Skin Cancer Detection with: Convolutional Neural Network and Vision Transformer

William Gagné
University of Toronto

Boyuan Cui
University of Toronto

Zhonghan Chen
University of Toronto

Abstract

Skin cancer is a common and deadly disease, and early detection is crucial for successful treatment. In this study, we compare the performance of two deep learning models, the Convolutional Neural Network (CNN) and Transformer, on the task of skin lesion detection. Using diagnostic results from dermatologists in the HAM10000 dataset, we trained the Inception V3 model and Vision Transformer and tested them on the ISIC 2018 challenge test dataset. We found that class imbalance in the training data had a significant impact on model effectiveness, but by employing various methods such as oversampling and focal loss, we were able to address this issue. Our results show that the CNN achieved a two-way accuracy of 81.68% and an AUC score of 0.89, while the Transformer had an accuracy of 75% and an AUC score of 0.78. These findings suggest that the CNN is more robust than the Transformer when working with limited data for skin lesion detection. [1]

1 Introduction

According to World Health Organization (WHO) data, skin cancer has emerged as the most frequently diagnosed cancer globally, with approximately 325,000 new melanoma cases and 57,000 related deaths reported in 2020 (Skin Cancer – IARC, n.d.). Early detection is crucial, as the estimated 5-year survival rate for melanoma plummets from over 99% when identified in its earliest stages to a mere 14% when detected in its most advanced stages (Esteva & al., 2017). Therefore, enhancing the efficiency, affordability, and accessibility of skin lesion diagnostics is of paramount importance in saving lives through early intervention.

2 Dataset

We utilized the HAM10000 Dataset (Tschandl et al., 2018), prevalent in skin cancer research, for training and validation and the official ISIC 2018 test dataset (Codella et al., 2018) for testing. The training data comprises 10,015 dermatoscopic images with dermatologist-provided diagnoses, over half of which are histopathology-confirmed. The HAM10000 Dataset is imbalanced. Specifically, it contains 1.41% of vascular lesions (vasc), 1.14% of dermatofibroma (df), 3.2% of Actinic keratoses and intraepithelial carcinoma disease (akiec), 5.12% of basal cell carci-noma (bcc), 10.96% of benign keratosis-like lesions (bkl), 11.10% of melanoma, and 66.94% of melanocytic nevi (nv). We will demonstrate how we handle the imbalance in later sections.

3 Convolutional Neural Network (CNN)

Convolutional Neural Networks (CNNs) are versatile function approximators highly suited for grid-like data structures, including images (Wang & Ba, 2023). They consist of convolutional layers, pooling layers, and fully connected layers. Convolutional layers serve as feature extractors, utilizing

[1] `https://github.com/OscarC9912/Skin_Cancer_Detect_w_CNN_ViT.git`

filters or kernels to emphasize specific aspects of images, such as edges or curves. Weight sharing across filter applications significantly reduces model complexity. Next, pooling layers diminish spatial dimensions and boost generalization by applying specific rules to extract features. For instance, max-pooling layers obtain the highest value from a feature set. Both convolutional and pooling layers feature neurons connected to only a few neurons from the preceding layer, yielding translation-invariant features. This is crucial for image tasks, as images can vary significantly due to lighting and angles, necessitating robustness to variations. Lastly, fully connected layers execute the final task, which is classification in our case (Wang & Ba, 2023).

3.1 Related Work

Since finding a high volume of good-quality skin lesion data is difficult, many scientists chose to use a pre-trained model, often trained on ImageNet, to implement a skin lesion classifier CNN. One such paper that effectively uses transfer learning to achieve remarkable results in classifying skin lesions is Esteva & al. (2017). In this experiment, the researchers fine-tuned Google Inception V3 using 129 450 images. Their astonishing results from fully biopsy-proofed test data became the new benchmark for skin lesion classification. They achieved a carcinoma AUC of 0.94 and a melanoma AUC of 0.96. Their accuracies for classifying skin lesions into nine (nine-way accuracy) and three (three-way accuracy) classes were 55.4% and 71.2%, respectively. In the present work, we will test the performance of Esteva & al's CNN on a different training and test data set.

3.2 Method

First, we follow the guidelines provided Esteva & al. (2017) to implement the CNN. Thus, we utilize the Inception v3 model, pre-trained on the ImageNet dataset, as our base architecture. All layers of the Inception v3 base model are set to be trainable. We initialize the learning rate at 0.001 and compile the model using the RMSProp (decay: 0.9, momentum: 0.9, epsilon: 0.1). We use categorical cross-entropy. The model is trained until the validation loss ceases to decrease for four consecutive epochs (see Appendix Figure 3). To address the class imbalance, we limit the maximum number of samples per class to 1000. We also implement Esteva & al.'s custom data augmentation function that performs random rotations of an image between 0 and 359 degrees, crops the largest upright inscribed rectangle and flips the image vertically with a 0.5 probability.

After evaluating the classifier's accuracy, we opt for a second model that better addresses severe class imbalance. We implement an over-sampler and under-sampler to randomly sample 1000 training images for each minority and majority class (OpenAI, 2023). This process is repeated at each epoch, creating a balanced dataset for each epoch. To further account for class imbalance, we assign a weight to each class using sklearn and fit the model accordingly. We replace the custom data augmentation function with classical augmentation techniques such as rotation, zoom, shift, and flip, and add a dropout layer before the softmax layer. We substitute the categorical cross-entropy loss function with focal loss (gamma: 2.0, alpha: 0.25), which better handles class imbalance (OpenAI, 2023). Instead of RMS, we use the Adam optimizer (beta1: 0.9, beta2: 0.999, epsilon: 1e-8, decay: 1e-6). We reduce the learning rate to 0.000001 and train the model until convergence (refer to training and validation plots in Appendix Figure 4). See the algorithm box in the Appendix for more details on the training loop (Algorithm 1). Finally, we perform the same testing as we did on the original model.

4 Vision Transformer (ViT)

Transformers are a state-of-the-art neural network architecture in natural language processing (NLP), but their use in computer vision (CV) has been limited (Dosovitskiy et al., 2021). The Vision Transformer (ViT) is a neural network model that applies the transformer architecture to image recognition tasks. The ViT model consists of three components: a projection layer that converts an image into a sequence of embeddings, a transformer encoder, and a fully connected layer for classification.

To process an image, the ViT model first divides it into fixed-sized patches and maps each patch to a D-dimensional embedding using a trainable linear layer or a convolutional layer followed by a flattening operation. Positional encoding is then applied to each embedding before being fed to the transformer encoder. Similar to NLP classification tasks, the ViT model adds a special CLS token to the input sequence. After passing through the transformer encoder, the first token (i.e., the CLS token) is extracted and fed to the fully connected layer for classification. See Appendix Algorithm 2 for more details.

4.1 Related Work

While Vision Transformer (ViT) has been found to be a state-of-the-art model for natural language processing (NLP) tasks, its application in computer vision (CV) is relatively recent and still limited, particularly in skin lesion classification. Despite this, some studies in the literature have employed ViT for this task, showcasing promising results.

For instance, Aladhadh et al. (2022) proposed a custom Medical Vision Transformer architecture with an input image size of 72x72 and a patch size of 8x8, achieving an F1-score of 97% and an accuracy of 96.14% on the HAM10000 dataset. Another study by Xin et al. (2022) used ViT for skin lesion classification with a custom contrastive loss in addition to the traditional cross-entropy loss during training, achieving an accuracy and F-1 score of 94.1% on the same dataset.

Overall, these studies provide evidence of the potential of ViT in skin lesion classification, although further research is needed to explore its full capabilities in this domain.

4.2 Method

Our study uses the ViT-Base architecture with a patch size of 16x16 (vit_b_16) pre-trained on ImageNet data. The last classification layer of ViT is replaced with a linear layer mapping 768 to 7 in order to fit our classification of seven skin lesion classes. In order to use the pre-trained weights, we need to adjust our input image size to 224x224 to fit the ViT model. Due to the imbalance of classes in the dataset, we randomly oversampled the training data with weights such that each class has an equal probability of being selected into the batch. We also use focal loss with $\alpha = 0.25$ and $\gamma = 2$ to train the ViT model to further counter the class imbalance issue.

The ViT model is trained by Adam optimizer with a learning rate of 0.001, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and a weight decay of 0.1. We trained the model for 15 epochs with a batch size of 64 and only kept the model with the highest validation accuracy. All parameters of ViT are frozen during training except for the last classification linear layer.

5 Results

5.1 CNN Results

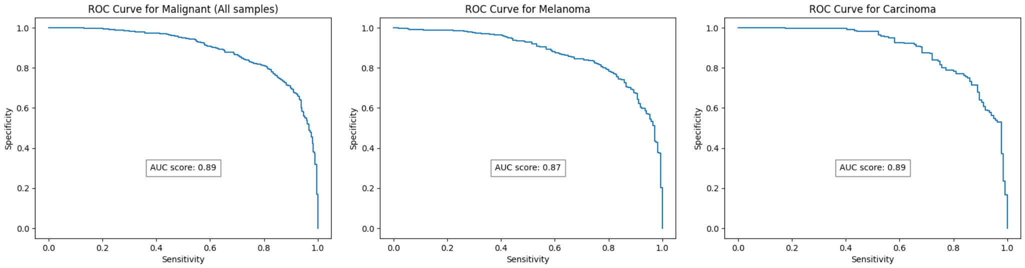


Figure 1: ROC curve for Class Imbalance CNN

As demonstrated in Table 1 in Appendix, the second CNN (class imbalance CNN) significantly outperforms the paper's model. This improved performance can be attributed to several factors. Firstly, the paper's model hyper-parameters were tailored to a considerably larger dataset (127,462 training and validation images) compared to ours (10,015 training and validation images).

In addition to the dataset size discrepancy, the stark class imbalance in our dataset also contributes to the performance gap. While Esteva et al.'s model restricts the number of samples per class, it does not oversample the minority class. Conversely, the second model employs a loss function specifically designed to address class imbalances, assigning weights to each class during model fitting. The contrasting sampling strategies, weight assignments, and the new loss function likely account for the performance difference. Further, incorporating an extra dropout layer and switching to Adam's optimizer in the second model may have facilitated loss reduction, ultimately resulting in higher accuracy. Overall, the second model appears better suited for handling smaller datasets with severe class imbalances than Esteva et al.'s model.

Our findings underscore the critical role of dataset characteristics in training a CNN. Initially, we naively applied a high-impact scientific paper's training procedure to a new dataset, yielding unsatisfactory results. However, by considering our dataset's class imbalance and size, we devised a new training procedure that more closely replicated the renowned Esteva et al. CNN's results.

5.2 ViT Results

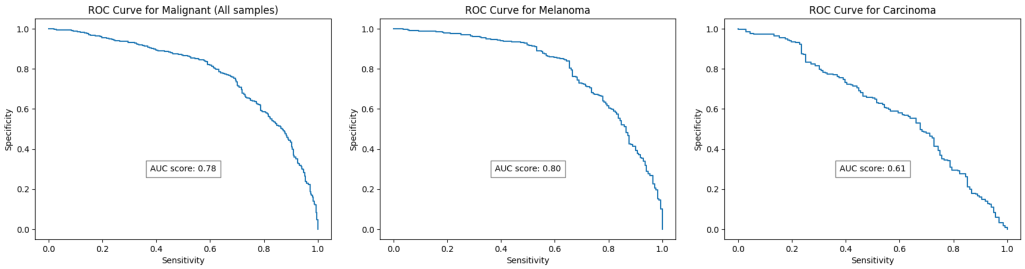


Figure 2: ROC curve for ViT

The ViT model converged quickly during training, with the training loss plateauing after just one epoch (see Appendix Figure 5). However, the model's performance, as demonstrated by the ROC curve (Figure 2) and confusion matrix (Appendix Figure 6b), is subpar. We attribute this to the fact that the pre-trained weights of the ViT model are frozen during training, meaning that only the last classification layer (a single layer MLP) is being fine-tuned. Since the architecture of the transformer is quite complex, with approximately 86 million parameters (Dosovitskiy et al., 2021), the single-layer MLP of the classification layer may not be sufficient to recognize skin lesion images effectively.

When we unfreeze the pre-trained weights, however, the ViT model consistently predicts one class (see Appendix Figure 7), indicating a failure to distinguish between different skin lesion classes. We believe that this is primarily due to our limited training dataset, which may not be large enough to support effective learning across such a complex neural network. If we can obtain a larger and more balanced dataset, we expect that unfreezing the pre-trained weights will improve the ViT model's performance.

5.3 CNN vs. ViT Analysis

We assessed the performance of CNN (the Class Imbalance CNN) and Vision Transformers (ViT) on a test dataset by comparing their confusion matrices and using carcinoma, melanoma, and malignancy classification ROC and AUC scores as performance metrics. Overall, CNN surpasses ViT in all prediction tasks (see Table 3 in Appendix for details).

Analyzing the confusion matrix (see Figure 6) reveals that ViT performs well for classes with sufficient training data but struggles with classes lacking data. While ViT's performance in predicting malignancy and melanoma is almost on par with CNN's, it falls short in carcinoma prediction, where CNN outperforms ViT by 0.28. This discrepancy stems from the limited data available for training a ViT model for carcinoma compared to the relatively abundant melanoma data, indicating that CNN achieves better results with less data.

Furthermore, CNN exhibits a higher image-specific inductive bias than ViT. Each layer in a CNN features locality, a two-dimensional neighborhood structure, and translation equivariance. In contrast, ViT displays locality and translation equivalency only in its MLP layers, while its self-attention layers are global (Dosovitskiy et al., 2021). This difference in inductive bias could be the underlying reason for ViT's inferior performance compared to CNN.

6 Conclusion

In conclusion, our study found that the second CNN (Class Imbalance CNN) significantly outperforms Esteva & al.'s model, particularly when handling smaller datasets with severe class imbalances. Dataset characteristics played a crucial role in achieving better results by adapting the training procedure. In comparison, the ViT model's rapid convergence and suboptimal performance can be attributed to frozen pre-trained weights and insufficient training data, which hinder its ability to effectively recognize skin lesion images and distinguish between different classes.

Lastly, CNN outperformed ViT in all prediction tasks, achieving better results with fewer data due to its higher image-specific inductive bias, while the difference in inductive bias likely contributed to ViT's weaker performance. Our findings emphasize the significance of considering dataset properties and inductive biases when selecting between CNN and ViT for image classification tasks.

Our findings are valuable insights in training deep-learning models to effectively classify skin lesions and, perhaps, other types of disease. Consequently, it could prevent more people from losing their life to treatable illnesses.

7 Acknowledgement

ChatGPT and GPT-4 were used for help with wording, formatting, and styling throughout this work.

References

Aladhadh, S., Alsanea, M., Aloraini, M., Khan, T., Habib, S., & Islam, M. (2022). An Effective Skin Cancer Classification Mechanism via Medical Vision Transformer. Sensors (Basel, Switzerland), 22(11), 4008. https://doi.org/10.3390/s22114008

Codella, N., Rotemberg, V., Tschandl, P., Celebi, M. E., Dusza, S., Gutman, D., Helba, B., Kalloo, A., Liopyris, K., Marchetti, M., Kittler, H., & Halpern, A. (2018). Skin Lesion Analysis Toward Melanoma Detection 2018: A Challenge Hosted by the International Skin Imaging Collaboration (ISIC). https://arxiv.org/abs/1902.03368

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., . . . Houlsby, N. (2021). An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. ArXiv [Cs.CV]. Retrieved from http://arxiv.org/abs/2010.11929

Esteva, A., Kuprel, B., Novoa, R. A., Ko, J. S., Swetter, S. M., Blau, H. M., & Thrun, S. (2017). Dermatologist-level classification of skin cancer with deep neural networks. Nature, 542(7639), 115–118. https://doi.org/10.1038/nature21056

OpenAI. (2023). ChatGPT (Mar 23 version) [Large language model]. https://chat.openai.com/chat

Skin cancer – IARC. (n.d.-b). https://www.iarc.who.int/cancer-type/skin-cancer/

Tschandl, P., Rosendahl, C. & Kittler, H. (2018). The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. Sci. Data 5, 180161 doi:10.1038/sdata.2018.161

Wang, B., Ba, J. (2023). Convolutional Neural Networks & Image Classification [Lecture notes]. The University of Toronto. https://uoft-csc413.github.io/2023/assets/slides/Lec04.pdf

Xin, C., Liu, Z., Zhao, K., Miao, L., Ma, Y., Zhu, X., Zhou, Q., Wang, S., Li, L., Yang, F., Xu, S., &; Chen, H. (2022). An improved transformer network for skin cancer classification. Computers in Biology and Medicine, 149, 105939. https://doi.org/10.1016/j.compbiomed.2022.105939

Appendix

A: Graphs and Plots

A1

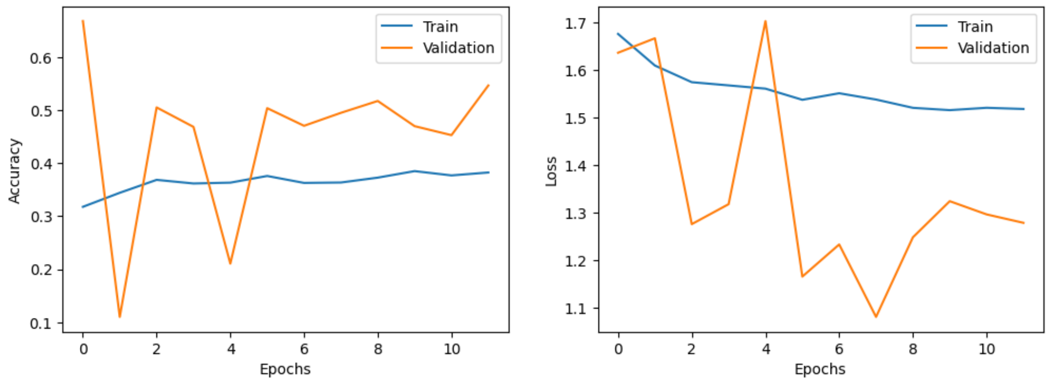


Figure 3: Training plot for Esteva & al's CNN on our dataset

A2

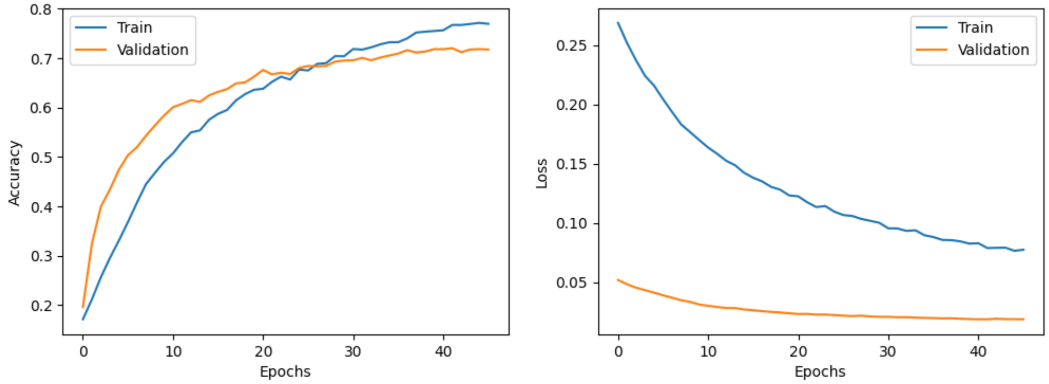


Figure 4: Training plot for Class Imbalance CNN

A3

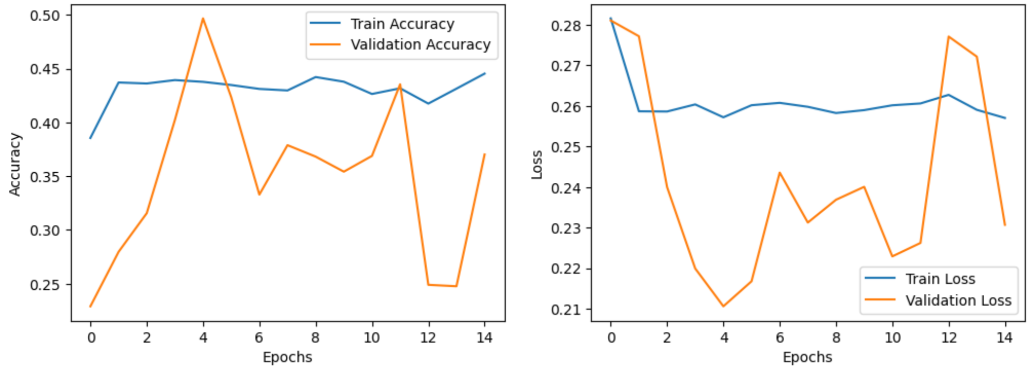


Figure 5: Training plot of ViT

B: Tables

Table 1: Note that the results presented for Esteva & al's CNN are the results of their method on our dataset, not the original results from the paper.

Test	Esteva & al's CNN	Class Imbalance CNN
Malignant AUC	0.79	0.89
Melanoma AUC	0.84	0.87
Carcinoma AUC	0.64	0.89
two-way accuracy	56.75%	81.68%
seven-way accuracy	44.51%	69.84%

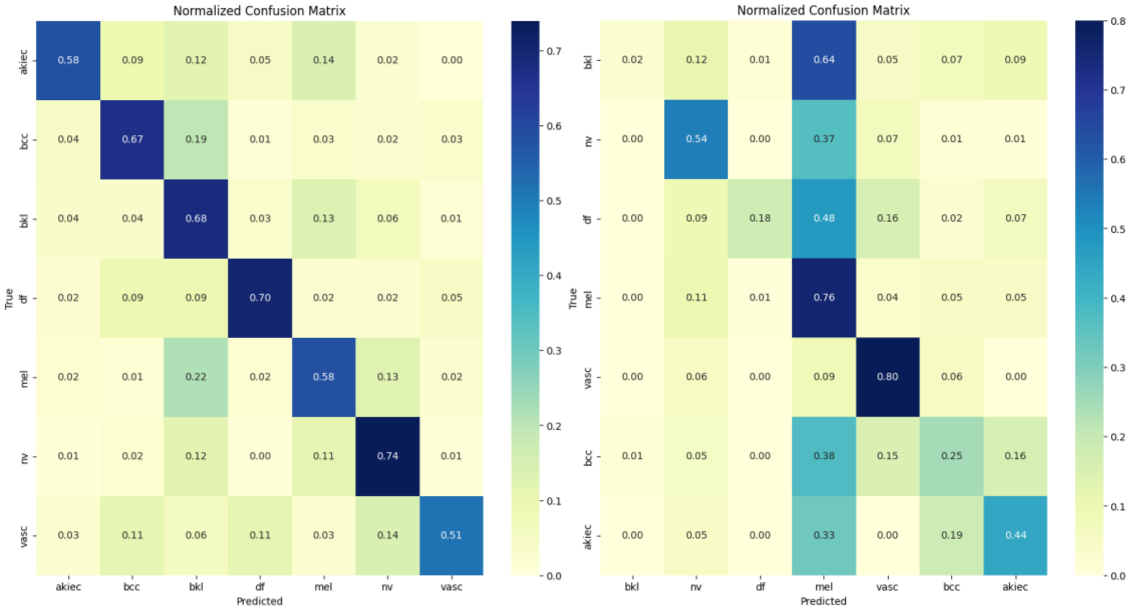
Table 2: ViT Evaluation Results

Test	ViT
Malignant AUC	0.78
Melanoma AUC	0.80
Carcinoma AUC	0.61
two-way accuracy	75%
seven-way accuracy	46.36%

Table 3: Comparison of Esteva & al's CNN, Class Imbalance CNN, and ViT

Test	Esteva & al's CNN	Class Imbalance CNN	ViT
Malignant AUC	0.79	0.89	0.78
Melanoma AUC	0.84	0.87	0.80
Carcinoma AUC	0.64	0.89	0.61
two-way accuracy	56.75%	81.68%	75%
seven-way accuracy	44.51%	69.84%	46.36%

C: Confusion Matrix



(a) Class Imbalance CNN Confusion Matrix

(b) ViT Confusion Matrix

Figure 6: Confusion Matrix Comparison

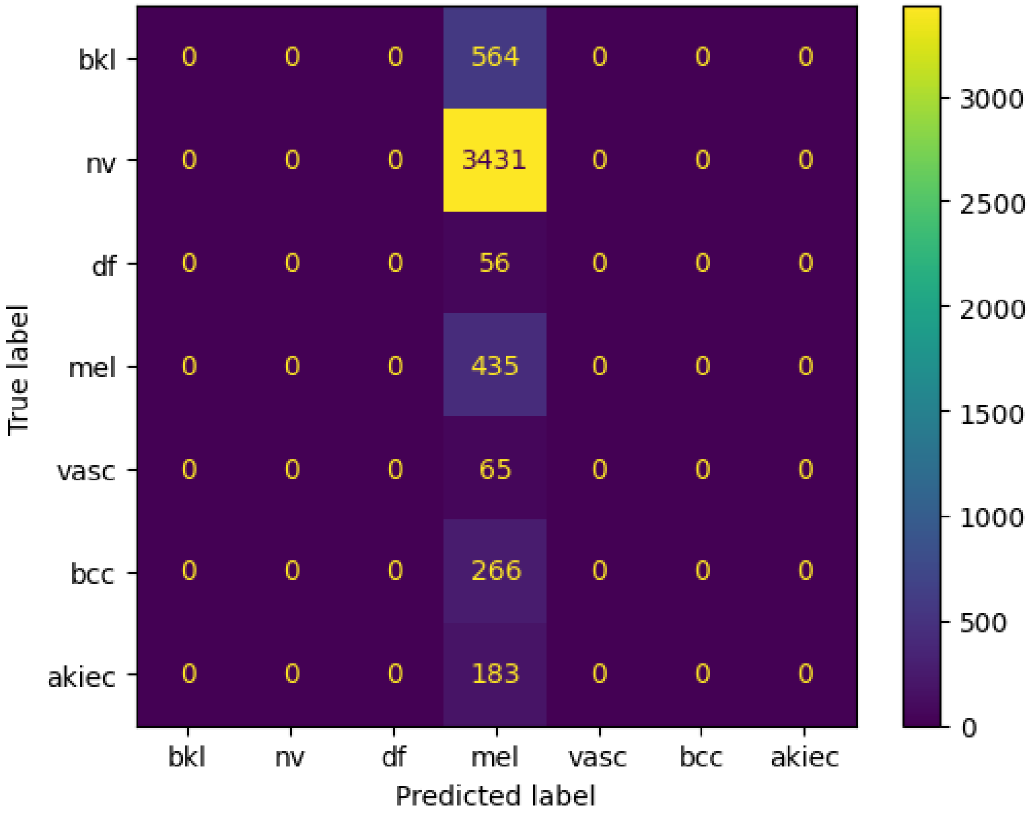


Figure 7: Confusion Matrix of ViT if pre-trained weights are unfrozen

D: Algorithm Box

Algorithm 1 Training Loop for Class Imbalance CNN

```
Sampling Strategy = {nv:1000, mel: 946, bkl: 934, bcc: 437, akiec: 278, vasc: 120, df: 98}
total_epochs $\leftarrow$ 60
early_stopping $\leftarrow$ EarlyStopping(monitor='val_loss', patience=4, ...)
model_checkpoint $\leftarrow$ ModelCheckpoint('best_model.Adam', monitor='val_loss', ...)
for epoch in range(total_epochs) do
    train_generator.on_epoch_end()            ▷ Resample using oversampler and undersampler
    for batch_idx, (X, y) in enumerate(train_generator) do
        X_augmented, y_augmented $\leftarrow$ train_generator.__data_generation(X, y)
        Compute loss using focal loss (gamma: 2.0, alpha: 0.25)
        Update model weights using Adam optimizer and backpropagation
        if batch_idx $\geq$ train_generator.n // train_generator.batch_size then
            break
        end if
    end for
    Compute validation loss and accuracy using valid_generator
    Update early_stopping and model_checkpoint based on validation loss
    if early_stopping condition is met then
        break
    end if
end for
```

Algorithm 2 Pipeline for a single image of the ViT model

```
Input1 image: with size (224,224,3)
Input2 true_label: ranging from 0 to 6, representing each of bkl, nv, df, mel, vasc, bcc, akiec
patches $\leftarrow$ ConvLayer(image)
flattened_patches $\leftarrow$ Flatten(patches)
sequence_of_embeddings $\leftarrow$ ProjectionLayer(flattened_patches)
transformer_input $\leftarrow$ [CLS] || sequence_of_embeddings
transformer_input $\leftarrow$ transformer_input + positional_encoding
transformer_output $\leftarrow$ TransformerEncoder(transformer_input)
CLS_token $\leftarrow$ transformer_output[0]
prediction_logits $\leftarrow$ ClassificationLayer(CLS_token)
loss $\leftarrow$ FocalLoss(prediction_logits, true_label)
backpropagate(loss)
```