FN} \tag{5}$$

Precision indicates the proportion of positive predictions that are truly positive. High precision reflects the model's ability to avoid false positives, ensuring its predictions are trustworthy.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{6}$$

F1-score is the harmonic mean of precision and sensitivity. It balances both false positives and false negatives, making it a reliable single metric when both types of error are costly.

$$F1\text{-score} = \frac{2 \times \text{Precision} \times \text{Sensitivity}}{\text{Precision} + \text{Sensitivity}} \tag{7}$$

By combining these evaluation metrics, we gain a well-rounded and nuanced understanding of the model's performance, particularly in identifying and managing EH cases with clinical reliability.

Together, these metrics allow for a nuanced and clinically relevant evaluation of model performance, going beyond a single number and helping identify potential weaknesses or biases in prediction.

2.6 Ethics

The SWET-labelled non-EH eczema images used in this study were shared with Imperial College London under a formal data-sharing agreement. The secondary use of these datasets has been approved by the Imperial Science Engineering Technology Research Ethics Committee (SETREC) under approval number 22IC7801. The EH images were taken from publicly available sources online and therefore did not require any ethical approval.

To comply with UK Data Protection Act (2018) regulations, all collected and processed data is fully anonymised and does not contain personally identifiable information. Each image is assigned a unique identifier accessible only to the clinician, ensuring patient confidentiality. Since the images do not contain personal data, they are not classified as "personal data" under UK law.

All stored data, including images used within the app, is securely maintained on the Imperial Research Data Store (RDS), which serves as the master storage system within the college's infrastructure, ensuring controlled access and compliance with institutional data governance policies. As for the EczemaNet website, the user's data is stored within the app and forwarded to their clinician.

3 Results

3.1 GAN Training

Combinations with the lowest FID scores are listed in Table 1. The GAN model trained with hyperparameters of target 0.6, gamma 0.1 and learning rate of 0.002 provided the best FID score of 81. This suggests that minimal regularisation, moderate augmentation, and a balanced learning rate provided an optimal generative quality and balance for the dataset, which aligns with the findings outlined in (Karras et al., 2020).

Gamma	Learning Rate	Target	FID
0.1	0.002	0.6	81
0.1	0.001	0.5	85
0.5	0.002	0.7	89
0.1	0.002	0.7	90
0.1	0.003	0.6	91
0.5	0.001	0.7	93
0.1	0.003	0.4	95
0.1	0.001	0.4	98
0.5	0.001	0.4	103

Table 1: Hyperparameter configurations and corresponding FID scores. Lower FID indicates better generative quality.

Synthetic images generated using the GAN model with the lowest FID score are depicted by Figure 4.

Figure 4: Synthetic images generated using GAN (left) and real images (right) for comparison

3.1.1 Image Selection

As we were only able to evaluate an entire set of images with the FID score, we also decided to individually score each image generated in order to select and use those that were of higher quality, bearing more resemblance with the real EH image dataset. Two methods were utilised to do this: similarity scores and GAN discriminator scores.

Out of 1000 generated images, 192 images met the threshold set for similarity scores and 297 in the case of the discriminator scores. The two sets of images were joined, resulting in 425 images. It was observed that the images with low scores (and hence were left unselected) were generally largely distorted, so much so that they did not look like any sort of human body part. This selection process thus prevented the models from learning unwanted patterns in these images.

3.2 Model Performance

3.2.1 Custom CNN model

During the hyperparameter tuning stage, the custom CNN model achieved its best validation performance of 97.6% (note k-fold was not implemented at this stage to decrease training time) composed of 6 convolutional layers and 1 fully connected layer with a learning rate of 1.15E-04 and RMSprop optimiser. By further training using this hyperparameter setting, the highest test accuracy of $97.7 \pm 0.6\%$ (see Figure 6a and Table 2) was achieved on the balanced test dataset, using training data consisting of the synthetic to real EH images ratio of 1:2. Furthermore, it is clear that increased use of synthetic EH images increased the custom CNN model's performance across all evaluation metrics. It also achieved a stable performance (97.3% accuracy) seen from Figure 7b on the biased test dataset.

3.2.2 Transfer Learning Based Model

The InceptionV3 model provided the highest test accuracy of $98.1 \pm 1\%$ (see Figure 5) on the balanced test dataset when trained with 34 unfrozen trainable layers and learning rate of 0.001. Four layers (1 pooling, 3 fully connected layers as seen in Table 3) were appended to allow the model to adapt to EH images and convert the extracted features into predictions. Validation accuracy calculated using 5-Fold cross validation gave $96.8 \pm 1.2\%$. It also performed well on the biased test dataset (see Figure 7b) with 97.9% accuracy. Four best-performing models based on balanced test dataset accuracies were further evaluated by training with synthetic images, with synthetic to real data ratios of 1:2 and 1:5. Notably, sensitivity decreased across all models when synthetic data was used in training as evident in Table 2.

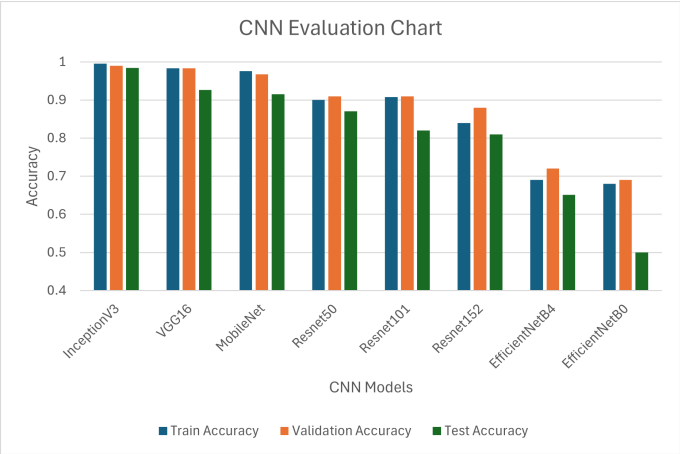


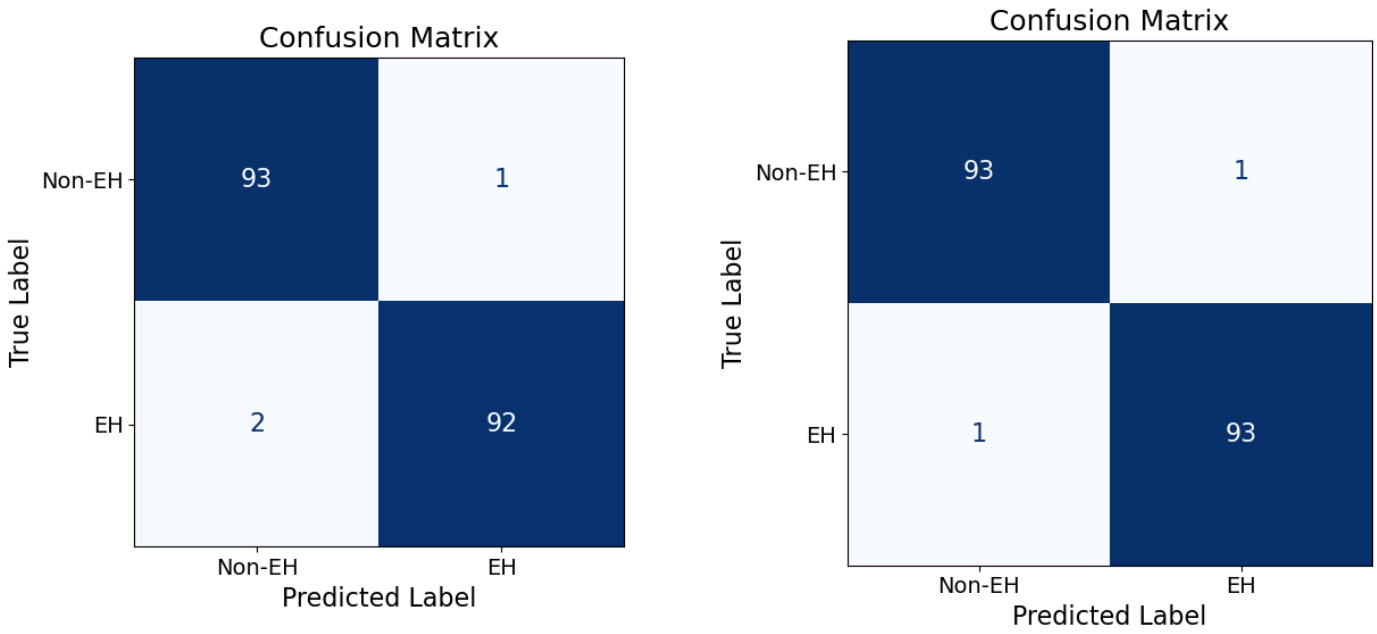
Figure 5: Transfer learning performance for different configurations of the CNN model.

3.2.3 Model Evaluation Metrics Comparison

The custom CNN model achieved the highest precision with $99.1 \pm 0.9\%$ while InceptionV3-based transfer learning models achieved higher performance across all other metrics including sensitivity, at $98.1 \pm 1\%$.

Table 2: Classification report for EH detection from different models

Model	**Synthetic : Real Training Images**	**Precision (%)**	**Sensitivity (%)**	**F1-score (%)**	**Accuracy (%)**	**Total Parameters (in millions)**
Custom CNN	No Synthetic images	94.8 ± 3	92.3 ± 0.9	93.6 ± 1	93.6 ± 2	
	1:5	98.5 ± 1	94.7 ± 1	96.5 ± 0.3	96.6 ± 0.3	~ 1.0
	1:2	$\mathbf{99.1} \pm 0.9$	96.2 ± 1	97.6 ± 0.7	97.7 ± 0.6	
Inception V3	No Synthetic images	98.1 ± 1	$\mathbf{98.1} \pm 0.9$	$\mathbf{98.1} \pm 0.5$	$\mathbf{98.1} \pm 1$	
	1:5	96.8 ± 0.8	90.2 ± 2	93.4 ± 1	93.6 ± 1	~ 23.0
	1:2	94.4 ± 1	92.1 ± 2	93.2 ± 1	93.4 ± 1	
VGG16	No Synthetic images	94.5	90.5	92.5	92.6	
	1:5	94.9	89.5	92.1	92.1	~ 138.0
	1:2	92.5	90.5	91.5	91.5	
MobileNet	No Synthetic images	92.5	90.5	91.5	91.5	
	1:5	92.5	90.5	91.5	91.5	~ 4.2
	1:2	80.6	83.2	81.9	81.3	
ResNet50	No Synthetic images	88.4	85.7	87.0	87.0	
	1:5	73.6	70.5	72.0	72.5	~ 25.6
	1:2	71.1	72.6	71.8	71.7	

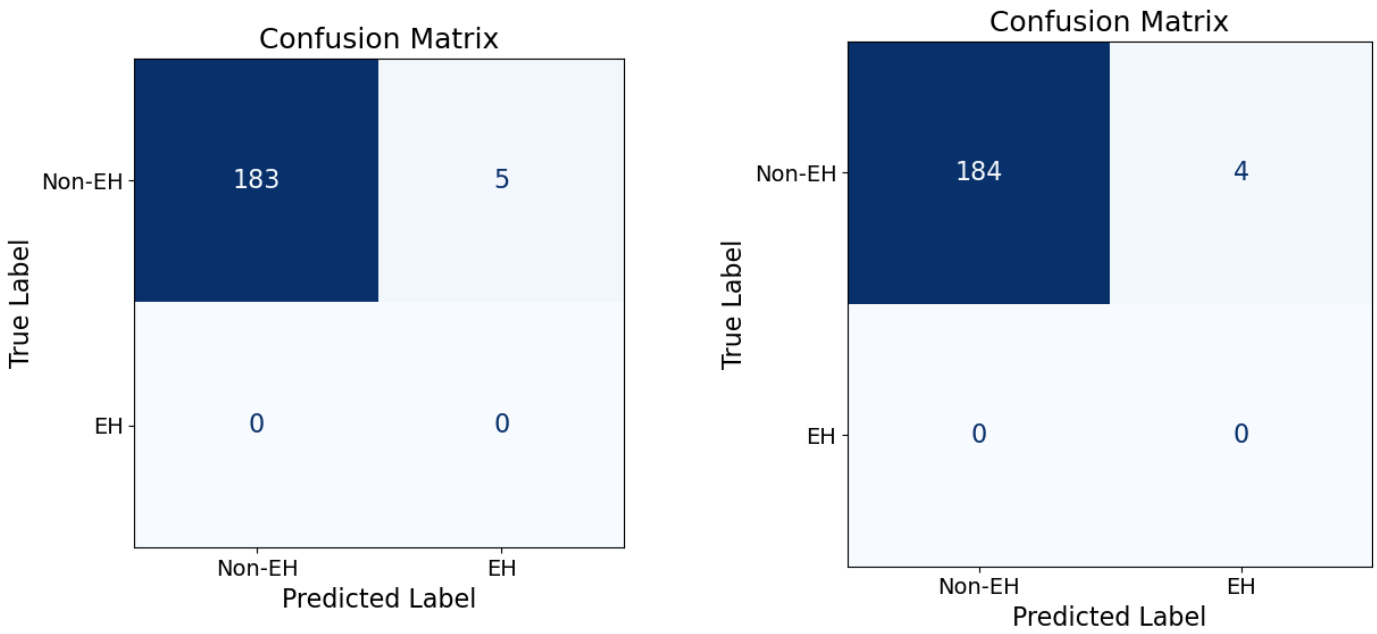


(a) Custom CNN (1:2 Synthetic to Real images ratio)

(b) InceptionV3 (without synthetic images)

Figure 6: Confusion matrix representing the models' performances on the test dataset using the best CNN and InceptionV3 models

---

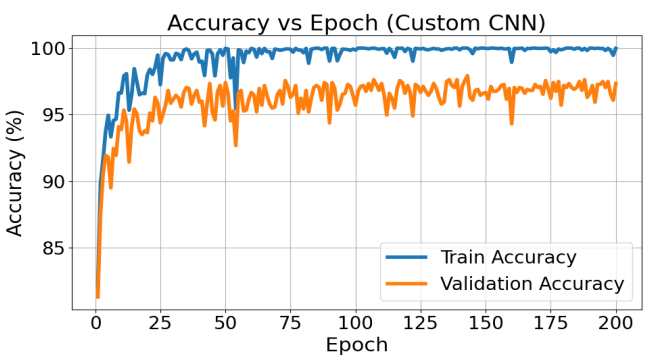


(a) Custom CNN (1:2 Synthetic to Real images ratio)

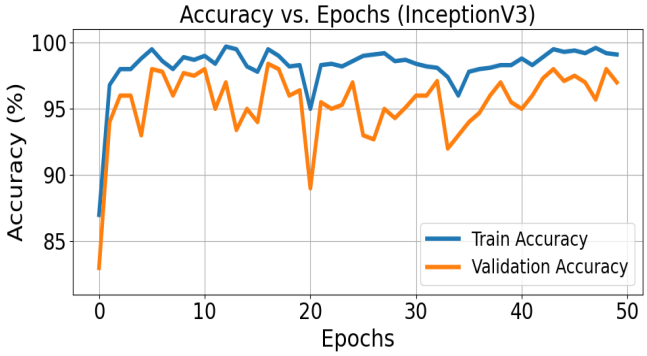
(b) InceptionV3 (without synthetic images)

Figure 7: Confusion matrix representing models' performances on a biased test dataset consisted of 188 non-EH images from SWET dataset using best performing CNN and InceptionV3 models

---

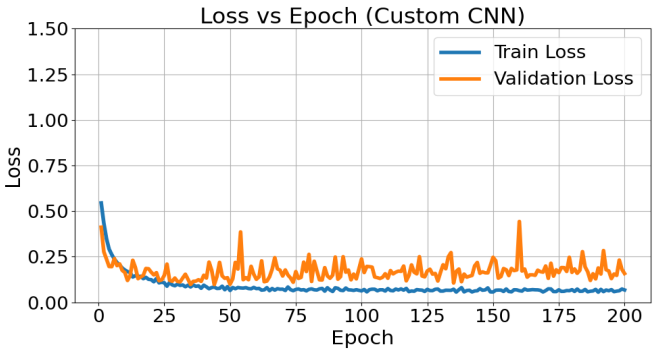


(a) Custom CNN (1:2 Synthetic to Real images ratio)

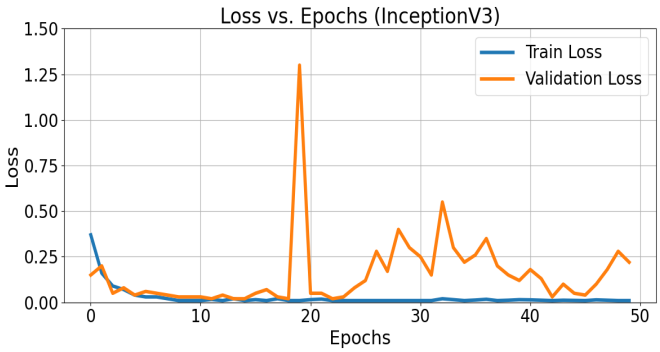


(b) InceptionV3 (without synthetic images)

Figure 8: Accuracy curves for best performing CNN and InceptionV3 models, showing training and validation accuracy with increasing training epochs



(a) Custom CNN (1:2 Synthetic to Real images ratio)



(b) InceptionV3 (without synthetic images)

Figure 9: Loss curves for best performing CNN and InceptionV3 models, showing training and validation losses across training epochs

Table 3: Summary of best-performing InceptionV3-based Transfer Learning architecture

Layer	**Feature Map**	**Size**	**Kernel**	**Stride**	**Activation**	**Params**
InceptionV3 Base (See Figure 2)	–	299 × 299	–	–	–	∼21.8M (last 34 trainable layers unfrozen)
GlobalAveragePooling2D	–	1 × 1 × 2048	–	–	–	0
Fully Connected Layer 1	–	512	–	–	ReLU	1,049,088
Fully Connected Layer 2	–	256	–	–	ReLU	131,328
Fully Connected Layer 3	–	2	–	–	Softmax	514
					Total Parameters:	**∼23M**
					Trainable Parameters:	**∼1.18M**

Table 4: Summary of best-performing Custom CNN architecture

Layer	Feature Map	Size	Kernel	Stride	Activation	Params
Convolution Layer 1	32	256 × 256	3 × 3	1	ReLU	896
Batch Normalization Layer 1	32	256 × 256	–	–	–	64
ReLU Activation	32	256 × 256	–	–	ReLU	0
Convolution Layer 2	64	256 × 256	3 × 3	1	ReLU	18,496
Batch Normalization Layer 2	64	256 × 256	–	–	–	128
ReLU Activation	64	256 × 256	–	–	ReLU	0
Max Pooling Layer 1	64	128 × 128	2 × 2	2	–	0
Convolution Layer 3	96	128 × 128	3 × 3	1	ReLU	55,392
Batch Normalization Layer 3	96	128 × 128	–	–	–	192
ReLU Activation	96	128 × 128	–	–	ReLU	0
Max Pooling Layer 2	96	64 × 64	2 × 2	2	–	0
Convolution Layer 4	128	64 × 64	3 × 3	1	ReLU	110,720
Batch Normalization Layer 4	128	64 × 64	–	–	–	256
ReLU Activation	128	64 × 64	–	–	ReLU	0
Max Pooling Layer 3	128	32 × 32	2 × 2	2	–	0
Convolution Layer 5	160	32 × 32	3 × 3	1	ReLU	184,480
Batch Normalization Layer 5	160	32 × 32	–	–	–	320
ReLU Activation	160	32 × 32	–	–	ReLU	0
Max Pooling Layer 4	160	16 × 16	2 × 2	2	–	0
Convolution Layer 6	192	16 × 16	3 × 3	1	ReLU	276,672
Batch Normalization Layer 6	192	16 × 16	–	–	–	384
ReLU Activation	192	16 × 16	–	–	ReLU	0
Max Pooling Layer 5	192	8 × 8	2 × 2	2	–	0
Fully Connected Layer 1	–	32	–	–	ReLU	393,248
Dropout (p=0.5)	–	32	–	–	–	0
Fully Connected Layer 2	–	2	–	–	Softmax	66
					Total Parameters:	**1,041,324**

Table 5: Specifications of the High Performance Computing (HPC) environment.

Component	Specification
HPC Cluster	CX3 HPC Cluster (Imperial)(RCS, 2025)
GPU	1 × RTX6000
CPU	24 CPUs
Memory	96 GB

Table 6: Training time comparison for Custom CNN and InceptionV3 models (in seconds)

Model	Tuning (s)	Training (s)	Total (s)	Avg Inference (s)
Custom CNN	21,780	18,354	39780	0.04
InceptionV3	_	2,134	2,134	0.07

4 Discussion

We proposed two main approaches for EH screening using clinical images: a custom CNN model and transfer learning using various pre-trained models. For model evaluation, standard classification metrics including accuracy, precision, sensitivity (recall), and F1-score were computed from confusion matrices along with the two best-performing models' classification reports, their accuracy curves and loss curves.

The specified aims are to develop a screening model that is accurate, sensitive, and generalisable for unseen EH images. The project also investigates how the use of synthetic images influences the model's performance. Additionally, it intends to build a computationally efficient EH screening model and evaluate different machine learning approaches.

Through carefully chosen evaluation metrics, we were able to build accurate and sensitive EH screening models with a highest accuracy of $98.1 \pm 1\%$ and sensitivity of $98.1 \pm 0.9\%$ (see Table 2) on a balanced test dataset, and 97.9% accuracy on a biased test dataset (see Figure 7b) using the InceptionV3-based transfer learning model. It was also observed that increased use of synthetic images improved the custom CNN model performance, while contrarily worsening the InceptionV3-based transfer learning performance (see Table 2).

As seen from Table 6, training the InceptionV3 model took 2134 seconds — about 1/20th of the time required for our custom CNN model, indicating its high training efficiency. In contrast, the inference time (testing time per image) was 0.07s for the InceptionV3 model and 0.04s for the custom CNN model. This indicates the trade-off for the InceptionV3 model in inference speed stemming from its deeper architecture with 23 million parameters (see Table 3) compared to 1 million for the custom CNN model.

4.1 Balanced Test Dataset Performance

4.1.1 Custom CNN Models

The custom CNN (see Table 2, when trained with a 1:2 synthetic to real training image ratio, achieved the best performance during testing stage. It maintained high performance on the balanced test dataset across measures: Precision: $99.1 \pm 0.9\%$, Sensitivity: $96.2 \pm 1\%$ and F1-score: $97.6 \pm 0.7\%$, and Accuracy: $97.6 \pm 0.6\%$ also visualized from its confusion matrix (see Figure 6a). This demonstrates the potential of using lightweight, task-specific architectures under a data-limited environment (474 EH images present).

4.1.2 Transfer Learning Models

The InceptionV3 model (see Table 2), when trained without any synthetic images, achieved the highest performance among all tested configurations. In the balanced test dataset, it maintained a Precision of $98.1 \pm 1\%$, Sensitivity of $98.1 \pm 0.9\%$, an F1-score of $98.1 \pm 0.5\%$, and an Accuracy of $98.1 \pm 1\%$, also seen from Figure 6b).

Despite the deep architecture of VGG16, which contains approximately 138 million parameters (see Table 2), its best classification accuracy reached only 92.6%. In comparison, the InceptionV3 model, with a significantly smaller parameter count of about 23 million, achieved a superior accuracy of 98.1%. Likewise, although ResNet50 has a similar parameter size (25.6 million) to InceptionV3, its performance was notably lower, with a best accuracy of 87%. Overall, the results across different pre-trained models underscore the importance of appropriate model selection and that deeper or more complex models do not guarantee better performance, especially in such data-constrained and domain-specific scenarios.

4.1.3 Effect of synthetic images

It is clear from Table 2 that increased usage of synthetic training images causes a consistent increase in model performance for custom CNN models while decreasing the model performance for transfer learning models including InceptionV3.

The results indicate that InceptionV3 struggled to effectively extract relevant EH image features from the synthetic images generated. This observation aligns with expectations given the high FID score of 81, exceeding the typical range of less than 30 required for the generation of high-quality synthetic images (Jayasumana et al., 2024). Despite the limitation, the custom CNN model successfully extracted meaningful EH image features from these relatively low-quality synthetic images, demonstrating its robustness and suitability for feature extraction tailored to the characteristics of the specific dataset employed. The weaker performance with synthetic images for InceptionV3 most likely stems from mismatched image features between pre-trained real images from ImageNet (Deng et al., 2009), and the training dataset that includes synthetic images.

The findings of custom CNN models with carefully synthesised images highlight its potential in more severely data-limited environments. Where insufficient data may hinder optimal training of transfer learning models, the custom CNN models may benefit from an increased dataset using synthetic images. However, for the case of EH classification, transfer learning demonstrated superior performance, suggesting that it is the most effective approach despite data limitations in this circumstance.

4.1.4 Prediction Accuracy and Loss

The accuracy and loss curves presented in Figure 8 and Figure 9 illustrate the learning ability and

adaptability to unseen data of the custom CNN and the InceptionV3-based transfer learning model. The custom CNN demonstrated stable convergence with steadily improving validation accuracy over the 300 epochs. In contrast, InceptionV3 achieved higher validation accuracy within 20 epochs while experiencing larger fluctuations, which may suggest mild overfitting. Overall, while both models reached reasonable convergence with mild fluctuations, InceptionV3 offered more accurate performance while the custom CNN offered higher stability. Nonetheless, these results indicate the importance of further validation with larger datasets to increase the reliability of models with unseen data.

4.2 Biased Test Dataset Performance

As illustrated in Figure 7a, the custom CNN correctly classified 183 images without EH (97.3% accuracy) while the InceptionV3-based transfer learning model (see Figure 7b) correctly classified 184 images (97.9% accuracy). This suggests that both models maintain robust performance on significant class imbalance with a low risk of over-fitting, while noting that the InceptionV3 model appears to be more capable of avoiding false positives.

Although directly evaluating sensitivity is challenging due to the absence of EH images in the biased dataset, the consistent performance observed with this biased dataset supports the reliability of the high sensitivity values obtained from the balanced test dataset ($98.1 \pm 0.9\%$ for InceptionV3). A better approach would be to have an additional balanced EH test dataset with SWET non-EH images, but further considerations would be needed due to the limited EH dataset as the additional split further decreases the number of training images.

4.3 Summary of Best-performing Model Architectures

The custom CNN configuration resulted in a total of 1.04 million parameters, where its detailed structure is described in Table 4. In contrast, InceptionV3-based transfer learning employed a fixed architecture shown in Table 3 with no additional hyperparameter tuning. The base InceptionV3 layers (except the last 34 trainable layers) were frozen where additional fully connected layers were fine-tuned. This model contained approximately 23 million parameters from which 1.18 million were trainable. This indicates a higher demand for memory and computational power for the InceptionV3 model, possibly indicating the custom CNN model is a better fit for further integration into memory-limited telemedicine applications. However, such an issue can be mitigated by implementing web-based applications (Reddy and Rajalakshmi, 2019) with an external server, while increasing the cost. Also, as prioritising model accuracy is critical in medical applications where erroneous predictions can have significant clinical consequences, InceptionV3 stands as the most convincing approach.

4.4 Computational Efficiency

The custom CNN model required approximately 11 hours (39,780 seconds) of total training time, while approximately 6 hours (21,780 seconds) were dedicated to the tuning process. In contrast, InceptionV3 required no tuning process and completed training in only 2,134 seconds. This indicates the exceptional training efficiency of the InceptionV3 compared to training a custom CNN architecture due to its pre-optimized structure.

The average inference time per image, t_{avg}, is estimated as:

$$t_{\text{avg}}(s/\text{image}) = \frac{T_{\text{total}}(s)}{N_{\text{test}}} \tag{8}$$

where t_{avg} is the average inference time per image, T_{total} is the total testing time, and N_{test} is the number of testing images, fixed at 188. The inference time was 0.07 s for the Inception model compared to 0.04 s for the custom CNN model. This suggests that although InceptionV3 is significantly faster to train due to its optimized architecture, it experiences a modest trade-off in inference speed. The slightly longer testing time per image may be due to the deeper and more complex structure of InceptionV3 (with $\sim$23 million parameters), which enables faster convergence during training but requires additional computation for each prediction. These results highlight the importance of considering the trade-off between model accuracy, computational power requirements, and latency (inference time) when integrating the model into a telemedicine mobile application.

4.5 Impact and Future Applications

As stated at the outset, EH is a deadly infection that quickly progresses, giving rise to the need for quick and reliable detection of the condition.

Currently, there are many groups which have endeavoured to develop machine learning models for detection of various skin diseases, where Shanthi et al. reported having achieved an accuracy of 93.3% in classifying EH (Shanthi et al., 2020). Our work presents a significant step forward, with a $98.1 \pm 1\%$ rate of correct EH predictions.

Although there is distrust in integrating machine learning tools into clinical workflows (**?**), using explainability methods in collaboration with clinicians may help address these concerns. This may lead to faster and more accurate diagnoses, as well as stan-

dardisation and streamlining of the triaging process by reducing the dependence on clinical suspicion.

We aim to further integrate the proposed EH screening model to the existing EczemaNet (Attar et al., 2023) to provide a telemedicine-based mobile app for eczema area detection and severity testing, as well as reliable EH screening. Specific future works to achieve these involve considering and implementing:

4.5.1 EH Variability Across Different Skin Tones

EH presents differently across different skin tones; for example, the red, blister-like spots that are characteristic of EH become harder to spot for darker skin colours. (pra, 2023). This variability in appearance may affect model performance. To mitigate this, further examination of the diversity of the datasets will be performed as well as increasing the diversity of the dataset, particularly where the current dataset lacks Asian skin types.

4.5.2 Explainability and Clinical Interpretability

To improve transparency and facilitate future integration into telemedicine platforms or clinical decision support systems, we plan to implement explainability features. These will include visual tools such as Grad-CAM (Selvaraju et al., 2017), which leverage gradients flowing into the final convolutional layer to generate heatmaps that highlight key image regions that influence model predictions (EH or non-EH). This will provide clinicians with insight into the decision-making process.

4.5.3 Image Segmentation using SAM

To ensure that the model focuses on relevant features, implementing the Segment Anything Model (SAM) in the image preprocessing pipeline can enhance this effect. SAM generates high-quality masks that distinguish patient body parts from irrelevant background information (Kirillov et al., 2023). This segmentation step enhances the signal-to-noise ratio of the input data by isolating informative features, ultimately aiding the model in learning more generalisable patterns associated with EH.

4.5.4 Patient and Clinican User Feedback

An essential future direction involves conducting structured usability testing and feedback collection from both clinicians and patients. While our model shows promising technical performance, its real-world utility depends on how well it integrates into clinical workflows and how comfortable users feel interacting with it. For clinicians, semi-structured interviews and simulated diagnostic scenarios could help assess the interpretability of the model output, and evaluate whether the tool aids or hinders decision-making. For patients, feedback could be collected through user experience surveys and focus groups, exploring trust in AI-driven diagnostics, interface clarity, and willingness to act on model recommendations. This dual feedback loop would be critical in refining both the technical and user-facing aspects of the system, ensuring it is clinically viable, ethically responsible, and aligned with user needs.

5 Conclusion

In this study, various image classification techniques were explored to develop an accurate EH screening model. Among the approaches used, the InceptionV3-based transfer learning model achieved the highest performance with test accuracy and sensitivity of 98.1%, indicating a strong potential for clinical use. Furthermore, custom CNN achieved test accuracy of 97.7% and sensitivity of 96.2%.

These results hold strong implications for future research and applications for image classification of rare skin diseases beyond EH, particularly where available datasets are limited. While both models displayed strong potential, they show distinct advantages depending on the availability of training data. InceptionV3 may be a suitable option due to its higher accuracy when trained with real datasets, while custom CNN could be a viable alternative when augmented with synthetic data where limited datasets can be expanded with GAN-generated synthetic images.

Although additional validation techniques were implemented to enhance the reliability of our results, further evaluation using a broader and more diverse dataset of real EH images would be essential to strengthen the clinical reliability of the model. Nonetheless, our current results suggest that the proposed models have potential to be utilised as a diagnostic aid, supporting clinicians with EH identification.

6 Appendix

6.1 Code Repository

The repository containing the complete source codes for training the custom CNN models, along with the best-performing InceptionV3-based transfer learning model, can be found at `https://github.ic.ac.uk/tanaka-group/Eczema_Herpeticum_Y3_Project.git`.
If you encounter access issues due to repository restrictions, please note that access requests can be directed to r.tanaka@imperial.ac.uk.

6.2 Project Timeline and Management

6.2.1 Project Timeline

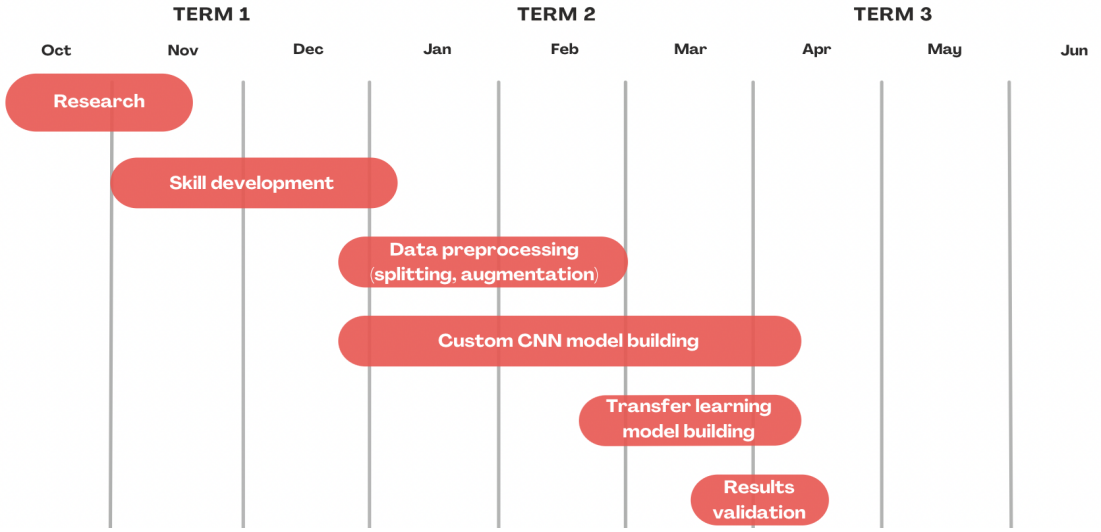


Figure 10: Gantt chart illustrating the different stages of our project over the academic year

6.2.2 Project Management

While we initially planned to use the Segment Anything Model (SAM) for image segmentation, we realised midway through the project that we didn't have enough time to properly integrate and test it. Instead, we focused on simpler image processing methods that were more manageable given the constraints. Similarly, we had hoped to include explainability features—like saliency maps or model interpretation tools—but this was one of the first things we had to drop when time started running short.

Time management ended up being more difficult than expected. Some tasks, particularly running certain scripts and debugging, took longer than planned. We also faced a few logistical issues—one team member had to take a break due to illness, and there were times when we struggled to stay fully in sync as a group.

The initial objective was to build an app, integrating our EH screening model, with the existing EczemaNet model for eczema area and severity detection. However, due to time constraints, we adjusted our scope to focus on building an EH screening model with accurate, and reliable performance.

6.2.3 Lessons Learned

Balancing Around Personal Commitments: One of the biggest challenges we faced was juggling the project alongside everyone's existing responsibilities. Early on into the project we realised that effective communication was the best way to overcome this; by knowing in advance each other's commitments we were able to plan the distribution of work accordingly, allowing us to stay on track.

At the end of term 1, one of our team members became ill and had to take a break from studies, which naturally affected our momentum. We did not fully appreciate this at this time, and their work went unfinished for more than a month before we found out. This made us realise the importance of clear communications. If we had been better at this, we could've planned more flexibly and been better prepared for any setbacks. We learned that making time to understand and support each other as teammates is just as important to keep things running smoothly.

Importance of Contingency Planning: We learned the hard way how important it is to have backups when things don't go to plan. Some parts of our project, like running certain scripts, took way longer than

expected. We also had technical issues such as difficulties connecting to Imperial's VPN during the Christmas break, which slowed everything down. Looking back, we should have thought more carefully about what might go wrong and had some alternatives ready. We went in quite confident with our roadmap, but real projects are rarely that smooth. Next time, we'd definitely spend more time upfront thinking through "what ifs" and building flexibility into the plan.

Clearer Documentation: We weren't consistent in our documenting and sharing of progress. Sometimes two people would end up doing similar tasks, or we'd forget what had already been tried. It made finishing the report harder too, since not everyone had the full picture. We didn't have a proper system for keeping track of progress or experiments, which meant it was harder to go back and check. In reality, setting up a running document where we log what we've done would keep everyone in the loop. Good coordination saves time, avoids confusion, and makes it easier to work as a team while juggling other responsibilities.

References

(2023). Eczema herpeticum in a 20-year-old woman with a history of atopic dermatitis (ad) and herpes labialis. *Practical Dermatology*.

Abdelmaksoud, E. et al. (2023). Skin cancer classification using transfer learning and deep learning techniques with dermoscopic images using the inception v3 model. *Intelligent Medicine*, 3:100027.

Attar, R., Hurault, G., Wang, Z., Mokhtari, R., Pan, K., Olabi, B., Earp, E., Steele, L., Williams, H., and Tanaka, R. (2023). Reliable detection of eczema areas for fully automated assessment of eczema severity from digital camera images. *JID Innovations*, 3(5):100213.

Bridle, J. (1990). Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation of parameters. In *Advances in Neural Information Processing Systems*, pages 211–217.

Deng, J., Dong, W., Socher, R., Li, L., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255. IEEE.

Fan, J. (2024). Analysis of generative adversarial networks (gans) and their variants based on encoders and decoders. *Proceedings of the 1st International Conference on Engineering Management, Information Technology and Intelligence*.

Han, S., Park, I., Chang, S., Lim, W., Kim, M., Park, G., Chae, J., Huh, C., and Na, J.-I. (2020). Augmented intelligence dermatology: Deep neural networks empower medical professionals in diagnosing skin cancer and predicting treatment options for 134 skin disorders. *Journal of Investigative Dermatology*, 140(9):1753–1761.

Huh, M., Agrawal, P., and Efros, A. A. (2016). What makes imagenet good for transfer learning? `https://arxiv.org/pdf/1608.08614`. Available at: `https://arxiv.org/pdf/1608.08614`.

Hull University Teaching Hospitals NHS Trust (2022). Virology turnaround times. `https://www.hey.nhs.uk/pathology/departmentofinfection/virology/virology-turnaround-times/`. [online].

Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, pages 448–456.

Irfan Rasyid, Muhammad Resa Arif Yudianto, M. T. A. P. (2023). Electronic product recommendation system using the cosine similarity algorithm and vgg-16. *Sinkron*, 8(4).

Jayasumana, S. et al. (2024). Rethinking fid: Towards a better evaluation metric for image generation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, volume abs/1911.07023, pages 9307–9315.

Karras, T. et al. (2020). Training generative adversarial networks with limited data. `https://papers.nips.cc/paper/2020/file/8d30aa96e72440759f74bd2306c1fa3d-Paper.pdf`. [online].

Kim, H. E., Kim, Y., Cho, J. K., Jang, Y. L., Lee, M. W., and Ye, J. C. (2022). Transfer learning for medical image classification: a literature review. *BMC Medical Imaging*, 22(1).

Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., Dollár, P., and Girshick, R. (2023). Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4015–4026.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324. [online].

Leung, D. Y. M. (2013). Why is eczema herpeticum unexpectedly rare? *Antiviral Research*, 98(2):153–157. Epub 2013 Feb 19. NIHMS510869.

Lin, C.-Y. (2023). Eczema herpeticum — dermnet nz. `https://dermnetnz.org/topics/eczema-herpeticum`. [online].

M., H., H., R., Unterthiner T., N. B., and Hochreiter, S. (2017). Gans trained by a two time-scale update rule converge to a local nash equilibrium. https://arxiv.org/pdf/1706.08500. [online].

Nair, V. and Hinton, G. (2010). Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 807–814.

Optuna (2018). Optuna documentation. https://optuna.readthedocs.io/en/stable/index.html.

Patient.info (2023). Eczema herpeticum. https://patient.info/skin-conditions/atopic-eczema/eczema-herpeticum#how-is-eczema-herpeticum-diagnosed. [online].

Phung, V. and Rhee, E. (2018). A deep learning approach for classification of cloud image patches on small datasets. *Journal of Information and Communication Convergence Engineering*, 16:173–178. [online].

RCS, I. (2025). Hpc cluster specification. https://icl-rcs-user-guide.readthedocs.io/en/latest/hpc/cluster-specification/. [online].

Reddy, D. S. and Rajalakshmi, P. (2019). A novel web application framework for ubiquitous classification of fatty liver using ultrasound images. In *2019 IEEE 5th World Forum on Internet of Things (WF-IoT)*, pages 502–506. IEEE.

Robb, E., Chu, W.-S., Kumar, A., and Huang, J.-B. (2020). Few-shot adaptation of generative adversarial networks. https://arxiv.org/pdf/2010.11943. [online].

Robbins, H. and Monro, S. (1951). A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3):400–407.

Sadik, R., Majumder, A., Biswas, A., Ahammad, B., and Rahman, M. (2023). An in-depth analysis of convolutional neural network architectures with transfer learning for skin disease diagnosis. *Healthcare Analytics*, 3:100143.

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. (2017). Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 618–626.

Shanthi, T., Sabeenian, R., and Anand, R. (2020). Automatic diagnosis of skin diseases using convolution neural network. *Microprocessors and Microsystems*, 76:103074.

Simonyan, K. and Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Swinscow, T. V. D. (2021). *Statistics at Square One*. John Wiley Sons.

Szegedy, C. et al. (2016). Rethinking the inception architecture for computer vision. https://www.cv-foundation.org/openaccess/content_cvpr_2016/papers/Szegedy_Rethinking_the_Inception_CVPR_2016_paper.pdf. [online].

Thomas, K. S. et al. (2011). A randomised controlled trial of ion-exchange water softeners for the treatment of eczema in children. *PLoS Medicine*, 8(2):e1000395.

Tieleman, T. and Hinton, G. (2012). Lecture 6.5 - rmsprop: Divide the gradient by a running average of its recent magnitude. Coursera Lecture Notes for Neural Networks for Machine Learning.

Wang, J. and Perez, L. (2017). The effectiveness of data augmentation in image classification using deep learning. https://arxiv.org/pdf/1712.04621. [online].

Ward, A., Li, J., Wang, J., Lakshminarasimhan, S., Carrick, A., Campana, B., Hartford, J., Sreenivasaiah, P., Tiyasirisokchai, T., Virmani, S., Wong, R., Matias, Y., Corrado, G., Webster, D., Smith, M., Siegel, D., Lin, S., Ko, J., Karthikesalingam, A., Semturs, C., and Rao, P. (2024). Creating an empirical dermatology dataset through crowdsourcing with web search advertisements. *JAMA Network Open*, 7(11):e2446615.

Xiao, A. and Tsuchiya, A. (2021). Eczema herpeticum. https://www.ncbi.nlm.nih.gov/books/NBK560781/. [online].