Article

SKIN LESION CLASSIFICATION ON IMBALANCED DATA USING DEEP LEARNING WITH SOFT ATTENTION

Viet Dung Nguyen [1,†,*], Ngoc Dung Bui [2,†] and Hoang Khoi Do [1,†]

[1] School of Electrical and Electronic Engineering, Hanoi university of Science and Technology, Vietnam; dung.nguyenviet1@hust.edu.vn
[2] Faculty of Information Technology, University of Transport and Communications, Vietnam; dnbui@utc.edu.vn
[3] School of Electrical and Electronic Engineering, Hanoi university of Science and Technology, Vietnam; khoi.dh200322@sis.hust.edu.vn
* Correspondence: dung.nguyenviet1@hust.edu.vn; Tel.: +84-9834-443-22 (N.V.D.)
† Current address: 1st Dai Co Viet Street, Ha Noi, Vietnam

Abstract: Today, the rapid development of industrial zones leads to an increased incidence of skin diseases because of polluted air. According to a report by the American Cancer Society, it is estimated that in 2022 there will be about 100,000 people suffering from skin cancer and more than 7600 of these people will not survive. In the context that doctors at provincial hospitals and health facilities are overloaded, doctors at lower levels lack experience and having a tool to support doctors in the process of diagnosing skin diseases quickly and accurately is essential. Along with the strong development of artificial intelligence technologies, many solutions to support the diagnosis of skin diseases have been researched and developed. In this paper, a combination of SOTA model such as DenseNet, InceptionNet, ResNet, NasNet, and MobileNet and Soft-Attention is proposed. Furthermore, personal information including age and gender are also used. It is worth to note that a new loss function that takes into account the data imbalance is also proposed. Experimental results on data set HAM10000 show that using InceptionResNetV2 with Soft-Attention and new loss function gives 90 percent accuracy, mean of precision, f1-score, recall-score, and AUC scores of 0.81, 0.81, 0.82, and 0.989, respectively. Besides, using MobileNetV3Large combined with Soft-Attention and new loss function, even though the number of parameters is 11 times less, the number of hidden layers is 4 times less, it achieves 0.86 accuracy and 30 times faster diagnosis than InceptionResNetV2.

Keywords: Skin Lesions, Classification, Deep Learning, Soft-Attention, Imbalance

Citation: Nguyen, V.D.; Bui, N.D.; Do, H.K. SKIN LESION CLASSIFICATION ON IMBALANCED DATA USING DEEP LEARNING WITH SOFT ATTENTION. *Sensors* **2022**, *1*, 0. https://doi.org/

Received:
Accepted:
Published:

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

1. Introduction

1.1. Problem Statement

Skin cancer is one of the most common cancers leading to worldwide death. Every day, more than 9500 [14] people in the United States are diagnosed with skin cancer. Otherwise, 3.6 [14] million people are diagnosed with basal cell skin cancer each year. According to the Skin Cancer Foundation, the global incidence of skin cancer continues to increase [13]. In 2019, it is estimated that 192,310 cases of melanoma will be diagnosed in the United States [13]. On the other hand, if patients are early diagnosed, the survival rate is correlated with 99 percent. However, once the disease progresses beyond the skin, survival is poor [13]. Moreover, with the increasing incidence of skin cancers, low awareness among a growing population, and a lack of adequate clinical expertise and services, there is a need for effective solution.

Recently, deep learning particularly, and machine learning in general algorithms have emerged to achieve excellent performance on various tasks, especially in skin disease diagnosis tasks. AI-enabled computer-aided diagnostics (CAD) has solutions in three main categories: Diagnosis, Prognosis, and Medical Treatment. Medical imaging, including ultrasound, computed tomography, magnetic resonance imaging, and X-ray image is used extensively in clinical practice. In Diagnosis, Artificial Intelligence (AI) algorithms are applied for disease detection to save progress execution before these diagnosis results are considered by a doctor. In Prognosis, AI algorithms are used to predict the survival rate of a patient based on his/her history and medical data. In Medical Treatment, AI models are applied to build solutions to a specific disease, medicine revolution is an example. In various studies, AI algorithms have provided various end-to-end solutions to the detection of abnormalities such as breast cancer, brain tumors, lung cancer, esophageal cancer, skin lesions, and foot ulcers across multiple image modalities of medical imaging [13].

To adapt the rise in skin cancer case, AI algorithms over the last decade has a great performance. Some typical models that can be mentioned are DenseNet [17], EfficientNet [20], Inception [19], MobileNets[20][21][22], ResNet [23] [24], and NasNet [33]. Some of these models which have been used as a backbone model in this paper will be discussed in the Related Work section.

1.2. Related Works

Skin lesion classification is not a new area, since there are many great performance models constructed. One of the most cutting-edge technologies that have been used is Soft-Attention as stated in [14]. Soumyyak et al construct several models formed by the combination of a backbone model including DenseNet201 [17], InceptionResNetV2 [19], ResNet50 [23] [24], VGG16 [25] and Soft-Attention layer. Their approach is to add the Soft-Attention layer at the end or the middle of the backbone model. For ResNet50 and VGG16, the Soft-Attention layer is added after the third residual block and CNN block, respectively. DenseNet201 and InceptionResNetV2, otherwise concatenate with the Soft-Attention before a fully-connected layer, and then soft-max layer.

Using those above backbones has been tried by many previous papers. Rishu Garg et al [3] uses transfer learning approach with CNN based model: ResNet50 and VGG16 which are pretrained with ImageNet data set. Besides, they also use data augmentation to avoid the imbalance of the data set. Histogram Equalization is also used to increase the contrast of the skin lesions before feeding into the Machine Learning algorithms including Random Forest, XGBoost, Support Vector Machine.

Amirreaza et al [5] do not only use those above backbone model but also used InceptionV3 [19] model. In that research, the dataset HAM10000 and PH^2 are combined to create a 8 classes data set. Before feeding into the Deep CNN models, the image is resized to (224, 224) for DenseNet201, ResNet152, InceptionResNetV2 and (229, 229) for InceptionV3.

Another paper that uses the backbone models is [9], Hemanth et al decide to use EfficientNet [18] and SeNET [35] instead and CutOut [36] method which involves creating holes of different sizes on the images i.e. technically making a random portion of image inactive during data augmentation process.

Otherwise, [12] also used Deep Convolution Neural Network, Peng Yao et al used RandArgument which crops an image into several images from a fixed size, DropBlock which is used for regularization, Multi-Weighted New Loss which is used for dealing with the imbalanced data problem, end-to-end Cumulative Learning Strategy which can more effectively balance representation learning and classifier learning without additional computational cost.

Another state of the art is GradCam and Kernel SHAP [6], Kyle Young et al create model agnostic, local interpretable methods that can highlight pixels that the trained network deems relevant for the final classification. In that research they use three data sets containing HAM10000, BCN-20000 and MSK. Before feeding into the models, the images are preprocessed by binarized with a very low threshold to find the center of mass.

On the other hand, there are also many state of the art whose great performance on skin lesion classification. The Student and Teacher Model is also a high performance model in 2021 [2], which is created by Xiaohan Xing et al as the combination of two model which share the memory with each other. Therefore, they can take full advantage of what others learn.

Research	Deep Learning	Machine Learning	Data Augmentation	Feture Extraction	Data set
[14]	x		x		HAM10000
[3]	x	x	x	x	HAM10000
[5]	x	x	x		HAM10000, PH^2
[9]	x		x		HAM10000
[12]	x		x		HAM10000
[6]	x		x	x	HAM10000, BCN-20000, MSK
[2]	x		x		HAM10000
[15]	x		x		HAM10000
[16]	x		x		HAM10000
[10]	x		x		HAM10000
[8]		x	x	x	HAM10000

Table 1. Related Works Summary.

SkinLinkNet [15] and WonderM [16] are both tested the effect of segmentation on skin lesion classification problem created by Amirreza et al and Yeong Chan et al, respectively. In WonderM, the method used is padding the image so that the image has the shape increased from (450, 600) to (600, 600). In SkinLinkNet, instead resize the image down to (448, 448). Both of SkinLinkNet and WonderM use UNet to do the segmentation task, though they use EfficientNetB0 and DenseNet to do the classification task, respectively.

Another approach is using metadata including gender, age, and capturing position as stated in [10] by Nil Gessert et al. The metadata is fed into fully connected neural network after being encoded into one-hot vector. All missing data point of age is set to 0. To overcome the missing data problem, the research apply one-hot encoding to the group but the initial validation is poor performance then numerical encoding is applied.

On the other hand, skin lesion classification problems are not only applied by Deep Learning but also Machine Learning. Random Forest, XGBoost, and Support Vector Machines are tested by [3] of Rishu Garg et al. Besides, Isolation Forest is applied before the soft-max activation of the deep learning model to detect out of distribution skin lesion images as stated in [5] by Amirreza Rezvantalab et al. Matrix Transformation, besides is also applied before the soft-max activation function in [8] by Michele Alberti et al.

1.3. Proposed Method

In this research, a new model is constructed from the combination of:
- Backbone model including DenseNet201, InceptionResNetV2, ResNet50/152, NasNetLarge, NasNetMobile, and MobileNetV2/V3
- Using metadata including age, gender, localization as another input of the model
- Using Soft-Attention as a feature extractor of the model
- A new weight loss function

2. Materials and Methods

2.1. Materials

2.1.1. Image Data

The data set used in this paper is the HAM10000 data set published by Havard University Dataverse [7]. There are total 7 classes in this data set containing Actinic keratoses and intraepithelial carcinoma or Bowen's disease (AKIEC), Basal cell Carcinoma (BCC), benign keratosis-like lesions (solar lentigines / seborrheic keratoses andchen-planus like keratoses, BKL), dermatofibroma (DF), melanoma (MEL), melanocytic nevi (NV), and vascular lesions (angiomas, angiokeratomas, pyogenic granulomas and hemorrhage, VASC). The distribution of the data set is shown in the Table 2 below:

Class	AKIEC	BCC	BKL	DF	MEL	NV	VASC	Total
No. Sample	327	514	1099	115	1113	6705	142	10015

Table 2. Data Distribution in HAM10000.

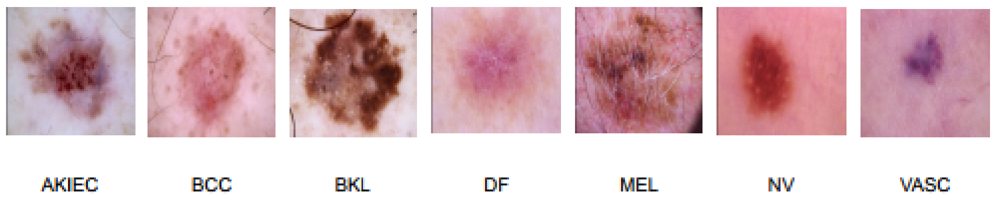

Figure 1. Example image of each class

More than 50 percent of lesions are confirmed through histopathology (HISTO), the ground truth for the rest of the cases is either follow-up examination (FOLLOWUP), expert consensus (CONSENSUS), or confirmation by in-vivo confocal microscopy (CONFOCAL). On the other hand, before being used for training the whole data is shuffled then split into two part. 90 percent and 10 percent of the data is used for training and validating respectively. Images in this data set has the type of RGB and shape of (450, 600). However, Each backbone need the different input size of image as well as the range of pixel value.

2.1.2. Metadata

The HAM10000 data set [7] also contain the metadata of patient including gender, age, and the capturing position illustrated in Table 3

ID	Age	Gender	Local
ISIC-00001	15	Male	back
ISIC-00002	85	Female	elbow

Table 3. Metadata example in the data set

2.2. Methodology

2.2.1. Overall Architecture

The whole architecture of the model is represented in the Figure 2. The model takes two input including Image data and Metadata. Metadata branch, otherwise is preprocessed before feeding into a dense layer then concatenate with the output of Soft-Attention layer.

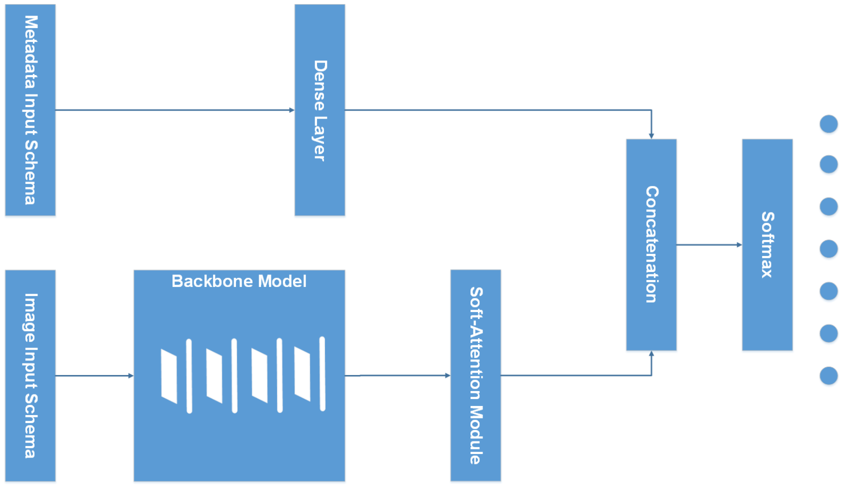

Figure 2. Overall Model Architecture

The Figure 3 illustrates the overall structures of the combination of backbone models and Soft-Attention, which is used in this research. In detailed the combination of DenseNet201 and Soft-Attention is formed by replacing the three last DenseBlock, Global Average Pooling, and the fully-connected layer with the Soft-Attention Module. Similarity, ResNet50 and ResNet152 is also replaced the three last Residual Block, Global Average Pooling, and the fully connected layer with the Soft-Attention module. InceptionResNetV2, on the other hand, is replaced the Average Pool and the last Dropout with the Soft-Attention Module. Besides, the two last Normal Cell in NasNetLarge is replaced with the Soft-Attention module. 137 138 139 140 141 142 143 144 145

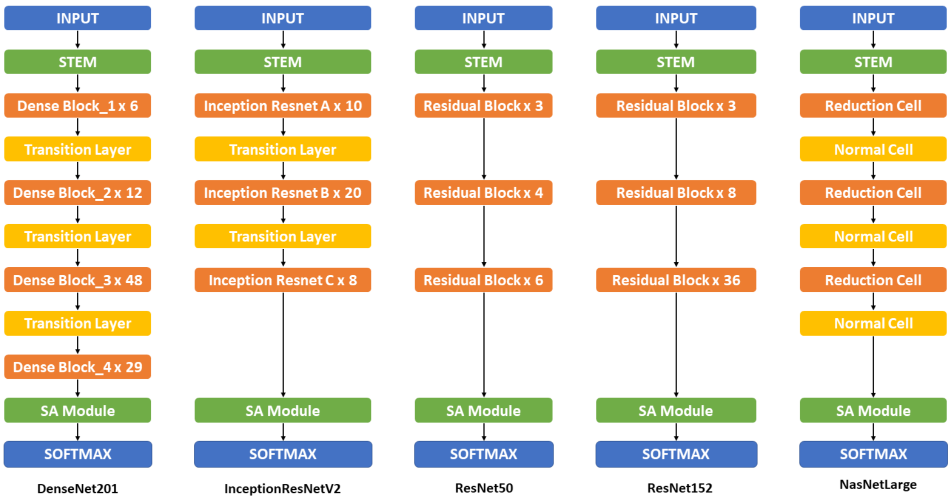

Figure 3. Overall Original Model Architecture. This figure show the overall structure of the backbone model (non mobile-based model) including DenseNet201, InceptionResNetV2, ResNet50, ResNet152, and NasNetLarge. The detail structure and information can be found at the Apendix A1

The Figure 4, on the other hand shows the detailed structure of mobile-based mobile and its combination with Soft-Attention. All of the MobileNet versions combine with the Soft-Attention module by replacing the two last convolution 1x1 by Soft-Attention module. The NasNetMobile, otherwise, combine with the Soft-Attention module by replacing the last Normal Cell. 146 147 148 149 150

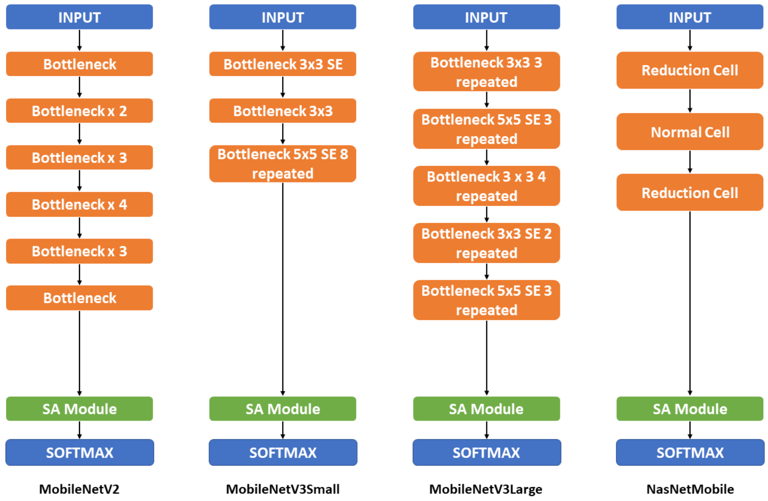

Figure 4. Overall Mobile-based Model Architecture. This figure show the overall structure of the mobile-based backbone model including MobileNetV2, MobileNetV3Small, MobileNetV3Large, and NasNetMobile. The detail structure and information can be found at the Apendix A2

2.2.2. Input Schema

In this research, the image data is both augmented for all class, the number of image increase to 18015 images and keep original form. Before feeding into the backbone model, the images is pre-processed by the input requirement of each model. DenseNet201 [17] require the input pixels values are scaled between 0 and 1 and each channel is normalized with respect to the ImageNet data set. In Resnet50 and Resnet152 [23] [24], the images are converted from RGB to BGR, then each color channel is zero-centered with respect to the ImageNet data set, without scaling. InceptionResNetV2[18], on the other hand, will scale input pixels between -1 and 1. Similarly, three versions of MobileNet [20] [21] [22], NasNetMobile and NasNetLarge [33] require the input pixel is in range of -1 and 1.

On the other hand, the metadata is also used as another input. In the research [10], they decide to keep the missing value and set its value to 0. The sex and anatomical site are categorical encoded. The age, on the other hand is numerical normalized. After processing, the metadata is fed into a two-layer neural network with 256 neurons each. Each layer contains batch normalization, a ReLU [34] activation, and dropout with $p = 0.4$. The network's output is concatenated with the CNN's feature vector after global average pooling. Especially, they use a simply data augmentation strategy to address the problem of missing values in metadata. During training, they randomly encode each property as missing with a probability of $p = 0.1$.

In this research, the unknowns is kept as a type as discussed in Metadata section. Sex, anatomical site and age are also category encoded and numerical normalized, respectively. After processing, the metadata is then concatenated and fed into a dense layer of 4096 neurons. Finally, this this dense layer is then concatenate with the output of Soft-Attention which is then discussed in Soft-Attention section. The Input schema is described in the Table 5

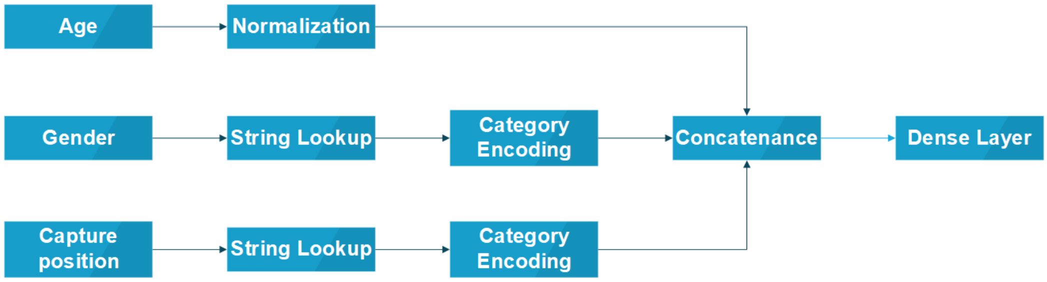

Figure 5. Input Schema

2.2.3. Backbone Model

In this paper, the backbone models used in this paper are DenseNet201 [17], Inception [19], MobileNets [20] [21] [22], ResNet [23] [24], and NasNet [33]. The combination of DenseNet201, InceptionResNetV2 and Soft-Attention layer are both tested by the previous paper [14] with a great performance. Otherwise, Resnet50 also well classify but with much less number of parameter and depth than based on its f1-score and precision stated. Therefore, in this paper, the performance of the model Resnet152 and NasnetLarge which has the larger number of parameter and depth is analyzed. On the other hand, three version of MobileNet and the NasnetMobile will also be analyzed which has a small number of parameter and depth.

Model	Size(MB)	Parameters	Depth
Resnet50	98	25.6M	107
Resnet152	232	60.4M	311
DenseNet201	80	20.2M	402
InceptionResNetV2	215	55.9M	449
MobileNet	16	4.3M	55
MobileNetV2	14	3.5M	105
MobileNetV3Small	Unknown	2.5M	88
MobileNetV3Large	Unknown	5.5M	118
NasnetMobile	23	5.3M	308
NasnetLarge	343	88.9M	533

Table 4. Size and Parameters and Depth of backbone model used in this paper.

2.2.4. Soft-Attention Module

Soft-Attention has been used in various applications: image caption generation such as [28] or handwriting verification [29]. Soft-Attention can ignore irrelevant areas of the image by multiplying the corresponding feature maps with low weights. Soft-Attention is described in below equation:

$$f_{sa} = \gamma t \sum_{k=1}^{K} softmax(W_k * t)$$

The Figure 6 shows the two main steps of applying Soft-Attention is applied in two main steps. Firstly, the input tensor is put in grid-based feature extraction from the high-resolution image, where each grid cell is analyzed in the whole slide to generate a feature map [30]. This feature map called $t \in R^{h \times w \times d}$ where $h, w,$ and d is the shape of tensor generated by a Convolution Neural Network (CNN), is then input to a 3D convolution layer whose weights is $W_k \in R^{h \times w \times d \times K}$. The output of this convolution is normalized using the soft-max function to generate K (a constant value) attention maps. These K attention maps are aggregated to produce a weight function called α. This α function is then multiplied with feature tensor t and scaled by γ, a learnable scalar. Finally, the output of Soft-Attention function f_{sa} is the concatenation of the beginning feature tensor t and the scaled attention maps.

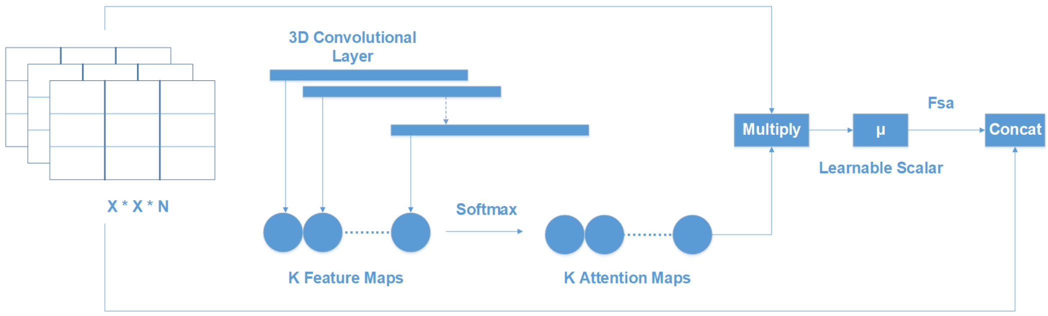

Figure 6. Soft-Attention Layer

In this research, the Soft-Attention layer is applied in the same way in this research [14]. The Soft-Attention module is described in the figure 7

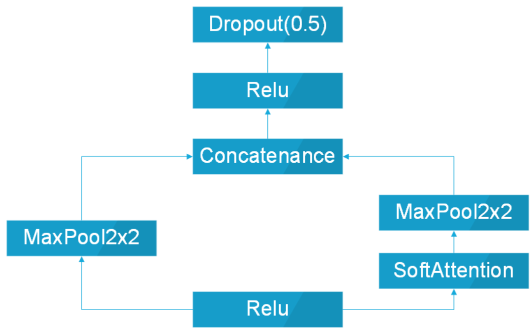

Figure 7. Soft-Attention Module

After feeding into ReLU function layer, the heat feature map is processed in two paths. The first path is the 2-dimensional Max Pooling. In the second path, the feature map, on the other hand is fed into the Soft-Attention Layer before the 2-dimensional Max Pooling. After all, these two paths are then concatenated, fed into a ReLU layer, and a Dropout with the probability of 0.5.

2.2.5. Loss Function

The loss function used in this paper is categorical cross-entropy. Consider $X = [x_1, x_2, \ldots, x_n]$ as the input feature, $\theta = [\theta_1, \theta_2, \ldots, \theta_n]$. Let N, and C is the number of training examples and number of class respectively. The categorical cross-entropy loss is presented as:

$$L(\theta, x_n) = -\frac{1}{N} \sum_{c=1}^{C} \sum_{n=1}^{N} W_c \times y_n^c \times \log(\hat{y}_n^c)$$

where $\hat{y}_i^c$ is the output of model and y_i^c is the target that the model should return, W_c is the weight of class c. Since the data set face the imbalanced problem then class weight for the loss is applied. In this research, both of the original weight and a new weight formula is implementation. Originally, the weight is calculated by taking the inverse of percentage that each class accounts for. The new weight formula is described as follow:

$$W = N \odot D$$

$$D = \begin{bmatrix} \frac{1}{C \times N_1} & \frac{1}{C \times N_2} & \cdots & \frac{1}{C \times N_n} \end{bmatrix} = \frac{1}{C} \odot \begin{bmatrix} \frac{1}{N_1} & \frac{1}{N_2} & \cdots & \frac{1}{N_n} \end{bmatrix}$$

where N is the number of training sample, C is the number of class, N_i is the number of sample in each class i. D is the matrix contain the inverse of $C \times N_i$.

3. Results

3.1. Experimental Setup

3.1.1. Training

Before training, the data set is split into two sub set for training (90 percent) and validation (10 percent). Test set, otherwise is provided by the HAM10000 data set, contains 857 images. To analyze the effect of augmented data on the model, during the training the image data is augmented by the following technique:
- Rotation Range: rotate the image in a fixed angle.
- Width and height shift range: Shift the image horizontally and vertically, respectively.
- Zoom Range: Zoom in or zoom out the image to create new image.
- Horizontal and vertical flipping: Flipping the image horizontally and vertically to create new image.

Otherwise, all of models is trained with the Adam Optimizer [27] with the learning rate of 0.001 which is reduced by a factor of 0.2 to a minimum learning rate of $0.1x10^6$, and the epsilon is set to 0.1. The initial epochs is set to 250 epochs and the Early Stopping is also applied to stop the training as the accuracy of validation set does not increase after 25 epochs. Besides, the batch size is set to 32.

3.1.2. Tools

TensorFlow and Keras are two of the most popular framework to build deep learning model. In this research, Keras based on TensorFlow is used to build, clone the backbone model which is pre-trained with the image-net data set. Otherwise, the models are trained by NVIDIA RTX TitanV and the data set is pre-processed with the CPU Intel I5 32 processors, RAM 32GB. In detail, the GPU is setup with CUDA 11.6, cuDNN 8.3 and ChipSRT as the requirement of TensorFlow version 2.7.0.

3.1.3. Evaluation Metrics

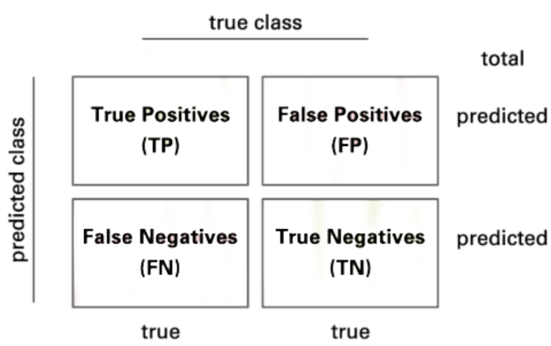

Figure 8. Confusion Matrix

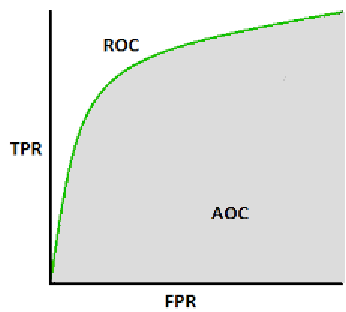

Figure 9. Area Under the Curve

The model is evaluated by using the confusion matrix and related metrics. The Figure 8 illustrates the presentation of a 2×2 confusion matrix used for 2 class. Consider a confusion matrix A with C number of class. Let A^i and A^j is the set of A rows and columns respectively, therefore A^i_k is the element at row i and column k

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1j} \\ a_{21} & a_{22} & \dots & a_{2j} \\ \vdots & \vdots & & \vdots \\ a_{i1} & a_{i2} & \dots & a_{ij} \end{bmatrix}$$

The True Positive(TP) of all class in this case is the main diagonal of the matrix A. The following method are used to calculate the False Positives(FP), False Negatives(FN), and True Negatives(TN) of all class:

$$FP = -TP + \sum_{k=1}^{i} A_k^i \qquad FN = -TP + \sum_{k=1}^{j} A_k^j$$

$$TN_c = \sum_{i=1}^{C}\sum_{j=1}^{C} a_{ij} - \left[\sum_{k=1}^{i} A_{i=ck}^i + \sum_{k=1}^{j} A_{j=ck}^j\right] + a_{i=cj=c} \implies TN = \begin{bmatrix} TN_1 & TN_2 & \dots & TN_c \end{bmatrix}$$

Then, the model is evaluated by the following metrics:

$$\text{Sensitivity(Sens)} = \frac{TP}{TP + FN} \qquad \text{Specificity(Spec)} = \frac{TN}{TN + FP}$$

where Sensitivity and Specificity mathematically describe the accuracy of a test which reports the presence or absence of a condition. Individuals for which the condition is satisfied are considered "positive" and those for which it is not are considered "negative". Sensitivity or true positive rate refers to the probability of a positive test, conditioned on truly being positive while Specificity or true negative rate refers to the probability of a negative test, conditioned on truly being negative.

$$\text{Precision} = \frac{TP}{TP + FP} \qquad \text{F1 Score} = \frac{2 \times TP}{2 \times TP + FP + FN + TN}$$

Precision or positive predictive value (PPV) is the probability of a positive test conditioned on both truly being positive or negative. F1-score, on the other hand refers the harmonic mean of precision and recall which mean the higher the f1-score is, the higher both precision and recall is. Besides, the expected value of precision, f1-score and recall-score are also applied because of the multi-class problem.

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN} \qquad \text{Balanced Accuracy} = \frac{\text{Sens} + \text{Spec}}{2}$$

The last metric is the AUC score standing for Area Under the Curve which is the Receiver Operating Curve (ROC) that indicate the probability of TP versus the probability of FP.

3.2. Discussion

According to the Table 5, it is clear that the model trained with metadata has a higher accuracy than the model trained with augmented data only. While InceptionResNetV2 and DenseNet201 trained with augmented data have accuracy of 0.79 and 0.84 respectively, their training with metadata are 0.90 and 0.89, respectively. Furthermore, Resnet50 trained with metadata data has the accuracy that outperform the Resnet50 trained with augmented data and is twice as high as Resnet152 trained with metadata. On the other hand, mobile model including MobileNetV2, MobileNetV3Large, and NasNetMobile, even though has a much smaller number of parameters and depth than the other model, they have a quite good accuracy of 0.81, 0.86, 0.86, respectively.

Model	ACC(AD)	ACC(MD)
InceptionResNetV2	0.79	**0.90**
DenseNet201	0.84	**0.89**
ResNet50	0.76	0.70
ResNet152	-	0.57
NasNetLarge	-	0.84
MobileNetV2	-	0.81
MobileNetV3Small	-	0.78
MobileNetV3Large	-	**0.86**
NasNetMobile	-	**0.86**

Table 5. Accuracy of all models. ACC stands for accuracy. AD stands for augmented data, this indicate that the model is trained with augmented data. MD stands for Metadata which indicate that the model is trained with Metadata

Moreover, the model trained with augmented data does not only have low accuracy but their f1-score and the recall score also are imbalanced according to Figure 10, 11, 12, and 13. As a results, augmented data model does not classify well on all class as InceptionResNetV2 trained on augmented data have f1-score on class df and akiec is just above 0.3 and 0.4, separately while InceptionResNetV2 trained on metadata and the new weight loss can classify well in a balanced way arcoding to the Figure 11. However, only DenseNet201, InceptionResNetV2, and NasNetLarge whose depth are equal or larger than 400 have balanced f1-score on class. The others still face the imbalanced term. Since this data set is not balanced, therefore using augmented data can make the model more bias to the class which has larger sample. Using the metadata, though still make the model bias, it does contributes to the improvement of the performance of the model.

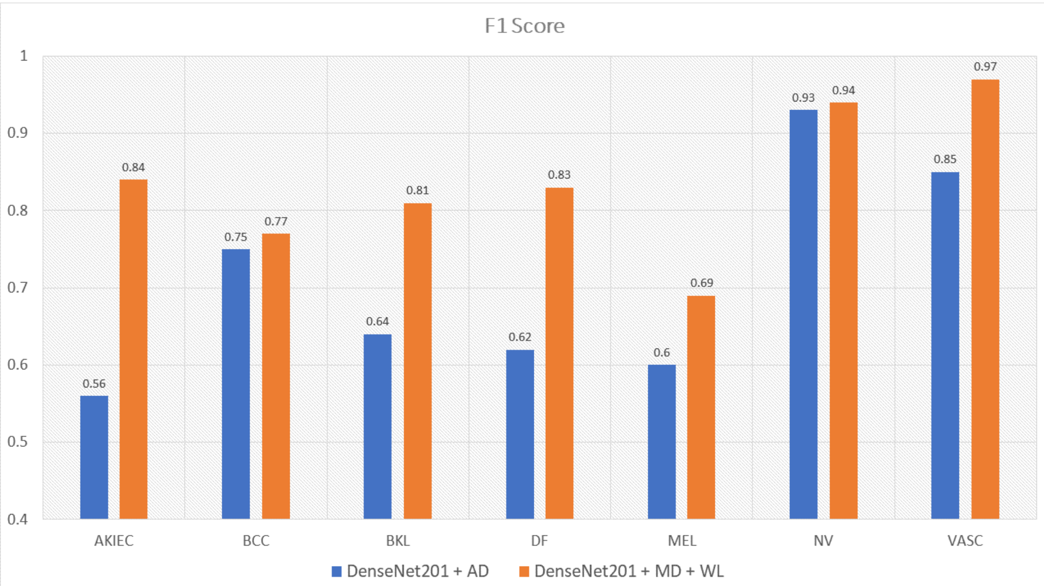

Figure 10. The comparison between f1 scores of DenseNet201 trained with augmented data and the one trained with metadata and weight loss

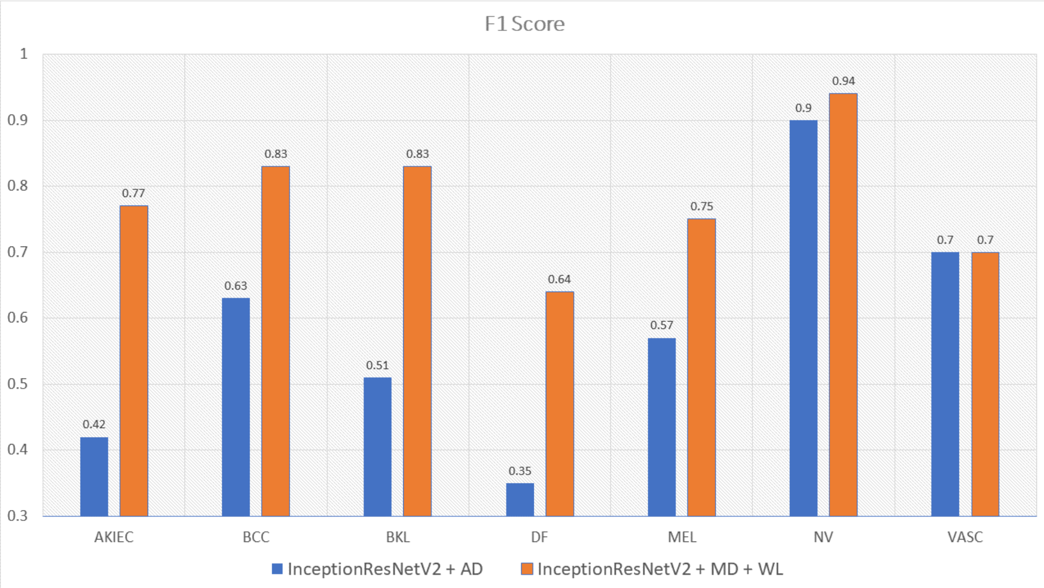

Figure 11. The comparison between f1 scores of InceptionResNetV2 trained with augmented data and the one trained with metadata and weight loss

This problem is also true with the recall score according to Figure 12 and 13. DenseNet201, InceptionResNetV2, trained with augmented data has expected value of recall of 0.56, 0.69, respectively, while the combination of DenseNet201, Metadata and the new weight loss function achieve the expected value of recall: 0.82. Therefore, metadata does improve the model performance by reducing the amount of data needed for achieving higher results. On the other hand, the reason why the model become much more balanced is the weighted loss function. Weight loss function has ability to solve the imbalanced class samples by adding a weight related to the number of samples in each class. DenseNet201, InceptionResNetV2 trained with the new weighted loss function have recall in akiec of 0.85. 0.82, respectively, as opposed to their training in akiec without weighted loss function: 0.65 and 0.37.

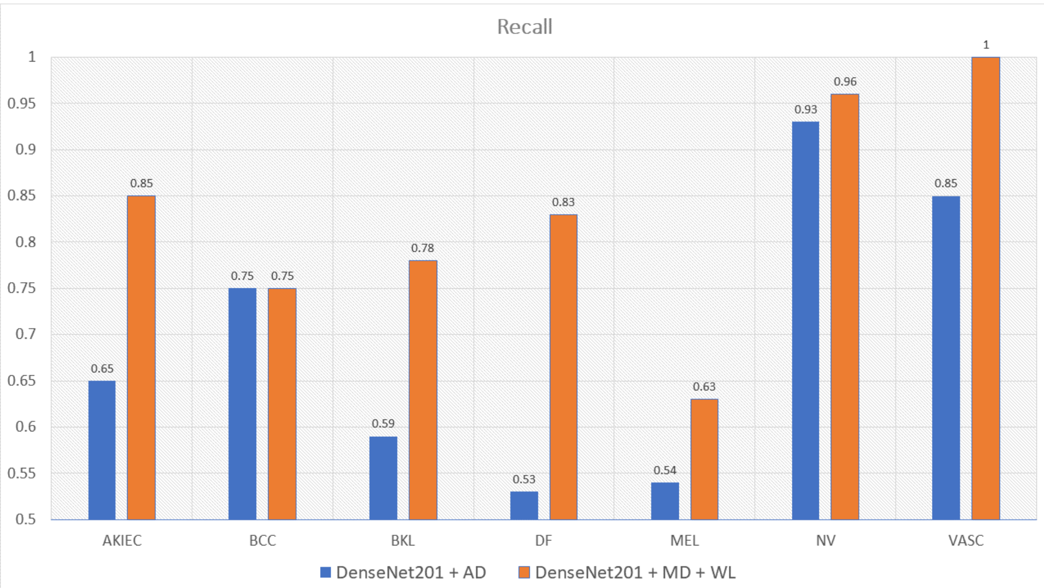

Figure 12. The comparison between recal scores of DenseNet201 trained with augmented data and the one trained with metadata and weight loss

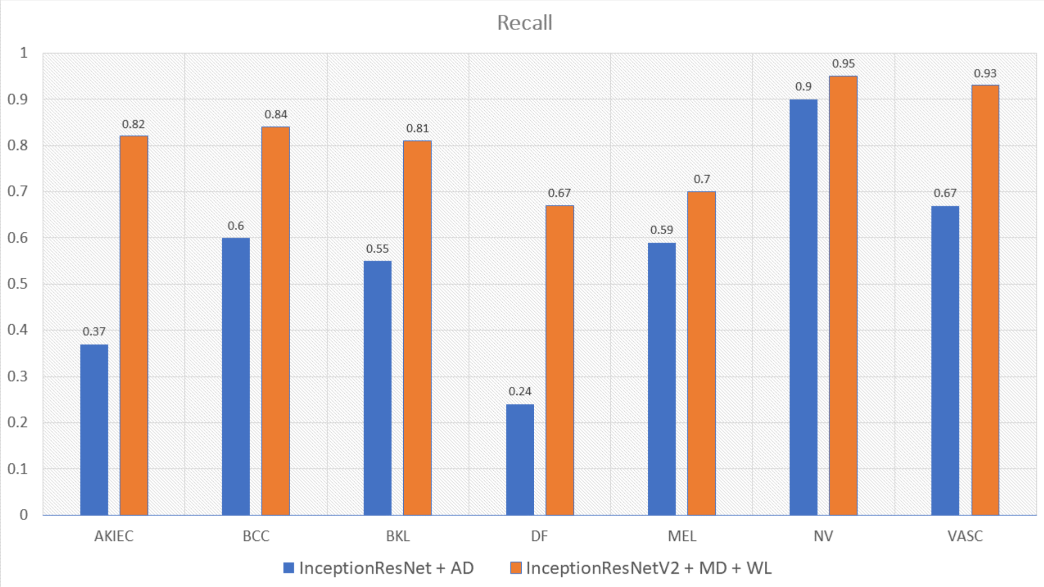

Figure 13. The comparison between recall scores of InceptionResNetV2 trained with augmented data and the one trained with metadata and weight loss

Another interesting point found during the experiment is that MobileNetV2, MobileNetV3 and NasNetMobile have small number of parameters and depth, though have relative good performance. MobileV3large, MobileV3Small, NasNetLarge and NasNetMobile outperform others on classifying class df with the recall score of 0.92, 1, 0.92, 0.92, separately according to the Table A5. It's transparent that MobileNetV3Large and NasNetMobile are the two best performance model. Nevertheless, MobileNetV3Large has less number of parameters and depth than NasNetMobile.

Model	MobileNetV3Large	DenseNet201	InceptionResnetV2
No. Parameters	**5.5M**	20.2M	55.9M
Depth	**118**	402	449
Accuracy	0.86	0.89	0.90
Time Prediction(s/epochs)	**116**	1000	3500

Table 6. How Performance of MobileNetV3Large be optimized

The Table 6 shows that the MobileNetV3Large, though the number of parameters are much smaller than the DenseNet201, InceptionResNetV2, achieve the accuracy nearly to the others. In detail, MobileNetV3Large whose the number of parameters is 5.5 millions which is four and ten times less than DenseNet201 and InceptionResNetV2, respectively. The depth of MobileNetV3Large, on the other hand, is four times less than DenseNet201, InceptionResNetV2 which are 118 hidden layers as opposed to 402, 449 of DenseNet201 and InceptionResNetV2, separately. Although, the MobileNetV3Larege only achieve the accuracy of 0.86 the time need for prediction is 10 and 30 times less than the other opponents. If the MobileNetV3Large need a harder process of parameter hyper-tuning to achieve a better result, which is also the future target of this research.

The Table 7 shows the AUC score of the three models InceptionResNetV2, Densenet201, and ResNet50 which is trained with only augmented data or metadata. It's transparent that the InceptionResNetV2 and DenseNet201 have higher AUC-score trained with metadata: 0.974 and 0.971 as opposed to 0.972 and 0.93, respectively. ResNet50 trained with augmented data, on the other hand have higher auc-score: 0.95 as compared to 0.93 of ResNet50 trained with metadata. Overall, InceptionResNetV2 trained with metadata reach the peak with the auc-score of 0.974. The InceptionResNetV2 trained with metadata is also compared with the others to find out the best models trained. According to the figure

15, the InceptionResNetV2 still hit the peak of 0.974 auc-score. ResNet152, otherwise is the worst model with the auc-score of 0.87. Other models, on the other hand have the approximately same of auc-score.

Model	AUC(AD)	AUC(MD)
InceptionResNetV2	0.971	**0.974**
DenseNet201	0.93	**0.97**
ResNet50	**0.95**	0.93
ResNet152	-	0.87
NasNetLarge	-	**0.96**
MobileNetV2	-	**0.97**
MobileNetV3Small	-	**0.96**
MobileNetV3Large	-	**0.97**
NasNetMobile	-	**0.97**

Table 7. AUC(area under the curve) of all models. AD stands for augmented data, this indicate that the model is trained with augmented data. MD stands for Metadata which indicate that the model is trained with Metadata

Besides the comparison between original weight loss calculated by the sample percentage of each class model and the new weight loss based model is also conducted on the three best performance model including InceptionResNetV2, DenseNet201, MobileNetV3. After the experiment, it is found out that the new weight loss function dose not only contribute to the model to overcome the data imbalance problem but it also make the accuracy increase. The performance of models is described in the Table 8

Model	No Weight	Original Loss Accuracy	New Loss Accuracy
InceptionResNetV2	0.74	0.79	0.90
DenseNet201	0.81	0.84	0.89
MobileNetV3	0.79	0.80	0.86

Table 8. Loss based model accuracy comparison

After reviewing, the InceptionResNetV2 is found to be the best model trained. Futhermore, the InceptionResNetV2 is compared with the other state of the art researched model. According to the Table 9, there are six research that use the same data set: HAM10000 but different approaches. These models used in that research are also SOTA models sorted in ascending order. The Table shows that the accuracy of the combination of InceptionResNetV2 with Soft-Attention, metadata and weight loss in this research is less than the InceptionResNetV2 with Soft-Attention and augmented data: 0.90 compared to 0.93 respectively. However, since Soumyyak et al use data augmentation for all class of an imbalanced data set, the f1-score and recall score are much lower. This is because the model in that research can only classify well on NV and VASC class which have the highest number of samples. On the other hand, the InceptionResNetV2 in this research also outperform the other model according to 5 indicator: accuracy, precision, f1-score, recall score, and auc score.

Model	Accuracy	Precision	f1-score	recall	auc-score
Our Proposed	0.9	0.86	**0.86**	**0.81**	0.974
InceptionResNetV2[14]	0.93	0.89	0.75	0.71	0.97
[3]	-	0.88	0.77	0.74	-
[9]	0.88	-	-	-	-
[12]	0.86	-	-	-	-
GradCam and Kernel SHAP[6]	0.88	-	-	-	-
Student and Teacher[2]	0.85	0.76	0.76	-	-

Table 9. Comparative Analysis

However, there is still some drawbacks of the model that the InceptionResNetV2 cannot well classify the melanoma and the nevus. According to Figure 14 the model sometime classify the black nevus as the melanoma because of the same color between them. However, this problem is not true for the hard black or big melanoma or the red black nevus. Some future approach that can be proposed are change the type of color to other to fix the same color problem.

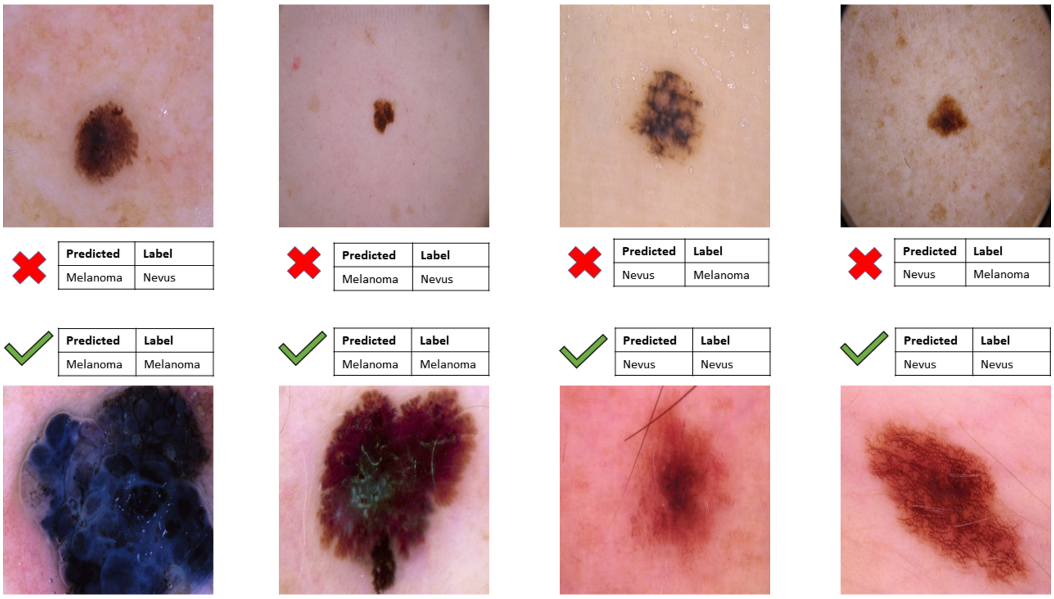

Figure 14. Model ability to classify melanoma and nevus

4. Conclusions

In this research, our proposed is to construct a model as a combination of backbone models and Soft-Attention. Moreover, the model takes two inputs including image data and metadata. besides, a new weight loss function is applied to figure out the data imbalance problem. Finally, the combination of InceptionResNetV2, Soft-Attention, and Metadata is the best model. Although the accuracy and the precision of the model are not the highest, the f1-score, recall, and AUC-score is the highest and the most balanced indicator. Therefore, InceptionResnetV2 can classify well in all classes including low samples classes. Otherwise, during the experiment, the combination of MobileNetV3, Soft-Attention, and Metadata achieve an accuracy that is nearly the same as InceptionResNetV2, though with fewer number parameters and depth. Therefore the infer time is much less than the InceptionResNetV2. This result opens the door to constructing a great performance model that can be applied to mobile, IoT devices.

Author Contributions: Conceptualization, V.D.N. and H.K.D.; methodology, V.D.N. and H.K.D.; software, K.D.; validation, V.D.N., N.D.B. and H.K.D.; formal analysis, V.D.N. and H.K.D.; investigation, V.D.N., N.D.B. and H.K.D.; resources, V.D.N.; data curation, H.K.D.; writing—original draft preparation, H.K.D.; writing—review and editing, V.D.N. and N.D.B.; visualization, H.K.D.; supervision, V.D.N. and N.D.B.; project administration, V.D.N. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding

Institutional Review Board Statement: Not applicable

Informed Consent Statement: Not applicable

Data Availability Statement: The code and the data analysis report can be found here: https://github.com/KhoiDOO/Skin-Disease-Detection-HAM100000.git

Acknowledgments:

Conflicts of Interest: The authors declare no conflict of interest.

Sample Availability:

Abbreviations

The following abbreviations are used in this manuscript:

CAD	Computer aided diagnosis
AI	Artificial Intelligence
AKIEC	Actinic keratoses and intraepithelial carcinoma or Bowen's disease
BCC	Basal Cell Carcinoma
BKL	Benign Keratosis-like Lesions
DF	Dermatofibroma
MEL	Melanoma
NV	Melanocytic Nevi
VASC	Vascular Lesions
HISTO	Histopathology
FOLLOWUP	Follow-up examination
CONSENSUS	Expert Consensus
CONFOCAL	Confocal Microscopy
RGB	Red Green Blue
BGR	Blue Green Red
TP	True Positives
FN	False Negatives
TN	True Negatives
FP	False Positives
Sens	Sensitivity
Spec	Specificity
AUC	Area Under the Curve
ROC	Receiver Operating Curve

Appendix A Detailed Model Structure

DenseNet-201	**DenseNet-201 + SA**	**Inception-ResNetV2**	**Inception-ResNetV2 + SA**	**ResNet-50**	**ResNet-50 + SA**	**ResNet-152**	**ResNet-152 + SA**	**NasNet-Large**	**NasNet-Large + SA**
Conv2D 7x7	Conv2D 7x7	STEM	STEM	Conv2D 7x7	Conv2D 7x7	Conv2D 7x7	Conv2D 7x7	Conv2D 3x3	Conv2D 3x3
Pooling 3x3	Pooling 3x3			Pooling 3x3	Pooling 3x3	Pooling 3x3	Pooling 3x3	Pooling	Pooling
DenseBlock x 6	DenseBlock x 6	Inception ResNet A x 10	Inception ResNet A x 10	Residual Block x 3	Residual Block x 3	Residual Block x 3	Residual Block x 3	Reduction Cell x 2	Reduction Cell x 2
Conv2D 1x1	Conv2D 1x1	Reduction A	Reduction A					Normal Cell x N	Normal Cell x N
Average pool 2x2	Average pool 2x2								
DenseBlock x 12	DenseBlock x 12	Inception ResNet B x 20	Inception ResNet B x 20	Residual Block x 4	Residual Block x 4	Residual Block x 8	Residual Block x 8	Reduction Cell	Reduction Cell
Conv2D 1x1	Conv2D 1x1	Reduction B	Reduction B					Normal Cell x N	Normal Cell x N
Average pool 2x2	Average pool 2x2								
DenseBlock x 48	DenseBlock x 12	Inception ResNet C x 5	Inception ResNet C x 5	Residual Block x 6	Residual Block x 6	Residual Block x 36	Residual Block x 36	Reduction Cell	Reduction Cell
Conv2D 1x1	Conv2D 1x1							Normal Cell x N	Normal Cell x N-2
Average pool 2x2	Average pool 2x2								
DenseBlock x 29	DenseBlock x 29			Residual Block x 3		Residual Block x 3			
DenseBlock x 3	**SA Module**		**SA Module**		**SA Module**		**SA Module**		**SA Module**
GAP 7x7		Average pool		GAP 7x7		GAP 7x7			
FC 1000D		Dropout (0.8)		FC 1000D		FC 1000D			
SoftMax	SoftMax	SoftMax	SoftMax	SoftMax	SoftMax	SoftMax	SoftMax	SoftMax	SoftMax

Table A1. Details Structure of Models except Mobile models. SA stands for Soft-Attention, SA Module denotes whether that model use Soft-Attention Module. GAP stands for Global Average Pooling. FC stands for Fully-Connected Layer

Appendix B Detailed Mobile-based Model Structure

MobileNetV2	**MobileNetV2 + SA**	**MobileNetV3 Small**	**MobileNetV3 Small + SA**	**MobileNetV3 Large**	**MobileNetV3 Large + SA**	**NasNet Mobile**	**NasNetMobile + SA**
Conv2D	Conv2D	Conv2D 3x3	Conv2D 3x3	Conv2D 3x3	Conv2D 3x3	Normal Cell	Normal Cell
bottleneck	bottleneck	bottleneck 3x3 SE	bottleneck 3x3 SE	bottleneck 3x3 3 repeated	bottleneck 3x3 3 repeated	Reduction Cell	Reduction Cell
bottleneck 2 repeated	bottleneck 2 repeated	bottleneck 3x3	bottleneck 3x3	bottleneck 5x5 SE 3 repeated	bottleneck 5x5 SE 3 repeated	Normal Cell	Normal Cell
bottleneck 3 repeated	bottleneck 3 repeated	bottleneck 5x5 SE 8 repeated	bottleneck 5x5 SE 8 repeated	bottleneck 3x3 4 repeated	bottleneck 3x3 4 repeated	Reduction Cell	Reduction Cell
bottleneck 4 repeated	bottleneck 4 repeated			bottleneck 3x3 SE 2 repeated	bottleneck 3x3 SE 2 repeated	Normal Cell	
bottleneck 3 repeated	bottleneck 3 repeated			bottleneck 5x5 SE 3 repeated	bottleneck 5x5 SE 3 repeated		
bottleneck 3 repeated	bottleneck						
bottleneck							
Conv2D 1x1		Conv2D 1x1 SE	Conv2D 1x1 SE	Conv2D 1x1	Conv2D 1x1		
AP 7x7		Pool 7x7	Pool 7x7	Pool 7x7	Pool 7x7		
Conv2D 1x1	**SA Module**	Conv2D 1x1 2 repeated	**SA Module**	Conv2D 1x1 2 repeated	**SA Module**		**SA Module**
Softmax	Softmax	Softmax	Softmax	Softmax	Softmax	Softmax	Softmax

Table A2. Details Structure of Mobile based Models. SA stands for Soft-Attention, SA Module denotes whether that model use Soft-Attention Module. SE which stands for Squeeze-And-Excite shows whether that block has Squeeze-And-Excite.

Appendix C Detailed Model Performance

Appendix C.1 F1-Score Model Performance

Model	akiec	bcc	bkl	df	mel	nv	vasc	Mean
DenseNet201 with Augmented Data	0.56	0.75	0.64	0.62	0.60	0.93	0.85	0.70
InceptionResNetV2 with Augmented Data	0.42	0.63	0.51	0.35	0.57	0.9	0.7	0.58
Resnet50 with Augmented Data	0.39	0.59	0.42	0.6	0.42	0.88	0.79	0.58
VGG16 with Augmented Data	0.35	0.62	0.42	0.32	0.47	0.89	0.77	0.54
DenseNet201 with Metadata and WeightLoss	**0.84**	0.77	**0.81**	**0.83**	**0.69**	0.94	0.97	**0.83**
InceptionResNetV2 with Metadata and WeightLoss	0.77	0.83	**0.83**	0.64	**0.75**	0.94	0.7	**0.81**
Resnet50 with Metadata and WeightLoss	0.49	0.59	0.55	0.36	0.45	0.83	0.8	0.58
Resnet152 with Metadata and WeightLoss	0.42	0.38	0.41	0.15	0.4	0.75	0.75	0.46
NasNetLarge with Metadata and WeightLoss	0.79	0.79	0.8	0.74	0.65	0.92	0.92	**0.80**
MobileNetV2 with Metadata and WeightLoss	0.68	0.79	0.66	0.78	0.54	0.9	**0.9**	0.75
MobileNetV3Large with Metadata and WeightLoss	0.72	0.76	0.75	0.92	0.58	0.92	**0.92**	0.79
MobileNetV3Small with Metadata and WeightLoss	0.6	0.72	0.61	0.75	0.47	0.89	**0.89**	0.70
NasNetMobile with Metadata and WeightLoss	0.76	0.74	0.78	0.73	0.63	0.93	**0.93**	0.78

Table A3. F1-Score of each class: akiec, bcc, bkl, df, mel, nv, vasc which are denoted in the abbreviation. The last column is the expected value of f1-score from each model. All model in the first column is the models trained in this research. The term "with Augmented Data" means that model is trained with data augmenting during the training, there is no metadata or weight loss contribution. The term "with Metadata and WeightLoss" means that the model is trained with metadata including age, gender, localization and the weight loss function, there is no augmented data contribution

Appendix C.2 Recall Model Performance 354

Model	akiec	bcc	bkl	df	mel	nv	vasc	Mean
DenseNet201 with Augmented Data	0.65	0.75	0.59	0.53	0.54	0.93	0.85	0.69
InceptionResNetV2 with Augmented Data	0.37	0.60	0.55	0.24	0.59	0.9	0.67	0.56
Resnet50 with Augmented Data	0.33	0.56	0.38	0.53	0.40	0.92	0.81	0.56
VGG16 with Augmented Data	0.31	0.66	0.37	0.24	0.40	0.94	0.71	0.51
DenseNet201 with Metadata and WeightLoss	**0.85**	0.75	0.78	0.83	0.63	0.96	**1**	**0.82**
InceptionResNetV2 with Metadata and WeightLoss	**0.82**	0.84	0.81	0.67	0.7	0.95	0.93	**0.81**
Resnet50 with Metadata and WeightLoss	0.67	0.63	0.54	0.83	0.63	0.74	0.86	0.70
Resnet152 with Metadata and WeightLoss	0.51	0.49	0.35	0.76	0.47	0.63	0.48	0.52
NasNetLarge with Metadata and WeightLoss	0.73	0.71	**0.83**	**0.92**	0.59	0.9	0.93	**0.81**
MobileNetV2 with Metadata and WeightLoss	0.7	0.86	0.72	0.75	0.58	0.86	**1**	0.78
MobileNetV3Large with Metadata and WeightLoss	0.72	0.76	0.75	**0.92**	0.58	0.92	0.92	**0.80**
MobileNetV3Small with Metadata and WeightLoss	0.76	0.84	0.68	**1**	0.52	0.82	0.93	0.79
NasNetMobile with Metadata and WeightLoss	**0.82**	0.73	**0.83**	**0.92**	0.53	0.93	0.93	**0.81**

Table A4. Recall score of each class and the expected value of recall score from each model

Appendix C.3 Detailed Mobile Model Perform 355

Model	[21]	[22]Small	[22]Large	[33]Mobile
Accuracy(avg)	0.81	0.78	0.86	**0.86**
Balanced Accuracy(avg)	0.86	0.87	0.87	**0.88**
Precision(avg)	0.71	0.63	**0.75**	0.73
F1-score(avg)	0.75	0.70	**0.79**	0.78
Sensitivity(avg)	0.78	0.79	0.80	**0.81**
Specificity(avg)	0.95	0.95	0.95	**0.96**
ROC-AUC-score(avg)	0.96	0.95	0.96	**0.97**

Table A5. Deeper analyzing of mobile model. This table illustrate the other indicators of the four mobile-based models including MobileNetV2, MobileNetV3Small, MobileNetV3Large, NasNetMobile. The indicators are Accuracy, Balanced Accuracy, Precision, F1-score, Sensitivity, Specificity, and ROC - AUC score. All of them are average indicator

References 356

1. Katherine M. Li and Evelyn C. Li. Skin Lesion Analysis Towards Melanoma Detection via End-to-end Deep Learning of Convolutional Neural Networks. *Sensors* **2018**. 357 358
2. Xiaohan Xing and Yuenan Hou and Hang Li and Yixuan Yuan and Hongsheng Li and Max Q.-H. Meng Categorical Relation-Preserving Contrastive Knowledge Distillation for Medical Image Classification. *Springer Link* **2021**. 359 360
3. Rishu Garg and Saumil Maheshwari and Anupam Shukla Decision Support System for Detection and Classification of Skin Cancer using CNN. *Springer Link* **2019**. 361 362
4. Xuan Li and Yuchen Lu and Christian Desrosiers and Xue Liu Out-of-Distribution Detection for Skin Lesion Images with Deep Isolation Forest. *Springer Link* **2020**. 363 364

5. Amirreza Rezvantalab and Habib Safigholi and Somayeh Karimijeshni Dermatologist Level Dermoscopy Skin Cancer Classification Using Different Deep Learning Convolutional Neural Networks Algorithms. *Arxiv* **2021**.
6. Kyle Young and Gareth Booth and Becks Simpson and Reuben Dutton and Sally Shrapnel Dermatologist Level Dermoscopy Deep neural network or dermatologist?. *Nature* **2021**.
7. Philipp Tschandl and Cliff Rosendahl and Harald Kittler The HAM10000 data set, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Nature* **2018**.
8. Michele Alberti and Angela Botros and Narayan Schuez and Rolf Ingold and Marcus Liwicki and Mathias Seuret Trainable Spectrally Initializable Matrix Transformations in Convolutional Neural Networks. *IEEE Xplore* **2019**.
9. Hemanth Nadipineni Method to Classify Skin Lesions using Dermoscopic images. *Arxiv* **2020**.
10. Nils Gessert and Maximilian Nielsen and Mohsin Shaikh and René Werner and Alexander Schlaefer Skin Lesion Classification Using Ensembles of Multi-Resolution EfficientNets with Meta Data. *Arxiv* **2020**.
11. Pranav Poduval and Hrushikesh Loya and Amit Sethi Functional Space Variational Inference for Uncertainty Estimation in Computer Aided Diagnosis. *Arxiv* **2020**.
12. Peng Yao and Shuwei Shen, Mengjuan Xu and Peng Liu and Fan Zhang and Jinyu Xing and Pengfei Shao and Benjamin Kaffenberger and Ronald X. Xu Single Model Deep Learning on Imbalanced Small Datasets for Skin Lesion Classification. *Arxiv* **2022**.
13. Manu Goyal and Thomas Knackstedt and Shaofeng Yan and Saeed Hassanpour Artificial Intelligence-Based Image Classification for Diagnosis of Skin Cancer: Challenges and Opportunities. *Arxiv* **2020**.
14. Soumyya Kanti Datta and Mohammad Abuzar Shaikh and Sargur N. Srihari and Mingchen Gao Soft-Attention Improves Skin Cancer Classification Performance. *SpringerLink* **2021**.
15. Amirreza Mahbod and Philipp Tschandl and Georg Langs and Rupert Ecker and Isabella Ellinger The Effects of Skin Lesion Segmentation on the Performance of Dermatoscopic Image Classification *Arxiv* **2020**.
16. Yeong Chan Lee and Sang-Hyuk Jung and Hong-Hee Won WonDerM: Skin Lesion Classification with Fine-tuned Neural Networks *Arxiv* **2019**.
17. Gao Huang and Zhuang Liu and Laurens van der Maaten and Kilian Q. Weinberger: Densely Connected Convolutional Network *IEEE Xplore* **2018**.
18. Mingxing Tan and Quoc V. Le EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks *Arxiv* **2020**.
19. Christian Szegedy and Vincent Vanhoucke and Sergey Ioffe and Jonathon Shlens and Zbigniew Wojna Rethinking the Inception Architecture for Computer Vision *IEEE Xplore* **2015**.
20. Andrew G. Howard and Menglong Zhu and Bo Chen and Dmitry Kalenichenko and Weijun Wang and Tobias Weyand and Marco Andreetto and Hartwig Adam MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications *Arxiv* **2017**.
21. Mark Sandler and Andrew Howard and Menglong Zhu and Andrey Zhmoginov and Liang-Chieh Chen MobileNetV2: Inverted Residuals and Linear Bottlenecks *IEEE Xplore* **2018**.
22. Andrew Howard and Mark Sandler and Grace Chu and Liang-Chieh Chen and Bo Chen and Mingxing Tan and Weijun Wang and Yukun Zhu and Ruoming Pang and Vijay Vasudevan and Quoc V. Le and Hartwig Adam Searching for MobileNetV3 *IEEE Xplore* **2019**.
23. Kaiming He and Xiangyu Zhang and Shaoqing Ren and Jian Sun Deep Residual Learning for Image Recognition *IEEE Xplore* **2015**.
24. Kaiming He and Xiangyu Zhang and Shaoqing Ren and Jian Sun Identity Mappings in Deep Residual Networks *Springer Link* **2016**.
25. Karen Simonyan and Andrew Zisserman Very Deep Convolutional Networks for Large-Scale Image Recognition *Arxiv* **2016**.
31. François Chollet Xception: Deep Learning with Depthwise Separable Convolutions *IEEE Xplore* **2017**.
27. Diederik P. Kingma, Jimmy Ba Adam: A Method for Stochastic Optimization *Arxiv* **2017**.
28. Kelvin Xu and Jimmy Ba and Ryan Kiros and Kyunghyun Cho and Aaron Courville and Ruslan Salakhutdinov and Richard Zemel and Yoshua Bengio Show, Attend and Tell: Neural Image Caption Generation with Visual Attention 2020 17th International Conference on Frontiers in Handwriting Recognition *PMLR* **2016**.
29. Mohammad Abuzar Shaikh and Tiehang Duan and Mihir Chauhan and Sargur N. Srihari. 2020 17th International Conference on Frontiers in Handwriting Recognition *IEEE Xplore* **2020**.
30. Naofumi Tomita and Behnaz Abdollahi and Jason Wei and Bing Ren and Arief Suriawinata and Saeed Hassanpour. Attention-Based Deep Neural Networks for Detection of Cancerous and Precancerous Esophagus Tissue on Histopathological Slides *Jama Network* **2020**.
31. Mohammad Abuzar Shaikh and Tiehang Duan and Mihir Chauhan and Sargur N. Srihari. 2020 17th International Conference on Frontiers in Handwriting Recognition *IEEE Xplore* **2019**.
32. Srihari. YAOSHIANG HO AND SAMUEL WOOKEY The Real-World-Weight Cross-Entropy Loss Function: Modeling the Costs of Mislabeling *IEEE Xplore* **2020**.
33. Barret Zoph, Vijay Vasudevan, Jonathon Shlens, Quoc V. Le Learning Transferable Architectures for Scalable Image Recognition *IEEE Xplore* **2017**.
34. Abien Fred Agarap Deep Learning using Rectified Linear Units (ReLU) *Arxiv* **2019**.
35. Jie Hu, Li Shen, Samuel Albanie, Gang Sun, Enhua Wu Squeeze-and-Excitation Networks *IEEE Xplore* **2019**.

36. Terrance DeVries, Graham W. Taylor Improved Regularization of Convolutional Neural Networks with Cutout *Arxiv* **2017**.
37. Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, Alex Alemi Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning *AAAI Conference* **2018**.