# Deep Learning to Classify Images of Eczema, Fungal and Viral Infections

Christine Liu, Fabrice Ingabire, Stephen Zhang, Michael Enaye

## 1. Introduction

Skin diseases are among the most common health problems in humans. There are more than 3000 identified varieties of skin diseases that cause many different symptoms ranging from itching to psychosocial effects or death. Many of these skin diseases can look similar to one another in terms of their appearance. Due to their similarity, these skin conditions are misdiagnosed often, creating even greater issues for the recipient of the disease in the future.

In 2019, Brinker *et al.* reported their findings in training a convolutional neural network (CNN) on dermoscopic images and then tested them on clinical close-up images. The CNN was trained on dermoscopic images, images specifically on melanomas and atypical nevi. These images were taken from the International Skin Imaging Collaboration (ISIC) and the HAM10000 dataset. From their findings, they were able to output a 0 or a 1 from each image processed, indicating if the image was classified correctly for the skin disease. They concluded the CNN performed just as well as a dermatologist for classifying clinical images.

In 2021, Saba reported the findings of different research groups on using computer vision to detect skin cancer. They mentioned the difficulty of detecting skin cancer or just in general, detecting the presence of a skin disease due to the different color and clinical features in the dermoscopic images. Researchers used both handcrafted and non-handcrafted features in their diagnosis of dermoscopic image sets. These data sets include: the ISIC data set, Asan data set, ISBI 2016 data set, ISBI 2017 data set, and UMCG. They concluded that trained systems can detect cancer from a large amount of medical images without human intervention but there is still room for further improvement.

In this work, we trained a CNN on three types of skin diseases: eczema, fungal infections, and viral infections. It is estimated that 30% of the U.S. population is affected by eczema. Fungal infections and viral infections are also very common. In order to help treat these diseases effectively, we try to incorporate techniques used in the past work of this field to reduce the prevalent misdiagnosis of skin diseases.

## 2. Methods

### 2.1 Data

For the data used to train and test our model, we had approximately 5500 images. All of the images included one of the three following three skin diseases: eczema, fungal infections, and viral infections. As shown in Figure 1, the images in the data set are not limited to one part of the human body. The figures labeled with “0”, “1”, and “2” are images of eczema, fungal infection, and viral infection respectively. In addition, some of the images in the dataset are taken from the DermNet dataset as shown in Figure 1 with some of the images containing a watermark in the center.

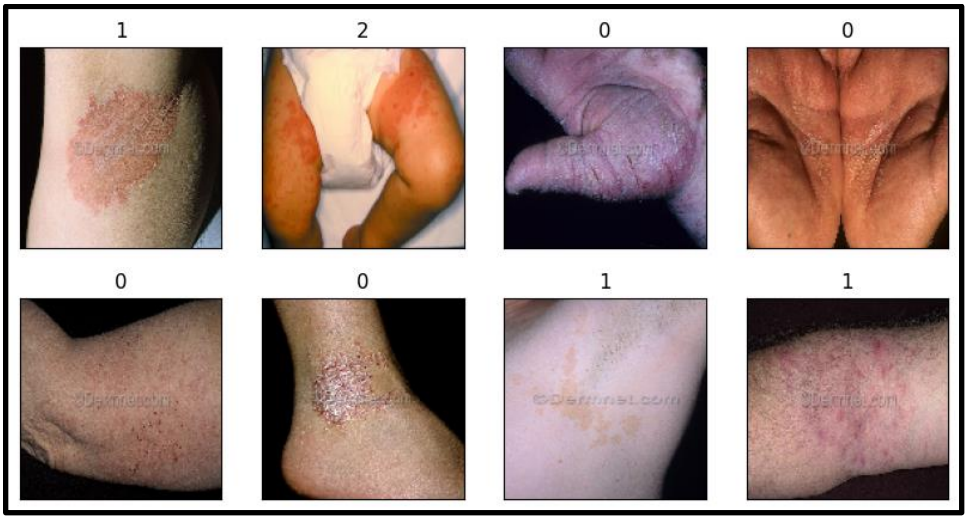

Fig 1. Some images in dataset with associated labels

### 2.2 Algorithm

To perform the classification, we used a CNN model called Xception, that we imported from keras. The architecture of this model comprises two main parts, namely depthwise separable convolutions and pointwise convolutions. Depthwise separable convolutions apply a convolutional kernel separately on each input channel which produces the output size with the same channels and input channels. Pointwise convolutions apply a 1x1xN kernel to the input, where N is the desired number of output channels. In the Xception model, classical convolutions were replaced with the combination of pointwise convolutions and depthwise convolutions with ReLU activation for both of them. In addition, Xception uses residual transference after every two separable convolutions, to bring forward past knowledge that could have been lost.

We trained the Xception model on our training data using Categorical Cross Entropy Rate. We chose Xception because it is as powerful as models like ResNet, but it uses far less parameters which reduces the training time.

### 2.3 Statistics

Our model uses categorical cross entropy loss. This loss function implements -log(softmax) loss for each class. For each class the loss functions Cross-Entropy Loss = $-log(\frac{e^{s_p}}{\sum_j e^{s_j}})$ where $s_p$ is our model's score of the correct class and $s_j$ is our model's score of the $j^{th}$ class. The softmax outputs the probability of the class p compared to other classes, and the probability is between 0 and 1. Therefore, applying log to the softmax function gives a negative value whose absolute value increases with the decrease in probability. Therefore, the cross-entropy loss increases when the probability of predicting the correct class is small. Our model uses gradient descent to minimize the loss and maximize the prediction accuracy of our model.

## 3. Results

The figure below shows the results of the training and validation loss and accuracy as a function of iterations for 100 epochs. As we would expect, the training loss continuously goes down towards zero and the training accuracy continuously increases. Validation loss decreases fairly continuously until about 0.6. The loss then appears to bounce up and down, but eventually seems to increase again. Similarly, the accuracy increases until it reaches about 80%, then goes slightly up and down, but instead may still be increasing at a slower rate.

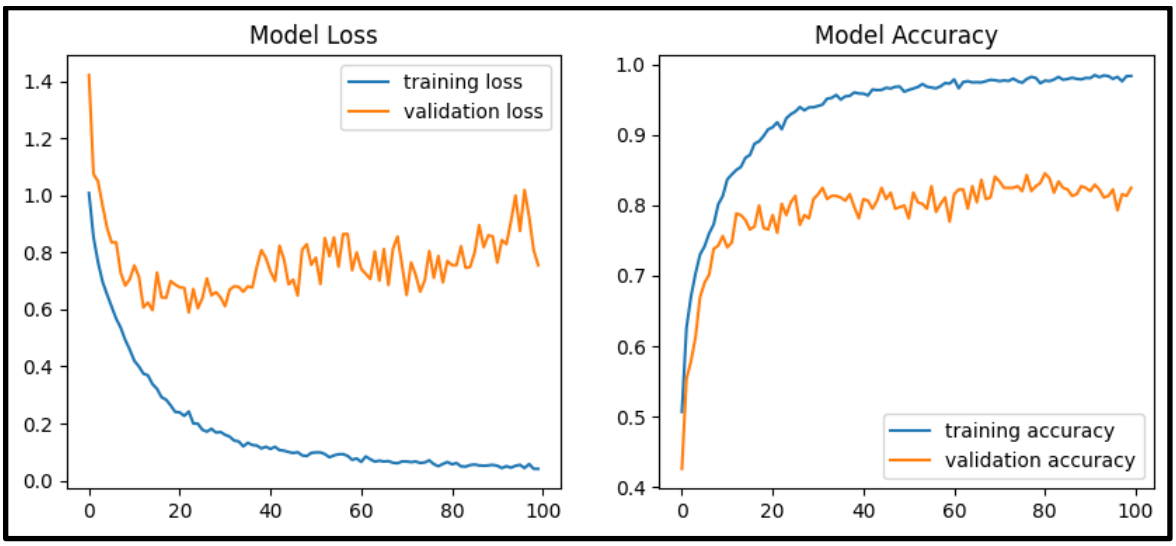

Fig 2. Loss and accuracy plots of training results for 100 epochs

Ultimately we decided on 25 epochs as not much improvement happens afterwards.

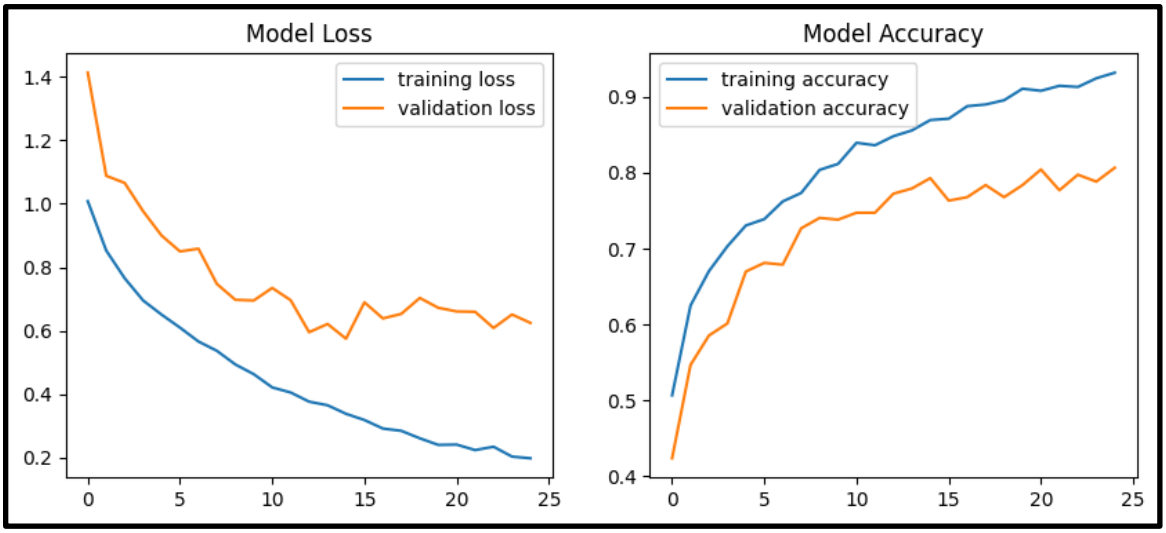

Fig 3. Loss and accuracy plots of training results for 25 epochs

When applying the model to our test data we achieved the following results: loss = 0.69, accuracy = 0.81. This is roughly what we observe for the validation loss and accuracy, meaning there isn't any significant underfitting or overfitting.

The figure below shows the confusion matrix of our model applied to our test data. The confusion matrix counts the occurrences of each actual label and predicted label combination. For our code: 0 = Eczema, 1 = Fungal Infection, 2 = Viral Infection.
It is clear that most images are classified correctly, since the diagonals of the matrix have the highest counts. There are some areas with higher misclassification such as

eczema being predicted as viral infections. However in general, our results show that there is enough difference in features of these different skin conditions that allow us to classify them with reasonably high accuracy.

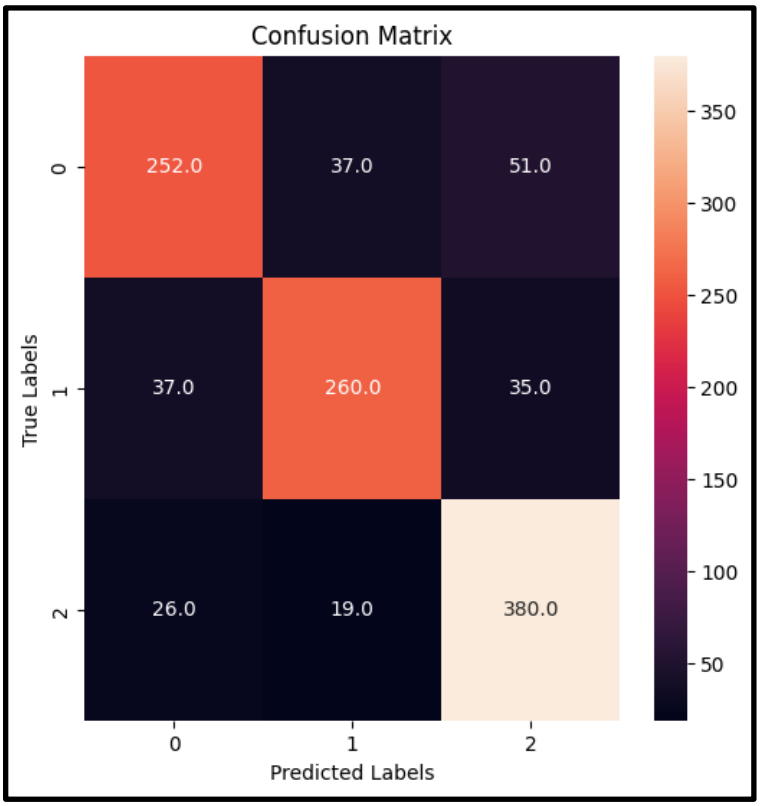


Fig 4. Confusion matrix on test data

Below are some other evaluation metrics related to the model on the test data.

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.80 | 0.74 | 0.77 | 340 |
| 1 | 0.82 | 0.78 | 0.80 | 332 |
| 2 | 0.82 | 0.89 | 0.85 | 425 |
| accuracy | | | 0.81 | 1097 |
| macro avg | 0.81 | 0.81 | 0.81 | 1097 |
| weighted avg | 0.81 | 0.81 | 0.81 | 1097 |

Fig 5. Other evaluation metrics on test data

Finally, we ran the model on some images taken from Google to see our model in action. To provide an example result, the model predicts the image below as [0.39, 0.21, 0.40]. This means that it has a 39% chance of being a picture of eczema, 21%

chance of being a picture of a fungal infection, and a 40% chance of being a picture of a viral infection. Although the probabilities are close, the image is predicted as a viral infection, which is correct. However, there are definitely still images that the model evaluates incorrectly.

Fig 6. Image of wart (viral infection) found on Google

## 4. Discussion

In summary, our model's performance on the data it was provided was strong. For the training and validation processes, we concluded with a training loss as low as 0.46 and training accuracy as high as 81% after 10 epochs with a batch size of 128 images each epoch. Model validation experienced worse results, with validation loss at 0.64 and validation accuracy at 76%. After we moved onto testing the model with the test dataset, we witnessed an expected, yet minimal drop in accuracy of 75%. However, our adjustment to 25 epochs made a difference to our results for the better, with training loss dipping to nearly 20% and training accuracy finally breaking 90%. After using the test dataset, our accuracy rose to 81%, which clearly means that our model is improving at classification when we observed similar dips and increases in loss and accuracy for validation (below .6 and approaching 80% respectively). As pictured in Fig. 5, our F1 scores for each diagnostic category are at or above 0.77, which indicates that its level of accuracy is fairly high.

This is supported when comparing our work to other similar work - for example, the models referred to in Tanzila Saba's paper regarding the use of handcrafted and non-handcrafted features for skin cancer diagnosis). Of course, our data is not focused on detecting melanoma and instead looks to detect other skin conditions, yet the ability to detect such conditions is strongly related to our shared goal of diagnosis and treatment

from images of the skin. In one of the reports mentioned in the paper, which used convolutional neural networks to classify melanoma, Brinker et. al reported a mean sensitivity value of 68.2% after testing it on clinical images, which is lower than our overall mean sensitivity of 80.6% (derived from our confusion matrix) after testing on our own clinical images. What is important to note here is that the CNN used for melanoma detection performed on par with dermatologists *without* training directly on clinical images; with this in mind, we should expect that our sensitivity values would be higher, since we are training on the exact type of data that we are testing with, unlike Brinker et al.'s neural network. Our results successfully report higher accuracy values from our testing when compared to Brinker et al.'s CNN, which means that our CNN is keeping up with expected accuracy values (observing worse sensitivity values would be especially concerning). With enough improvements, we believe that our model could be applicable in clinical settings, especially if it is able to perform on par or better than dermatologists at correctly classifying conditions in our three categories.

This by no means means that our model is perfect. In fact, our model suffers from similar problems as the mentioned works. One of these problems is the bias in our dataset; the patients in the DermNet images used for training our model were almost entirely made up of light-skinned individuals. This is a problem if we wish to apply this model on a large scale, since it fails to capture darker skin - this could have the consequence of darker-skinned patients being misdiagnosed when input into our model. This could be rectified by increasing the size of the dataset to properly accommodate images of dark-skinned patients, perhaps to proportions equivalent to the expected population that our model is expected to be used on.

An additional complication (a place where our model could be improved) is that there are weights that could be applied to probability scores that we do not consider. For example, if a patient has a family history of eczema, the probability that they would also develop eczema is increased. Of course, this information was not readily available to us for the data we trained with, given that it is a public dataset; yet, if we wish to use our model for diagnosis, it could perform better at its job by taking these factors into account through weighting the relevant categories.

## 5. Conclusion

From reviewing the results of our model, we believe that our model's performance on the data was strong as it was able to correctly classify most of the images in our dataset. There is currently a reasonable accuracy with most of the images being

classified correctly to one of the three types of skin diseases, but there is definitely room for improvement. From looking at the results, we can see that the training loss is going down and the training accuracy is going up. With these current results, we get ever closer to achieving our goal of being able to help reduce the misdiagnosis of different skin diseases as we are able to identify which of three skin diseases an image contains, given that one of them is guaranteed to have one of the three conditions. As described in our discussion, there are a variety of issues found in our work, yet none of them are out of the range of possibility to fix. With the motivation to reduce all types of misdiagnosis related to skin disease, our future work can iterate on this existing model with the resolution of our current issues and potentially include more categories of classification such that our model is more robust.

**References:**


- Saba, T. "Computer vision for microscopic skin cancer diagnosis using handcrafted and non-handcrafted features." *Microsc Res Tech*. 2021; 84: 1272–1283. https://doi.org/10.1002/jemt.23686.
- Martins, Carla. "Multiclass Image Classification-Hands-on with Keras and TensorFlow." *Medium*, Towards AI, 23 Sept. 2022, https://pub.towardsai.net/multiclass-image-classification-hands-on-with-keras-and-tensoflow-e1cf434f3467.
- Brinker, Titus J., et al. "A convolutional neural network trained with dermoscopic images performed on par with 145 dermatologists in a clinical melanoma image classification task." *European Journal of Cancer* 111 (2019): 148-154. https://doi.org/10.1016/j.ejca.2019.02.005


**Contribution Table:**

| Name | Coding | Presentation | Write-Up |
|---|---|---|---|
| Christine | Contribution: 60% Refactored code from article to create data arrays and run on Xception model, added multiple ways to evaluate results | Contribution: 25% Introduction and Results | Contribution: 15% Added images and wrote parts in the result section starting from the loss and accuracy plots. |

| | | | |
|---|---|---|---|
| Michael | Contribution: 10% Assisted in the importing of data for the project and the setup of the Google Drive file(s) containing our dataset | Contribution: 25% Created and presented the Our Work & Other Work and Issues slides | Contribution: 35% Wrote discussion portion and proofread/corrected all other portions |
| Stephen | Contribution: 10% Tried to alter the model in changing it from a binary classification to be able to label what type of disease an image contains. | Contribution: 25% Created and presented the Data and Conclusion & Future Work slides. | Contribution: 25% Wrote introduction, data, and conclusion section. |
| Fabrice | Contribution: 20% Tested DL architectures while choosing the model to use. After testing different custom DL architectures, we decided to go with an already existing Xception architecture. | Contribution 25% Created and presented the Algorithm and Stats slides. | Contribution: 30% Wrote algorithms, Statistics, and made a few edits in different parts of the write-up. |