Melanoma Classification Using Deep Learning

Ramiz Akhtar
University of Virginia
409 13th St. NW
Charlottesville, VA 22903
rsa5wj@virginia.edu

Rayaan Faruqi
University of Virginia
1815 Jefferson Park Avenue
Charlottesville, VA 22903
raf9dz@virginia.edu

Kunaal Sarnaik
University of Virginia
43288 John Danforth Court
Ashburn, VA 20147
kss7yy@virginia.edu

Motivation

According to the Center for Disease Control (CDC), melanoma is the most serious and deadly type of skin cancer.[1] The incidence and mortality of invasive melanoma in the United States has risen steadily, with the lifetime risk of developing the disease skyrocketing from 0.03% in 1930 to 1.82% in 2017.[2] Furthermore, from 2012 to 2016, approximately 77,698 new cases of melanoma occurred in the United States each year; previous studies have also found that melanoma is more likely to metastasize relative to other types of skin cancers.[3] Given that the global incidence and mortality of melanoma are expected to continue rising at an alarming rate, diagnosing and treating the disease early and efficiently is an urgent priority.[4] However, as can be observed in **Figure 1** below, distinguishing melanoma not only from other types of skin diseases, but also from benign skin moles (i.e. nevi), at the dermatologist-level is an arduous process that must account for various factors.

Figure 1: Depicted are three skin lesions, the similarities of which make it difficult to diagnose and treat melanoma early in its progression. **a.)** Skin lesion positive for melanoma. **b.)** Benign nevus negative for melanoma. **c.)** Skin lesion not positive for melanoma, but positive for seborrheic keratosis.

Compounded by the clinical complexities involved when determining the specific stage of a tumor, this difficulty in diagnosing melanoma results in many patients not noticing a potentially fatal tumor appearing on their skin.[5] Moreover, the current process for diagnosis, which involves presenting to a series of doctors, can not only be expensive but also intimidating for patients. Considering the social, economic, and political determinants of healthcare that additionally belabor the process, the patient may become discouraged and delay diagnosis until a later date. Worse yet, they may delay seeking diagnosis until experiencing an adverse outcome as a result of their endorsed malignancy. Consequently, patients who discover they have invasive melanoma at a later stage often experience worse clinical and financial outcomes, including metastasis, poor prognosis, depression, and higher insurance costs.[6]

Thus, providing diagnostic information for a potentially fatal melanoma tumor via a smartphone application through utilization of machine learning principles may make early detection more feasible,

accessible, and economical by way of reduced treatment costs. Furthermore, such an application may also lead to improved clinical outcomes when it comes to the metastasis and prognosis of an invasive melanoma tumor. Ultimately, by making the first step for diagnosis more convenient, effective, equitable, and accessible, clinical outcomes for patients may improve substantially.

Background

Melanoma arises from genetic mutations in melanocytes, which are the upper layer cells that give skin its pigmentation.[7,8] Exposure to ultraviolet (UV) radiation from the sun causes greater production of melanin by the melanocytes, effectively darkening the skin to protect it from the UV radiation.[8] In excess, however, UV radiation triggers mutations in the melanocytes, thereby causing uncontrollable cell growth.[8] Melanomas may also be hidden and develop in areas that are not exposed to UV light such as between fingers and on the scalp. Therefore, other unknown factors beyond UV light may also contribute to an individual's risk of developing melanoma.[1]

Considering that melanoma is the deadliest type of skin cancer, early diagnosis becomes paramount to a positive clinical outcome. However, melanoma is difficult to diagnose because it can develop anywhere on the human body.[1] Melanoma is often found in the form of a mole but can also occur in normal-looking skin, emphasizing that it can progress undetected. Intuitively, melanoma in moles is comparatively easier to detect than in normal-looking skin.[1] Initial symptoms of melanoma in moles include asymmetric shape, irregular border, uneven distribution of color, and time-based evolution of any of the aforementioned characteristics.[1]

Finally, there are several clinical, financial, and psychosocial implications of an untimely melanoma diagnosis that can result in detrimental consequences for the patient. For instance, similar to any other type of cancer, the progression of invasive melanoma is much more difficult to manage clinically in latter stages as the likelihood of metastasis and tumor recurrence increases substantially.[9] Furthermore, a late diagnosis can result in severe financial distress for the patient due to the sheer urgency, unpredictability, and expense associated with emergent treatment procedures.[10] Additionally, there are severe mental and psychiatric factors that toll patients while undergoing palliative care treatment associated with melanoma; these include substance abuse, depression, and anxiety.[11]

Related Work

The utilization of deep learning in health care, with applications including imaging diagnosis, digital pathology, prediction of hospital admission, drug design, and others, has been an emerging area of study.[12] With recent leaps in computational power and rapid advancements in artificial intelligence, many investigators have leveraged deep learning techniques to help mitigate one of the most prevalent diseases in the word: cancer.[13] Specifically, a hot topic has been classifying images of tumors using machine learning as a method of delivering efficient, timely, and accurate diagnoses that can decrease the number of adverse outcomes, treatments costs, and downstream complications commonly associated with cancer progression.[14]

The most substantial amount of work within this area in recent years has been breast cancer. In June, 2020, Sichuan and Yibin University researchers Lui et al. implemented a novel classification method to classify pathological images of breast cancer through utilization of deep convolutional neural networks (CNNs), DeepBC integrated Inception, ResNet, AlexNet, and feature extraction.[15] They achieved 92% and 96.43% accuracy in classifying patients and images, respectively, for both benign and malignant tumors across several types commonly associated with breast cancer, such as ductal and lobular carcinomas. Their

findings suggest robustness and generalizability of their method compared to the state-of-the-art, oncologist-level classification of breast cancer tumors, further reporting positive downstream outcomes which include increased diagnosis efficiency, reduction in pathologist workload, and greater aversion to the possibility of misdiagnosis.

Methods of classifying skin cancer tumors through utilization of deep learning have also been implemented; however, developments in this area are much more primitive. Esteva et al. at Stanford University demonstrated a dermatologist-level classification of skin cancer using deep CNNs trained on a dataset of 129,450 clinical images consisting of 2,032 different diseases.[16] However, the Stanford team reported several shortcomings in their findings, primarily noting a lack of analysis regarding the sensitivity and specificity of their design which can result in downstream socioeconomic complications such as mass hysteria from a high false-positive rate.

The work discussed in this proposal intends to build off of previous skin disease assessment models such as that of Esteva et al., yet focus the effort more towards optimizing sensitivity, specificity, and generalizability of the resulting deep CNN model.[17–19]

Claim / Target Task

The overarching goal of this project is to utilize a supervised multi-class classification algorithm to assess the properties of an image containing indications to either 1) melanoma, 2) seborrheic keratosis, or 3) non-melanoma skin (i.e. nevi). An 11-gigabyte kaggle dataset provided by user Pablo Lopez Santori that consists of labeled 2,750 images across these three classifications will be used for training, validation, and testing.[20]

The primary focus of this classification algorithm will be to minimize the false-negative rate as false negatives can result in severe consequences for a patient including treatment costs, adverse outcomes, and exacerbations of psychosocial symptoms. Secondarily, there will be an attempt to minimize the false-positive rate to decrease potential complications including mass hysteria and individual stress. If successful, a novel CNN will be created to provide a computational-based diagnosis that will be less costly relative to traditional diagnoses through a healthcare provider with regards to both time and money (**Figure 2**).

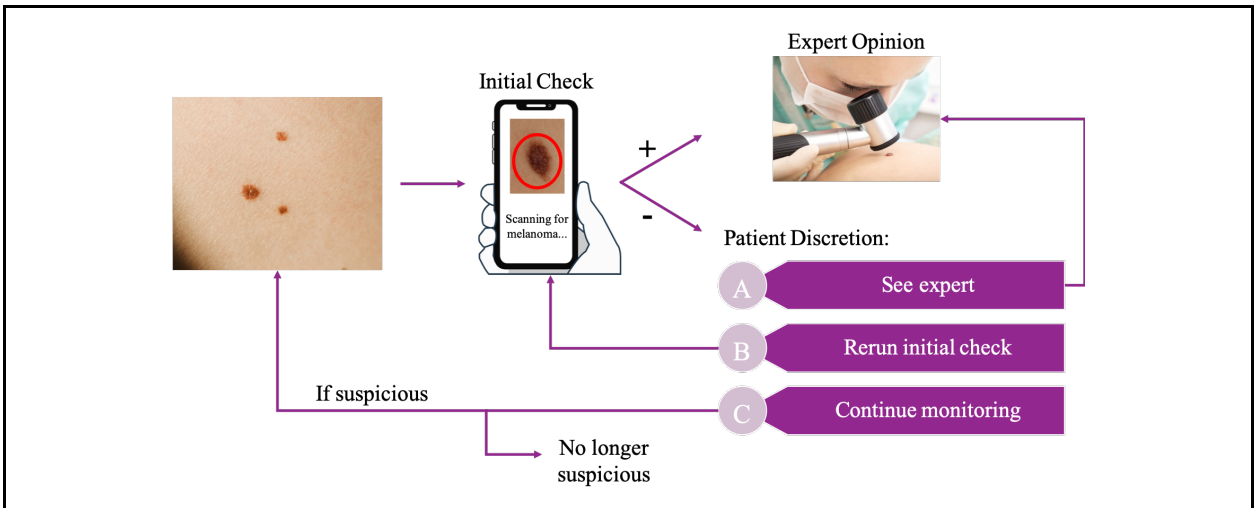


Figure 2: Depicted is an intuitive schematic showcasing a potential future timeline of diagnosis for melanoma and other skin conditions using models such as the one proposed. Patients will have the ability to scan their skin lesion of interest with the mobile application through utilization of the deep convolutional neural network and then choose

to act on the pre-diagnosis based on the possible options outlined above. Patients may choose inexpensive, in-home checks utilizing image classification models such as the one proposed. Depending on the results, patients may choose to seek an expert's opinion or make a clinical decision at their own discretion. Furthermore, as precision medicine procedures develop, future medical diagnoses may involve machine learning algorithms acting as cursory opinions and pre-diagnoses. Both cases necessitate that such algorithms and classification models are made as error-free as possible; an important indicator of an error-free model includes optimizing the model's specificity. In other words, robustness and usability of the model in real-world scenarios heavily depends on minimizing the number of false negatives. The threshold in the proposed model will lean edge cases towards a positive diagnosis for either melanoma or seborrheic keratosis, with the justification being that false positives are typically less harmful than false negatives in regard to long-term health outcomes.

Proposed Solution

We propose using TensorFlow with the Keras API in Python to create a deep CNN based off of the state-of-the-art DenseNet image classification architecture. In recent years, machine learning experts have consistently observed that deep CNN models can be made substantially deeper, more accurate, and more efficient to train if they contain shorter connections between the layers close to the input and those close to the output.[21] Thus, DenseNet, developed by Cornell University researchers Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Weinberger, was constructed to embrace this observation and connect each layer in a given CNN to every other layer in feed-forward fashion. Traditionally, convolutional neural networks with L layers only have L connections. However, the total direct connections between the layers possessed by a DenseNet model is given by **Equation 1** below:

$$Number\ of\ Connections = \frac{L * (L+2)}{2} \text{ [Eq. 1]}$$

According to the study conducted by Huang et al., a DenseNet possesses numerous advantages over a traditional CNN; these benefits include an avoidance to the vanishing-gradient problem, strengthening of feature propagation between layers, and a substantial reduction in the number of parameters required for training the model. Furthermore, previous work has supported the claim that DenseNets typically perform better on relatively small and undersampled datasets, which is why we felt this specific CNN architecture was suitable for the analysis described below. For further details on the specific DenseNet variation we utilized in our analysis, see the **Implementation** section below.

In order to minimize the possibility of overfitting in our model, we will also perform data augmentation to artificially boost the image dataset's diversity and "warp" image dimensions. We believe this will be well suited for skin-disease classification, as melanomas and other skin disorders may come in varying shapes and sizes with especially aberrant borders, complex pathoanatomies, and variant shades of color. Effectively, data augmentation will allow the DenseNet CNN to reduce dependence on size of the skin lesion in any given image while still accounting for aberrant border patterns, thus helping classify images that originate from a variety of sources and allowing for increased flexibility in image capture distance from the subject. With regard to the specific implementation of the image augmentation procedures described above, we will leverage the ImageDataGenerator class in the Keras API, which randomly alters (via scaling, rotation, zooming, etc.) the training images.

In order to maximize the specificity of our model, we will iteratively alter the layers of the deep convolutional neural network outlined previously and investigate which model will minimize the false-negative rate. This, in addition to achieving an overall accuracy of greater than 90%, will be the primary endpoint that we intend to achieve in our proposed solution. Moreover, we will also attempt to iteratively

alter the layers so that the sensitivity is maximized, until doing so will achieve a suboptimal model specificity. Both of these approaches will be carried out through careful investigation of any given model's threshold, or in other words, the tolerance utilized to classify an image as positive or negative.

The proposed solution will hopefully be a highly robust and generalizable deep CNN that can eventually be implemented in a mobile operating system, such as iOS or Android, to classify skin lesion images as melanoma positive skin, seborrheic keratosis positive skin, or non-melanoma, non-seborrheic keratosis nevi. Another endpoint that we hope to achieve with this proposed solution is transferability to other skin diseases for image classification, such as those of eczema, dermatitis, and psoriasis, with the intended goal of opening future avenues of investigation and research related to efficient, accurate, and low-cost dermatologist-level skin disease classification through utilization of deep CNNs.

Contributions

We believe our project contributed three things to the landscape of machine learning:

1) The use of a DenseNet for skin lesion classification on a non-optimal dataset.
2) A comparison between a cutting-edge DenseNet model versus a traditional CNN model in the context of classifying skin lesions.
3) The tuning of DenseNet architecture and hyperparameters to advance current research on medical skin lesion image classification.

We contend that the dataset was non-optimal for several reasons. First, the dataset was relatively small, possessing only 2,750 total images (11 GB) relative to other image classification projects which often operate on much larger datasets (i.e., on the order of terabytes). Second, not only was this dataset small, it was also heavily imbalanced. Even if the melanoma and seborrheic keratosis samples were combined, the nevus samples still outnumbered them by over a factor of two. Lastly, the dataset had to be substantially downsampled in order to run on Google's Colab platform, which restricted RAM and GPU resources. Needing to downsample the data is fundamentally a problem with computing power and not with the dataset itself. The dataset was downsampled from a median image size of 3008x2000 $pixels^2$ (LxW) to 128x128 $pixels^2$.

Utilizing DenseNet under these suboptimal conditions allowed us to demonstrate its effectiveness relative to a more traditional CNN model. Although the CNN model was technically higher in accuracy than the DenseNet, its massive overfitting to nevus data, which comprised the vast majority of the dataset, is responsible for the excessively high accuracy as discussed further in the **Experimental Analysis** section of this report. Overall, we demonstrated that a DenseNet model outperforms the traditional CNN in terms of generalizability when dealing with highly skewed training data.

We hope that our experimentation with the DenseNet's hyperparameters and architecture provide insight into future optimization cases for use in similar classification projects. Additionally, we hope that the DenseNet's performance here will inspire other researchers working on biomedical image classification projects to consider the DenseNet as a viable and practical solution in their work.

Implementation

Preprocessing

In order to attain a model with high accuracy, the RGB skin lesion input data required sufficient preprocessing. As previously mentioned, all images were resized to 128x128 $pixels^2$ to meet computational

power constraints. Image pixels in all images were then normalized to a floating point number between zero and one to ensure all pixel values were on the same scale without distorting the data.[22] Both of the previous pre-processing steps were done using the OpenCV library. The last preprocessing step was done using the Keras ImageDataGenerator function which automatically scales, flips (horizontally and vertically), rotates, randomly zooms, and modifies brightness on images within the dataset.[23] Performing the aforementioned augmentations is critical because medical imaging datasets, such as the skin lesion dataset of focus, are small in size, meaning each image in the training dataset holds more importance compared to each image in a larger training dataset. In addition, neural networks are not designed to be rotation or scale invariant.[24] Thus, the images must be augmented through random rotation, flipping, and scaling to ensure that the neural network learns the anatomical differences between nevus, melanoma, and seborrheic keratosis independent of image capture distance, angle, and perspective.

Model Implementation

A DenseNet architecture consists of three distinct sections: dense blocks, convolutional layers, and transition blocks. Dense blocks consist of convolution layers succeeded by a transition layer to connect to the next dense block. Each layer in a dense block is directly connected to each subsequent layer (as seen in **Equation 1** above). Dense blocks represent the cornerstone of the DenseNet architecture because they are responsible for down-sampling layers to reduce the size of feature maps. As depicted in **Figure 3**, the DenseNet architecture consisted of six dense blocks each containing four convolutional layers. Convolutional layers, which reside within dense blocks, each consist of three consecutive operations: a) batch normalization, b) rectified linear unit (ReLU), c) 3x3 convolution. Dropout layers were also added to randomly drop 20% of the extracted features to increase the generalizability of the DenseNet architectural system. Transition blocks perform batch normalization followed by ReLU and a 1x1 convolution instead of the previously mentioned 3x3 convolution in the convolutional layers. The 1x1 convolution acts as a bottleneck layer to reduce the number of input feature maps and improve computational efficiency.

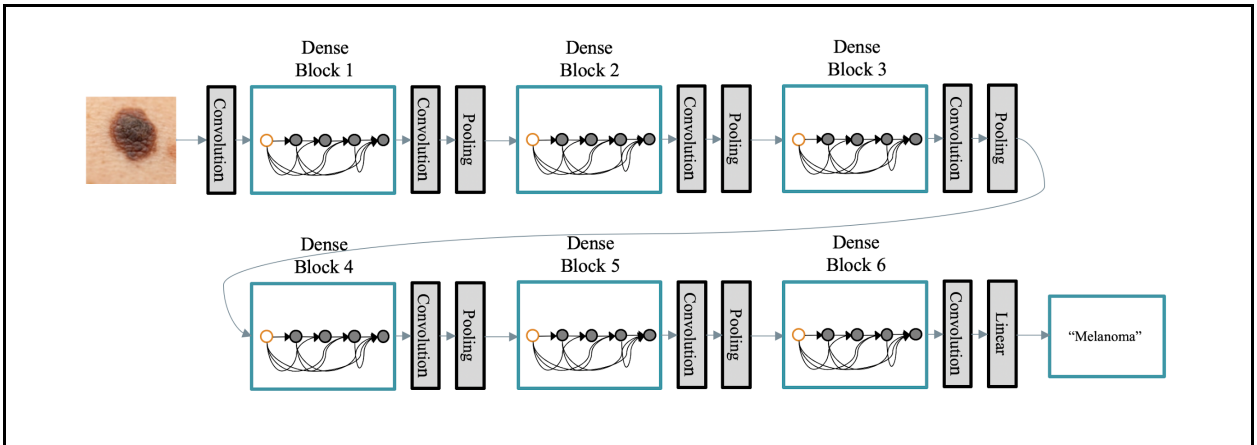


Figure 3: Depicted is the best DenseNet architecture used, consisting of 6 dense blocks, each possessing 4 convolutional layers. Orange circles represent the input convolutional layer for each dense block, and the convolutional and pooling layers immediately after dense blocks are referred to as transition layers. The final linear layer prior to ultimate output is used for converting the convolutional output into an appropriate format for categorical classification.

Hyperparameter Tuning & Selecting the Best Model

The DenseNet CNN model was created using the *create_and_train_cnn()* function. It takes in the x_train, y_train, x_val, and y_val preprocessed image numpy arrays as well as a dictionary of hyperparameters to create a specified DenseNet CNN model. The model is compiled using the Keras API with the 'categorical_crossentropy' loss function that is used when there are two or more label classes. The *fit()* function in the Keras API trains the model while repeatedly validating training based on the validation data inputted. The *fit()* function accepts arguments for *batch_size* and *epochs* were set to 32 and 12, respectively, for all DenseNet model experimentation.

The best DenseNet CNN model was selected using the *train_and_select_model()* function that accepts training and validation data and calls the aforementioned *create_and_train_cnn()* model on lists of potential values for various hyperparameters. The following hyperparameters were altered: learning rate, optimizer, and growth rate of the DenseNet. The learning rate defines how much to alter the model in response to the estimated error each time the model's weights are updated. The values tested were [0.001, 0.0001, 0.00001]. The optimizer function updates the model in response to the output of the loss function. The optimizers tested were ['adamax', 'nadam', 'adam']. The growth rate applies to each dense block in the DenseNet and refers to how many feature maps each dense block successively adds per layer it contains (e.g., the l^{th} layer has $k_0 + K(l - 1)$ input feature maps, where k is the growth rate). The values tested were [6, 12, 16, 24]. The 'activation' hyperparameter was not tuned because the DenseNet relies on specific activation functions for each block and layer. ReLU is used within the dense blocks and the model is concluded with a final Dense layer that contains a 'softmax' activation function.

A traditional CNN was created to compare accuracies and performance to the DenseNet model. The simple CNN architecture contained the following layers: convolution with 128 filters, maxpooling, flatten, dropout (20%), flatten, and a final dense layer. The arguments for *batch_size* and *epochs* were set to 8 and 8, respectively. The traditional CNN architecture is noticeably less complex than the DenseNet architecture, though the resulting traditional CNN model is substantially more complex than the DenseNet's, at approximately 5.9 million parameters versus roughly 600,000.

Data Summary

As mentioned previously, our project utilized a 2750 image, 11 GB dataset uploaded by user Pablo Lopez Santori on Kaggle. The dataset was originally sourced from leading clinical centers around the world, using a variety of devices, by the International Skin Imaging Archive (ISIC) in 2018, and reflects real world skin lesions that dermatologists treat. There are two key traits about the dataset that likely held a substantial effect on the generalizability, robustness, and validity of our results: 1) the heavily imbalanced distribution of nevus, melanoma, and seborrheic keratosis images (**Figure 4**), and 2) the size of the individual images before augmentation, which reflected a median width of 3008 *pixels* and median height of 2000 *pixels*.

Figure 4: Depicted is the class distribution of the skin lesion image dataset depending on the split (i.e., train set, validation set, or test set). As can be observed qualitatively in the plot, nevus images heavily dominated the data set (see **Table 1** for the quantitative statistics).

The first of the key traits, the uneven distribution of the data, is demonstrated by **Figure 4** and **Table 1**, which display the overrepresentation of nevus samples in the dataset qualitatively and quantitatively, respectively. This substantial imbalance in the dataset may likely have been the factor for the overfitting behavior of the traditional CNN model, as discussed later in the **Experimental Results** and **Experimental Analysis** sections.

The second key trait of the dataset that may have introduced inherent limitations of the analysis was the 3008x2000 $pixels^2$ size of the individual images. Due to the limited allocation of resources provided by Google Colaboratory, this image size necessitated that the images be downsampled to 128x128 $pixels^2$. As seen above in **Figure 5**, applying such a small scaling factor to the images resulted in the resized images possessing an area of 0.27% that of the original images. Such a large degree of downsampling removed a substantial amount of information from the image, which in turn may have hindered the effectiveness of CNN model training, regardless of the architecture utilized.

Distribution of Split Data by Class (Frequency)

	Nevus	*Melanoma*	*Seborrheic Keratosis (SBRK)*
Training	1372	374	254
Validation	78	30	42
Testing	393	117	90

Table 1: Depicted are the quantities of each class's images across the split of the given Kaggle dataset. As can be observed quantitatively in the table, nevus images heavily dominated the dataset, especially in the train split, where the quantity of nevus images were more than twice the amount of melanoma and seborrheic keratosis images combined. This suggests a high level of imbalance which could likely explain the suboptimal accuracy obtained from the DenseNet models (see **Experimental Results**).

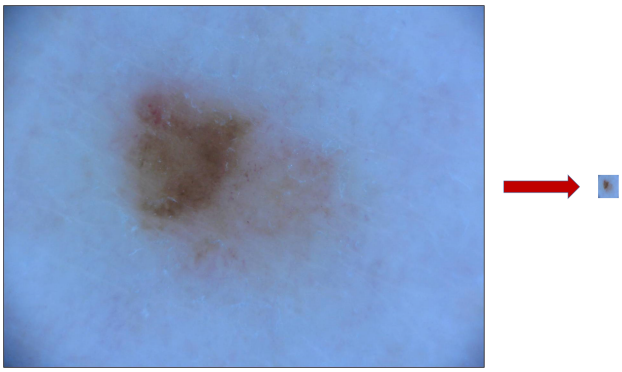


Figure 5: Depicted is a qualitative demonstration of the image resizing process that was necessary to be conducted secondary to the limitations of Google Colaboratory's computational resources. As can be seen visually, the images were resized from a median size of 3008x2000 $pixels^2$ to a size of 128x128 $pixels^2$. This substantial downsampling (area of the resized images was 0.27% of the original median image area) resulted in large amounts of image information being removed, which in turn may have limited the effectiveness of model training and even image augmentation.

Despite these inherent limitations, we still believe that the dataset utilized allowed us to effectively achieve the primary endpoints of the analysis (see **Contributions** section). Since the dataset was suboptimal for the reasons outlined above, the DenseNet model's evaluation of an admirable testing accuracy would allow us to achieve the first contribution. To our knowledge, such a DenseNet model evaluation has not been performed under similar dataset and computational restrictions. Furthermore, training a traditional CNN on the skin lesion images and comparing the testing accuracy obtained with that of the DenseNet model would allow us to achieve the second contribution. Finally, the tuning of the DenseNet architecture to construct a skin lesion image CNN classifier of this nature, which to our knowledge has not yet been performed, would allow us to achieve the third and final contribution.

Experimental Results

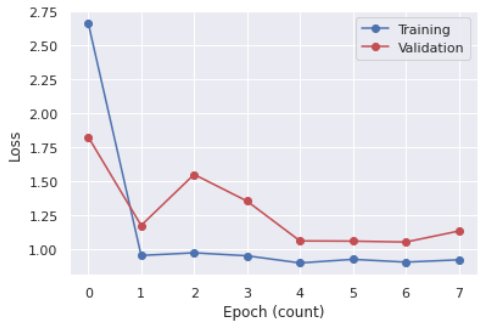
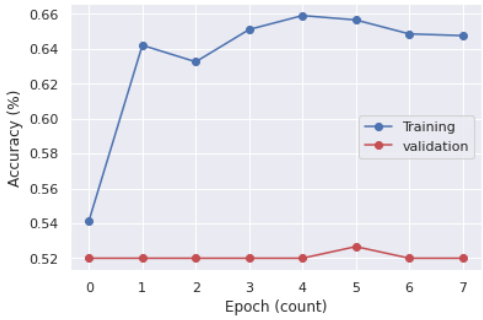
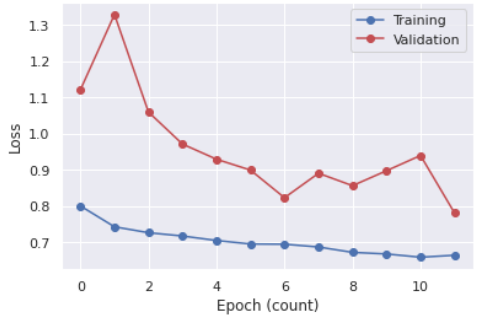
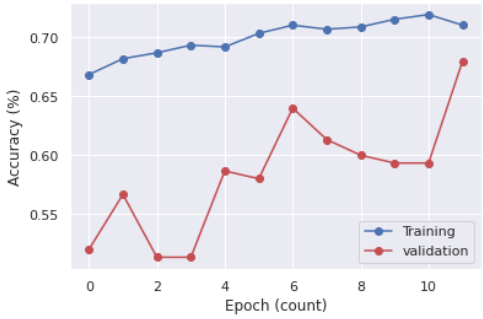
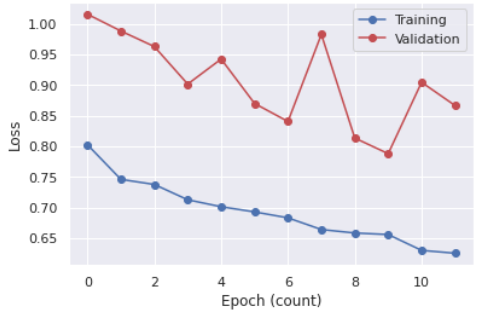
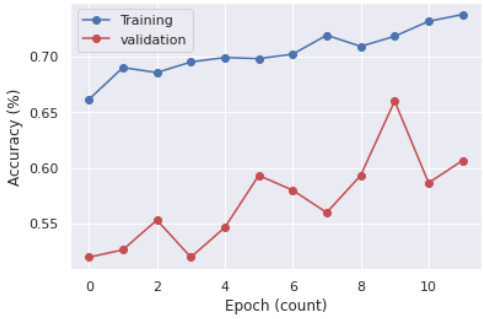
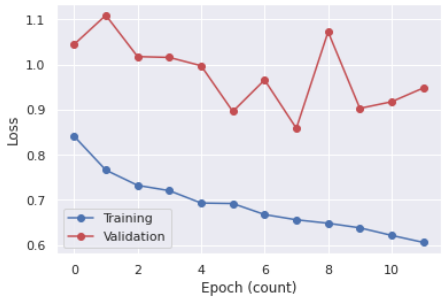
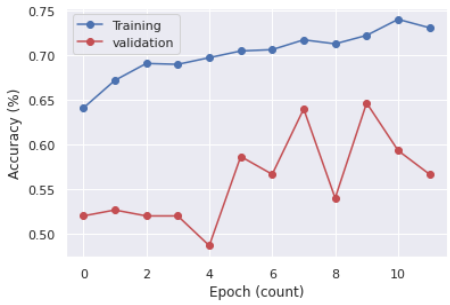
CNN Model	*Loss vs Epoch*	*Validation Accuracy vs Epochs*
A) *Traditional* *LR=0.0001* *Opt: 'adamax'* *AF: 'softmax'*		
B) *DenseNet* *LR: 0.0001* *Opt:'nadam'* *Growth Rate: 6* *Dense Blocks: 4* *Layers/Block: 4*		
C) *DenseNet* *LR: 0.00001* *Opt: 'nadam'* *Growth Rate: 6* *Dense Blocks: 4* *Layers/Block: 4*		
D) *DenseNet* *LR: 0.00001* *Opt: 'nadam'* *Growth Rate: 12* *Dense Blocks: 6* *Layers/Block: 4*		

Figure 6: Depicted are the loss and validation accuracy as a function of epoch for each of the models tested, including the traditional CNN architecture (**A**) and several DenseNet architectures, the best of which is designated as **D**. As seen visually, the validation accuracy for DenseNet models was variable across each epoch, whereas the traditional CNN model tended to be more stable but shows more signs of overfitting, i.e. the larger gulf between training and validation accuracy as shown in **A** when compared with **B-D**.

State of the art CNN models utilize larger, more balanced training and validation datasets; furthermore, they usually are trained under computational conditions permissive to the use of larger image sizes and more complex models. For example, Esteva et al. used transfer learning with a pre-trained GoogleNet Inception v3 CNN architecture on a dataset that possessed approximately 130,000 images with an average size of 299x299 $pixels^2$, though it is important to note that their model was trained using 757 disease classes as opposed to three. It is likely easier to achieve and exceed dermatologist-level classification under computational and dataset conditions such as these. However, to our knowledge, prior studies have not been carried out under conditions in which the training sets were heavily imbalanced or limited computational power necessitated downsampling to the degree shown in **Figure 5**.[16] Additionally, most state-of-the art models do not utilize the DenseNet architecture, and instead typically opt for more traditional CNN architectures that produce models with greater complexities due to the larger number of parameters required.[16,17]

Validation and Testing Accuracy for each Model				
	Model A	**Model B**	**Model C**	**Model D**
Best Validation Accuracy	52.66%	68.00%	66.00%	64.66%
Test Accuracy	65.00%	62.50%	57.50%	60.00%

Table 2: Depicted are the validation and testing accuracies for each model, with **Model A**, the traditional CNN, and **Models B**, **C**, and **D**, the DenseNet models. Though **Model D** did not have the highest testing or validation accuracies, it was deemed the best as its confusion matrix (**Figure 7**) was the most well distributed.

In accordance with our three contributions, several metrics were utilized in order to quantify the effectiveness of our model(s) in classifying the skin lesion images. First, the validation losses and validation accuracies were plotted as a function of the number of epochs for each model trained (**Figure 6**), the best validation accuracy of which was utilized to select the best model during any given run (**Table 2**). Second, the testing accuracies of the top three DenseNet models, in addition to the traditional CNN model, were compared (**Table 2**). Third, a confusion matrix was generated and the class accuracy, precision, recall, specificity, and F1-score was calculated for each model outlined in the previous sentence (see **Table 4** in **Appendix A** for the list of equations utilized to quantify these metrics). For the class metric determined through use of the model's confusion matrix, the quantity of true positives (TPs), true negatives (TNs), false positives (FPs), and false negatives (FNs) when predicting any given image class were utilized (see **Table 5** in **Appendix B** for the definitions of TP, TN, FP, and FN). Finally, the top three DenseNet models and the traditional CNN model were compared through utilization of these metrics to determine which model was most suitable for classifying skin lesion images.

Figure 6 depicts the validation losses and validation accuracies for each model as a function of the number of epochs. **Model A** represents the best traditional CNN model we found, as determined by its maximum validation accuracy of 52.66% (**Table 2**). Hyperparameter tuning of the traditional CNN models, as described in the **Implementation** section above, yielded an optimal learning rate of 0.0001, an 'adamax' optimizer, and a 'softmax' activation function for **Model A** (see .html file in attachments for the model architecture of the traditional CNNs utilized, each of which possessed 5,907,971 total parameters). The testing accuracy of **Model A** was found to be 65.00%. Although this relatively high testing accuracy may

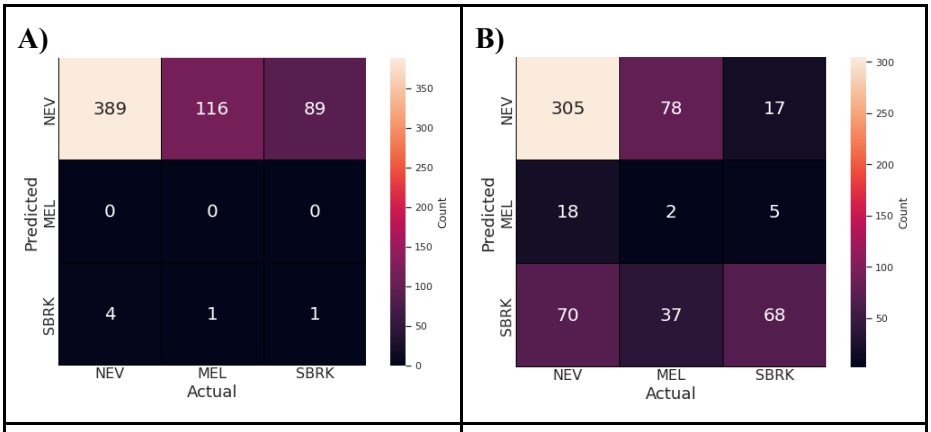


Figure 7: Depicted are the confusion matrices for each of the models tested. **A** refers to the traditional CNN model, where **B-D** refer to the DenseNet models. **D** represents the best model, with a more generalized solution than other models while still maintaining a relatively high accuracy. As seen in **A**, the traditional CNN model almost exclusively classified images as nevus, showing behavior consistent with overfitting to nevus samples.

make its performance seem robust and generalizable with respect to classifying skin lesion images, the confusion matrix of **Model A** (depicted in **Figure 7A**) suggests otherwise. This confusion matrix shows that **Model A** predicted 0 melanoma images and a total of 6 seborrheic keratosis images when evaluated on the testing dataset, suggesting that the tuned CNN with the highest validation accuracy was highly susceptible to the large nevus-heavy imbalance in the training dataset (see **Figure 4** for qualitative distributions and **Experimental Analysis** section below for further discussion). Furthermore, it rendered the quantitative determinations of class accuracy, precision, recall, specificity, and F1-score for the traditional CNN model unable to be calculated, as the melanoma class metrics were subject to a divide-by-zero error.

Models B-D (**Figure 6**), on the other hand, represent the three best DenseNet models found as a result of hyperparameter tuning and determining the highest validation accuracy achieved. **Model B** possessed a learning rate of 0.0001, a 'nadam' optimizer, a growth rate of 6, 4 dense blocks, and 4 convolutional layers per block. The only difference between the hyperparameters of **Model B** and **Model C** was the learning rate, as **Model C** possessed a learning rate of 0.00001. Finally, **Model D** possessed a learning rate of 0.00001, a 'nadam' optimizer, a growth rate of 12, 6 dense blocks, and 4 layers per block (see .html file for the architecture of all DenseNet models, each of which possessed 605,811 total parameters). The best validation accuracies of **Models B-D** were found to be 68.00%, 66.00%, and 64.66%, respectively. The testing accuracies of **Models B-D**, when evaluated on the testing dataset, were found to be 62.50%, 57.50%, and 60.00%, respectively.

When comparing the average testing accuracy of the three best DenseNet CNN models (**Models B-D**; approximately 60.00%) relative to the testing accuracy of the best traditional CNN model (**Model A**; 65.00%), it may be tempting to decide that **Model A**, the traditional CNN model, achieved the best performance. However, as mentioned previously, the confusion matrix of **Model A** (**Figure 7A**) suggested

that the traditional CNN was predicting virtually all nevus images from the testing data inputted, with 0 predictions for melanoma and only 6 predictions for seborrheic keratosis out of the 600 images predicted on. On the other hand, the confusion matrices of **Models B-D** were more balanced; furthermore, they were able to generate class accuracy, prediction, recall, specificity, and sensitivity metrics for nevus, melanoma, and seborrheic keratosis whereas **Model A** was unable to do so for melanoma (**Table 3**). Thus, through careful confusion matrix analysis, we were able to effectively achieve the second endpoint outlined in the **Contributions** section above—2) the display of a comparison between the performance of a cutting-edge DenseNet CNN model relative to that of a traditional CNN model. The testing accuracy, class accuracy, precision, recall, specificity, and F1-score metrics of **Models B-D** (**Table 2** and **Table 3**) also allowed us to highlight the first and third endpoints outlined in the **Contributions section** above—1) the use of a DenseNet for melanoma classification on a non-optimal dataset, and 3) the tuning of DenseNet hyperparameters and structure to advance work on medical skin lesion image classification investigations. Further discussion of how the results specifically suit the contributions is present in the **Experimental Analysis** section below.

Evaluation Metrics of Accuracy, Precision, Recall, Specificity, and F1-Score across each Class and Convolutional Neural Network Model

Class	Model	Accuracy	Precision	Recall	Specificity	F1-Score
Nevus	**A**	0.65	0.65	0.99	0.01	0.78
	B	0.69	0.76	0.78	0.54	0.77
	C	0.61	0.72	0.68	0.49	0.70
	D	0.68	0.79	0.70	0.64	0.74
Melanoma	**A**	0.81	UNDEF	0.0	1.00	UNDEF
	B	0.77	0.08	0.02	0.95	0.03
	C	0.79	0.39	0.14	0.95	0.20
	D	0.68	0.28	0.43	0.74	0.34
Seborrheic Keratosis	**A**	0.84	0.16	0.01	0.99	0.02
	B	0.79	0.39	0.76	0.79	0.51
	C	0.75	0.33	0.69	0.76	0.45
	D	0.84	0.46	0.38	0.92	0.41

Table 3: Depicted are the metrics used to quantify success of each model in classifying nevus, melanoma, and seborrheic keratosis data. The best model, **D**, is shaded in blue. Shaded in red is model **A**, the traditional CNN. Note that model **A** is nonsensical for melanoma, as values for precision and F1-score are undefined, and shows evidence of massively overfitting nevus data, with extremely high recall but extremely low specificity, with the converse being true for seborrheic keratosis data.

As depicted in **Table 3**, **Model B** achieved the highest nevus class accuracy (69.00%) and melanoma class accuracy (77.00%) among the three best DenseNet models tuned. Furthermore, we found that **Model B** possessed the highest F1-scores in relation to nevus and seborrheic keratosis classification (77.00% and 51.00%, respectively). However, the precision and recall rates of **Model B** with respect to melanoma were alarmingly low (8.00% and 2.00%, respectively). Overall, this suggests that **Model B** was also susceptible to the substantial nevus-heavy imbalance possessed in the dataset, similar to what was found when analyzing the results of **Model A**. The confusion matrix of **Model B** (**Figure 7B**) supports this limiting factor, as only 25 of the 600 predictions evaluated on the test dataset belonged to the melanoma class. As a result, the F1-score of **Model B** with respect to melanoma was found to be 3.00%, which similarly suggests a high susceptibility to avoiding melanoma classifications. The imbalance on the dataset likely had a similar effect on the performance of **Model C**, as its recall and F1-scores with respect to the melanoma class were relatively low (14.00% and 20.00%, respectively), while its metrics across the board for nevus were relatively high (61.00% nevus accuracy, 72.00% nevus precision, and 70.00% nevus F1-score). An interesting observation, however, with regard to **Model C** was that 186 out of its 600 predictions on the test dataset belonged to the seborrheic keratosis class, which was a higher quantity in comparison to the quantity of seborrheic keratosis predictions of both **Model B** and **Model D** (175 and 74, respectively). Nevertheless, we found that **Model D** possessed the most balanced, and overall high class accuracy, precision, recall, specificity, and F1-score rates with all three classes of interest (see **Table 3**).

Experimental Analysis

Discussion

To reiterate, the following are the contributions we intended to achieve from this analysis:

1) The use of a DenseNet for melanoma classification on a non-optimal dataset
2) The display of a comparison between a cutting-edge DenseNet model versus a traditional CNN model
3) The tuning of DenseNet hyperparameters and structure to advance current research on medical skin lesion image classification.

With regard to the first contribution, the skin lesion dataset we utilized in the analysis was heavily imbalanced in favor of nevus images (**Figure 4** and **Table 1**), yet the DenseNet CNN models (**Models B-D**) performed admirably, with testing accuracies resulting from predicting the testing dataset being 62.50%, 57.50%, and 60.00%, respectively (**Table 2**). Furthermore, the class accuracies resulting from confusion matrix analysis (**Table 3**) for the DenseNet models were all found to be over 60.00% with respect to nevus, melanoma, and seborrheic keratosis.

In regards to the second contribution, we trained a traditional CNN model (**Model A**), consisting of 5,907,971 total parameters, on the skin lesion images dataset and achieved a testing accuracy of 65.00% (**Table 2**). Although this was a higher testing accuracy than the average of all three testing accuracies achieved through use of DenseNet **Models B-D**, the confusion matrix of **Model A** was extremely sparse, as it predicted melanoma 0 times and seborrheic keratosis 6 times for the testing dataset. Furthermore, neither the melanoma class precision nor melanoma F1-score was able to be calculated for **Model A** due to the divide-by-zero error resulting from the confusion matrix. In contrast, the confusion matrices of **Models B-D** were more evenly distributed. As a result, our results suggest that the cutting-edge DenseNet CNN model performs better than a traditional CNN model for multiclass medical skin lesion image classification.

Finally, the results of a careful confusion matrix analysis and predictions on a testing dataset were compared on DenseNet **Models B-D** to achieve the third contribution. We found that **Model B** possessed the highest nevus class accuracy out of the three DenseNet models (**Table 3**), but also found similar patterns of melanoma classification avoidance in this model (23 out of 600 testing set predictions). This latter observation likely resulted in the melanoma precision and recall rates for **Model B** being extremely low (8.00% and 2.00%, respectively; **Table 3**). **Model C** yielded an improvement in the F-1 Score for melanoma classification (20.00% relative to 3.00% for **Model B**), yet the overall testing accuracy was quite low (57.50%; **Table 2**), and the lack of melanoma predictions (41 out of 600 testing set predictions) suggested similar limitations. Thus, we declared **Model D**, which was trained with a learning rate of 0.00001, a 'nadam' optimizer, a growth rate of 12, 6 dense blocks, and 4 blocks per layer, to be the best-performing DenseNet CNN model that we trained in classifying the skin lesion images. **Model D** achieved a testing accuracy of 60.00%, and had the most evenly distributed confusion matrix of the four outlined above (**Figure 7D**). The F-1 scores of **Model D** for the nevus, melanoma, and seborrheic keratosis classes were found to be 74.00%, 34.00%, and 41.00%, respectively.

Errors and Limitations

Several sources of error limit both the findings that were outlined in the previous section and the validity of our contributions as a whole. The most pertinent source of probable error stemmed from the dataset we utilized to train our models. Image classification deep learning algorithms and CNN architectures typically utilize far greater quantities of images (e.g., Esteva et al. used approximately 130,000 images to train their skin disease classification model).[16] The skin lesions dataset utilized in this analysis, however, possessed a mere 2,750 images, of which 2,000 were used for training. This relatively low training dataset size was further compounded by two secondary sources of error within the dataset. First, the quantity of nevus images in the training split of the dataset (**Figure 4**) largely overshadowed both the quantity of melanoma images and the quantity of seborrheic keratosis images (1372 compared to 374 and 254, respectively; **Table 1**). This skew represents a substantial imbalance in the dataset utilized for training, which resulted in overfitting of essentially all models to the nevus class. This is observed most clearly in evaluation metrics of **Model A** and **Model B**, both of which possessed low nevus specificities (1.00% and 54.00%, respectively), low melanoma recall rates (0.00% and 2.00%, respectively), and extremely high melanoma specificities (100.00% and 95.00%, respectively). Overall, the substantial nevus-heavy imbalance in the training dataset likely led these models to gain substantial information about images in this class relative to the images in the melanoma and seborrheic keratosis classes.

However, another problem also existed in the dataset, but at the individual image level rather than the dataset as a whole. This was the resizing of the images from a median size of 3008x2000 $pixels^2$ to an augmented size of 128x128 $pixels^2$. As depicted qualitatively in **Figure 5**, this resizing led to the areas of the resized images being 0.27% of the median area across the original images. Such substantial downsampling also likely contributed to the loss of information gained from the training dataset, yet it was necessary due to the computational resources we had in our possession. Since Google Colaboratory was utilized to conduct our analysis, the training of any images larger than 128x128 $pixels^2$ led to a resource exhaustion error. The computational limitations of Google Colaboratory also severely constrained the architecture of the CNN model we could construct, thus rendering the analysis of other cutting-edge image classification models—such as U-Net or ResNet—unable to be conducted. These resource limits required that we balanced model and image complexity with computational speed.

Conclusion and Future Work

Conclusion

2,750 skin lesion images containing indications to nevus, melanoma, or seborrheic keratosis were classifie through utilization of both a traditional CNN model architecture (5,907,971 total parameters) and a cutting-edge DenseNet CNN model architecture (605,811 total parameters). Both CNN model architectures were tuned to find the optimal hyperparameters during training, the best model(s) being determined by their maximum validation accuracy over a certain range of epochs. **Model A** represented the traditional CNN model, and it achieved a testing accuracy of 65.00%. However, the confusion matrix of **Model A** was extremely uneven, and it did not predict melanoma during the testing phase once. In contrast, the DenseNet CNN Models (**Models B-D)**, which achieved testing accuracies of 62.50%, 57.50%, and 60.00%, respectively, possessed more evenly distributed confusion matrices. However, we found that **Model D** performed the best out of these DenseNet models, as it balanced high nevus classification accuracies with more reasonable values for melanoma and seborrheic keratosis precision, recall, and F1-score metrics. Thus, we concluded that **Model D**, a DenseNet CNN model which possessed a learning rate of 0.00001, a 'nadam' optimizer, a growth rate of 12, 6 dense blocks, and 4 layers per block, exhibited the best performance of all CNN models that were trained and tested. In doing so, we also showed that DenseNet CNN models perform admirably on non-optimal datasets, found that DenseNet CNN models perform better than traditional CNN models when classifying skin lesion images, and contributed to current research on skin lesion image classification by tuning both the hyperparameters and structure of the cutting-edge DenseNet CNN model. However, limitations of the analysis included the relatively small quantity of images utilized during training, the heavily imbalanced skin lesion dataset being trained on, and the lack of computational resources at our disposal.

Future Work

Future avenues of research potentially include implementing the U-Net model architecture in our approach to classify the skin lesion images. This is a state-of-the-art CNN architecture being frequently implemented as a medical image segmentation model, and it functions by assigning labels for each pixel in any given image. Training the U-Net model on the images would allow for a higher robustness with regard to the coloration and shape of any given skin lesion. Furthermore, the DenseNet CNN Models constructed in this project's implementation should be trained on a larger and more balanced dataset of skin lesion images in order to achieve higher generalizability and robustness. Moreover, training the models on a dataset with a higher quantity of classes (e.g., addition of basal cell carcinoma and squamous cell carcinoma skin lesion images) could enhance the usability of such a model in regards to real-world dermatologist-level skin disease classifications. Finally, the analysis should be replicated several times with more computational power to reduce the downsampling of image data, which resulted in a substantial loss of information.

References

1. Melanoma - Symptoms and causes. Mayo Clinic. Accessed October 11, 2020. https://www.mayoclinic.org/diseases-conditions/melanoma/symptoms-causes/syc-20374884

2. Glazer AM, Winkelmann RR, Farberg AS, Rigel DS. Analysis of Trends in US Melanoma Incidence and Mortality. *JAMA Dermatol*. 2017;153(2):225. doi:10.1001/jamadermatol.2016.4512

3. Melanoma Incidence and Mortality, United States–2012–2016 | CDC. Published September 16, 2020. Accessed October 31, 2020. https://www.cdc.gov/cancer/uscs/about/data-briefs/no9-melanoma-incidence-mortality-UnitedStates-2012-2016.htm

4. American Cancer Society. What Is Melanoma Skin Cancer? | What Is Melanoma? Published August 14, 2019. Accessed October 11, 2020. https://www.cancer.org/cancer/melanoma-skin-cancer/about/what-is-melanoma.html

5. Marghoob AA, Scope A. The complexity of diagnosing melanoma. *J Invest Dermatol*. 2009;129(1):11-13. doi:10.1038/jid.2008.388

6. Melanoma - Diagnosis. Cancer.Net. Published June 25, 2012. Accessed October 31, 2020. https://www.cancer.net/cancer-types/melanoma/diagnosis

7. Domingues B, Lopes JM, Soares P, Pópulo H. Melanoma treatment in review. *ImmunoTargets Ther*. 2018;7:35-49. doi:10.2147/ITT.S134842

8. Halpern A, Marghoob A, Reiter O. Melanoma. The Skin Cancer Foundation. Accessed October 31, 2020. https://www.skincancer.org/skin-cancer-information/melanoma/

9. Voss RK, Woods TN, Cromwell KD, Nelson KC, Cormier JN. Improving outcomes in patients with melanoma: strategies to ensure an early diagnosis. *Patient Relat Outcome Meas*. 2015;6:229-242. doi:10.2147/PROM.S69351

10. Financial help for people who have skin cancer. Accessed October 31, 2020. https://www.aad.org/public/diseases/skin-cancer/types/common/melanoma/financial-help

11. Fawzy FI, Fawzy NW, Hyun CS, et al. Malignant melanoma. Effects of an early structured psychiatric intervention, coping, and affective state on recurrence and survival 6 years later. *Arch Gen Psychiatry*. 1993;50(9):681-689. doi:10.1001/archpsyc.1993.01820210015002

12. Zhu W, Xie L, Han J, Guo X. The Application of Deep Learning in Cancer Prognosis Prediction. *Cancers*. 2020;12(3). doi:10.3390/cancers12030603

13. Landhuis E. Deep learning takes on tumours. *Nature*. 2020;580(7804):551-553. doi:10.1038/d41586-020-01128-8

14. Nadeem MW, Ghamdi MAA, Hussain M, et al. Brain Tumor Analysis Empowered with Deep Learning: A Review, Taxonomy, and Future Challenges. *Brain Sci*. 2020;10(2). doi:10.3390/brainsci10020118

15. Wenzhong L, Huanlan L, Caijian H, Liangjun Z. *Classifications of Breast Cancer Images by Deep Learning*. Radiology and Imaging; 2020. doi:10.1101/2020.06.13.20130633

16. Esteva A, Kuprel B, Novoa RA, et al. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*. 2017;542(7639):115-118. doi:10.1038/nature21056

17. Chan S, Reddy V, Myers B, Thibodeaux Q, Brownstone N, Liao W. Machine Learning in Dermatology: Current Applications, Opportunities, and Limitations. *Dermatol Ther*. 2020;10(3):365-386. doi:10.1007/s13555-020-00372-0

18. Ray R, Abdullah AA, Mallick DK, Ranjan Dash S. Classification of Benign and Malignant Breast Cancer using Supervised Machine Learning Algorithms Based on Image and Numeric Datasets. *J Phys Conf Ser*. 2019;1372:012062. doi:10.1088/1742-6596/1372/1/012062

19. Dhahri H, Al Maghayreh E, Mahmood A, Elkilani W, Faisal Nagi M. Automated Breast Cancer Diagnosis Based on Machine Learning Algorithms. Journal of Healthcare Engineering. doi:https://doi.org/10.1155/2019/4253641

20. Melanoma Detection Dataset. Accessed October 31, 2020. https://kaggle.com/wanderdust/skin-lesion-analysis-toward-melanoma-detection

21. Huang G, Liu Z, van der Maaten L, Weinberger KQ. Densely Connected Convolutional Networks. *ArXiv160806993 Cs*. Published online January 28, 2018. Accessed December 6, 2020. http://arxiv.org/abs/1608.06993

22. Harper X. Normalize Data - Azure. Accessed December 9, 2020. https://docs.microsoft.com/en-us/azure/machine-learning/studio-module-reference/normalize-data

23. Brownlee J. How to Configure Image Data Augmentation in Keras. Machine Learning Mastery. Published April 11, 2019. Accessed December 6, 2020. https://machinelearningmastery.com/how-to-configure-image-data-augmentation-when-training-deep-learning-neural-networks/

24. Feng X, Kramer CM, Salerno M, Meyer H. Automatic Scar Segmentation from DE-MRI Using 2D Dilated UNet with Rotation-based Augmentation. :5.

Appendices

Appendix A

Overall Testing Accuracy, Class Accuracy, Precision, Recall, Specificity, and F1-Score Equations		
Metric	*Equation*	*Designation*
Testing Accuracy	$Num\ Correct\ /\ Total\ Num$	Eq. 2
Class Accuracy	$(TP + TN)\ /\ (TP + TN + FP + FN)$	Eq. 3
Precision	$TP\ /\ (TP + FP)$	Eq. 4
Recall	$TP\ /\ (TP + FN)$	Eq. 5
Specificity	$TN\ /\ (TN\ +\ FP)$	Eq. 6
F1	$(2 * Precision * Recall)\ /\ (Precision + Recall)$	Eq. 7

Table 4: Depicted are the equations utilized to calculate the overall testing accuracy, class accuracy, class precision, class recall, class specificity, and class F1-Score metrics for each model. Testing accuracy refers to the overall accuracy of the model's predictions on the testing dataset, and is calculated by dividing the total number of correct predictions by the quantity of images in the testing dataset (600). Class accuracy refers to the fraction of the total samples of the class that were correctly classified by the model. Class precision refers to the fraction of predictions as a positive class that were actually positive. Class recall, also known as sensitivity, true positive rate, or probability of detection, refers to the fraction of positive samples that were correctly classified as positive by the classifier. Class specificity, also known as the true negative rate, refers to the fraction of negative samples that were correctly predicted as negative by the classifier. Finally, the F1-Score, mathematically known as the harmonic mean of precision and recall, combines precision and recall into a single measure. For an ideal model, we would like the F1-Score of each class to be as close to 1 as possible.

Appendix B

Definitions of a True Positive, True Negative, False Positive, and False Negative	
Metric	*Definition*
True Positive (TP)	Quantity of predictions where the classifier correctly predicts the positive class as positive.
True Negative (TN)	Quantity of predictions where the classifier predicts the negative class as negative.
False Positive (FP)	Quantity of predictions where the classifier incorrectly predicts the negative class as positive. This is also known as a Type I error.
False Negative (FN)	Quantity of predictions where the classifier incorrectly predicts the positive class as negative. This is also known as a Type II error.

Table 5: Depicted are the definitions of a true positive, true negative, false positive, and false negative. These were used in the analysis to quantify the class accuracy, class precision, class recall, class specificity, and class F1-score metrics for each model's performance (see **Table 4**).