

Submitted by

DR Aditi Yadav

Melanoma Skin Cancer Detection

Abstract

In cancer, there are over 200 different forms. Out of 200, melanoma is the deadliest form of skin cancer. The diagnostic procedure for melanoma starts with clinical screening, followed by dermoscopic analysis and histopathological examination. Melanoma skin cancer is highly curable if it gets identified at the early stages. The first step of Melanoma skin cancer diagnosis is to conduct a visual examination of the skin's affected area. Dermatologists take the dermoscopic images of the skin lesions by the high-speed camera, which have an accuracy of 65-80% in the melanoma diagnosis without any additional technical support. With further visual examination by cancer treatment specialists and dermoscopic images, the overall prediction rate of melanoma diagnosis raised to 75-84% accuracy. The project aims to build an automated classification system based on image processing techniques to classify skin cancer using skin lesions images.

Problem Statement

In the skin biopsy, the dermatologist takes some part of the skin lesion and examines it under the microscope. The current process takes almost a week or more, starting from getting a dermatologist appointment to getting a biopsy report. The aims to shorten the current gap to just a couple of days by providing the predictive model. The approach uses Convolutional Neural Network (CNN) to classify nine types of skin cancer from outlier lesions images. This reduction of a gap has the opportunity to impact millions of people positively.

Motivation

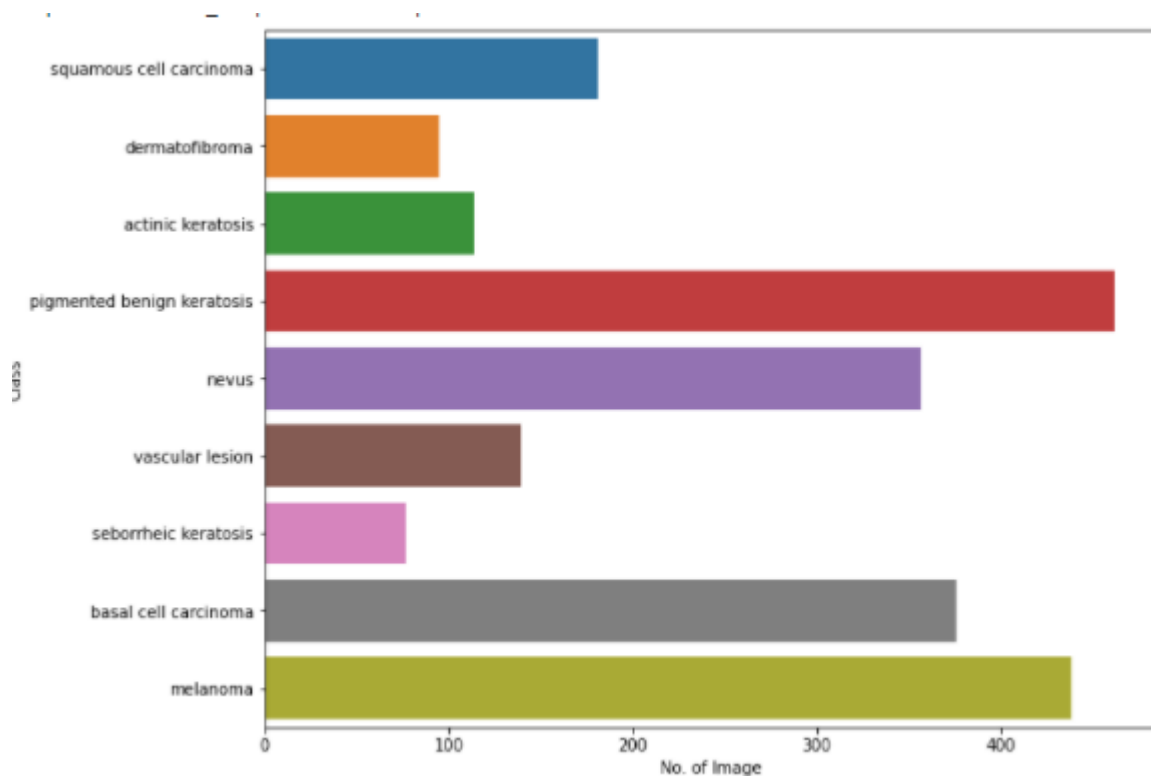
The overarching goal is to support the efforts to reduce the death caused by skin cancer. The primary motivation that drives the project is to use the advanced image classification technology for the well-being of the people. Computer vision has made good progress in machine learning and deep learning that are scalable across domains.

Dataset

The dataset consists of 2357 images of malignant and benign oncological diseases, which were formed from the International Skin Imaging Collaboration (ISIC). All images were sorted according to the classification taken with ISIC, and all subsets were divided into the same number of images.

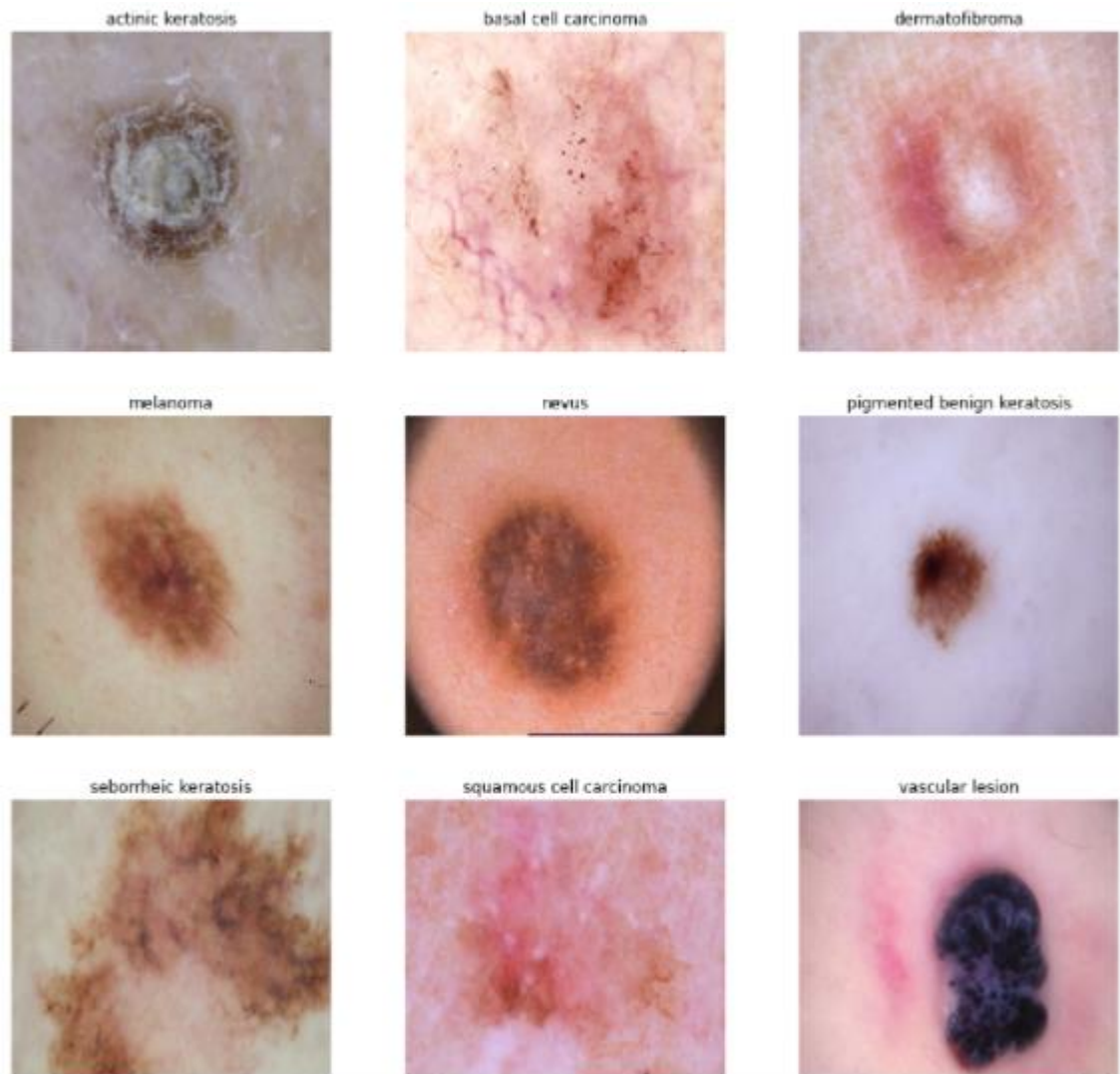
The data set contains the following diseases:

	Class	No. of Image
0	squamous cell carcinoma	181
1	dermatofibroma	95
2	actinic keratosis	114
3	pigmented benign keratosis	462
4	nevus	357
5	vascular lesion	139
6	seborrheic keratosis	77
7	basal cell carcinoma	376
8	melanoma	438



To overcome the issue of class imbalance, used a python package Augmentor (<https://augmentor.readthedocs.io/en/master/>) to add more samples across all classes so that none of the classes have very few samples.

Sample image from Dataset



CNN Architecture Design

To classify skin cancer using skin lesions images. To achieve higher accuracy and results on the classification task, I have built custom CNN model.

- Rescaling Layer - To rescale an input in the $[0, 255]$ range to be in the $[0, 1]$ range.
- Convolutional Layer - Convolutional layers apply a convolution operation to the input, passing the result to the next layer. A convolution converts all the pixels in its receptive field into a single value. For example, if you would apply a convolution to an image, you will be decreasing the image size as well as bringing all the information in the field together into a single pixel.
- Pooling Layer - Pooling layers are used to reduce the dimensions of the feature maps. Thus, it reduces the number of parameters to learn and the

amount of computation performed in the network. The pooling layer summarises the features present in a region of the feature map generated by a convolution layer.

- Dropout Layer - The Dropout layer randomly sets input units to 0 with a frequency of rate at each step during training time, which helps prevent overfitting.
- Flatten Layer - Flattening is converting the data into a 1-dimensional array for inputting it to the next layer. We flatten the output of the convolutional layers to create a single long feature vector. And it is connected to the final classification model, which is called a fully-connected layer.
- Dense Layer - The dense layer is a neural network layer that is connected deeply, which means each neuron in the dense layer receives input from all neurons of its previous layer.
- Activation Function(ReLU) - The rectified linear activation function or ReLU for short is a piecewise linear function that will output the input directly if it is positive, otherwise, it will output zero. The rectified linear activation function overcomes the vanishing gradient problem, allowing models to learn faster and perform better.
- Activation Function(Softmax) - The softmax function is used as the activation function in the output layer of neural network models that predict a multinomial probability distribution. The main advantage of using Softmax is the output probabilities range. The range will 0 to 1, and the sum of all the probabilities will be equal to one.

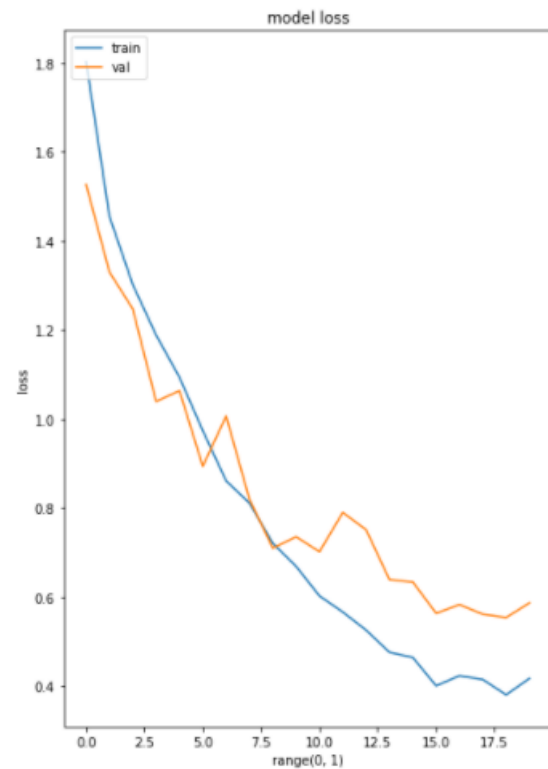
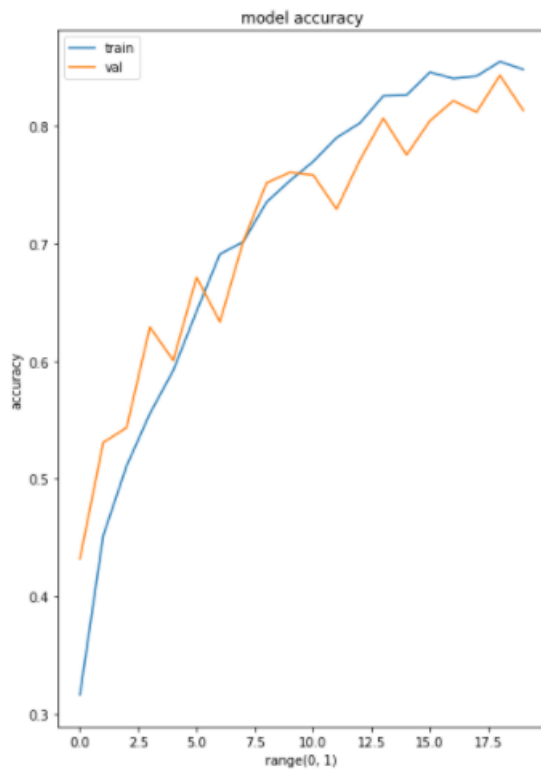
Model Architecture

Model: "sequential"

Layer (type)	Output Shape	Param #
rescaling (Rescaling)	(None, 180, 180, 3)	0
conv2d (Conv2D)	(None, 178, 178, 32)	896
max_pooling2d (MaxPooling2D)	(None, 89, 89, 32)	0
conv2d_1 (Conv2D)	(None, 87, 87, 64)	18496
max_pooling2d_1 (MaxPooling2D)	(None, 43, 43, 64)	0
conv2d_2 (Conv2D)	(None, 41, 41, 128)	73856
max_pooling2d_2 (MaxPooling2D)	(None, 20, 20, 128)	0
dropout (Dropout)	(None, 20, 20, 128)	0
flatten (Flatten)	(None, 51200)	0
dense (Dense)	(None, 128)	6553728
dropout_1 (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 9)	1161

=====
Total params: 6,648,137
Trainable params: 6,648,137
Non-trainable params: 0

Model Evaluation



References

Melanoma Skin Cancer from <https://www.cancer.org/cancer/melanoma-skin-cancer/about/what-is-melanoma.html>

Introduction to CNN
from <https://www.analyticsvidhya.com/blog/2021/05/convolutional-neural-networks-cnn/>

Image classification using CNN
from <https://www.analyticsvidhya.com/blog/2020/02/learn-image-classification-cnn-convolutional-neural-networks-3-datasets/>

Efficient way to build CNN architecture from <https://towardsdatascience.com/a-guide-to-an-efficient-way-to-build-neural-network-architectures-part-ii-hyperparameter-42efca01e5d7>

Github file link

<https://github.com/DrAditiY/melanoma-assignment>