Northeastern University

IE 7615: Skin Cancer Detection

Camron Garrett
Karan Paresh Samani
Karan Shah

Abstract

The most common cancer in the United States is skin cancer but it is also one of the lowest mortality rates due to early detection. By using convolutional neural networks on the International Skin Image Collaboration's public archive, the process of diagnosing lesions as malignant or benign can be done without going into the doctor's office. A CNN of three convolutional, three pooling layers and a dropout layer will be used as the constant model. A shallow CNN model is used to lower computation as this paper will focus on the preprocessing of data. The data will be preprocessed with flipping, rotating, contrasting, and adding gaussian noise with the goal to improve the model's accuracy without adding any additional layers to the CNN. The preprocessing raised the training accuracy of the model but usually lowered the validation accuracy due to overfitting of the data. The only successful preprocessing technique was adding a small amount of contrast to the images.

1. Introduction

Skin cancer is the most common cancer seen in the United States, approximately 9,500 people are diagnosed with skin cancer every day. Early-stage detection of skin cancer is primarily visual, where common signs of skin cancer are asymmetry, irregular borders, and color changes. Doctors routinely see patients to visually check whether a piece of skin should be biopsied for cancerous cells [1],[2]. The doctor's workload can be reduced by using an Artificial Neural Network (ANN) to diagnose malignant or benign cells quickly and accurately.

This paper proposes using a Convolutional Neural Network (CNN) and supervised learning to diagnose whether a mole is benign or cancerous. The labeled images come from the ISIC (International Skin Image Collaboration) archive [3]. This dataset contains 150,000 labeled images of which 56245 benign and 5854 malignant moles images have been made public. Due to the GPU limitations this dataset will be subsampled to reduce computation time.

The focus of this paper is to analyze the effect preprocessing data has on the overall accuracy of a CNN. To reduce computation time the testing is performed on a shallower network compared to industry standards like ResNET50. However, the unprocessed data will be passed through industry standard CNNs to show the relative accuracy of the shallower network. The goal is to improve the accuracy of a CNN without adding additional computations by slightly altering the inputs.

2. Background

Artificial neural networks have been applied to skin cancer detection in the past. One of the largest contributors to the field has been the ISIC. The ISIC started challenges in 2016 to develop ANNs for lesion segmentation, and disease classification from their dataset. These challenges continued from 2016 –2020 with increasing popularity and accuracy each year. Over the 4-year time span, the overall disease classification accuracy rose by 30%. A large contribution to this increase in accuracy is the increase to the ISIC dataset. The number of labeled lesion pictures grew by a factor of 100 [3]. The most effective model for lesion classification has been shown to be CNNs which perform on par with experts in dermatology. In general, models like Inception v3 and ResNET50 have shown to have the highest accuracy on the full public dataset [4].

3. Approach: Convolution Neural Networks (CNN)

CNN is one of the most popular deep learning models for processing data that contains images or more generally a 2D grid type pattern directly inspired from an animal's visual cortex [8],[9]. It is a development of Multilayer perceptron (MLP) and is designed to learn spatial hierarchies of the features adaptively and automatically [15]. It uses backpropagation method by integrating elemental blocks such as the convolution layer with ReLU activation, feature extraction layer – the pooling layer, and classification layer – the fully connected layer with Softmax activation. An example of a CNN architecture is shown in figure 1 [10].

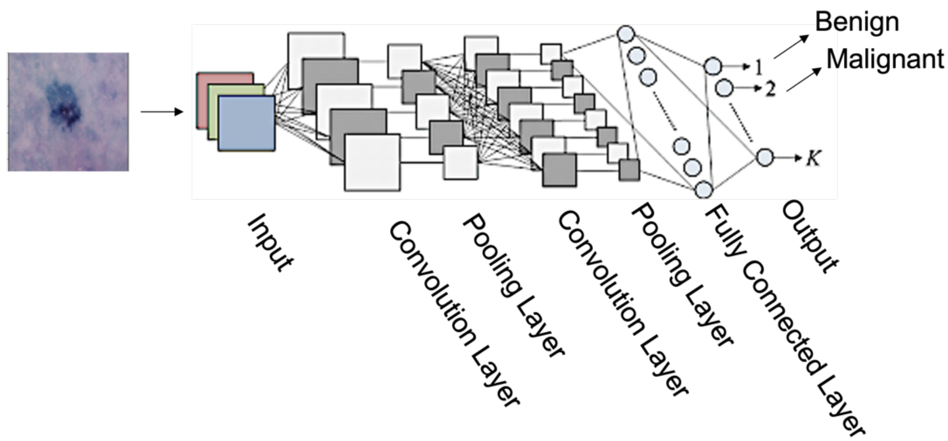

Figure 1: Common CNN Architecture

3.1. Convolution Layer

The first layer, convolution layer, consisting of collection of mathematical operations, plays an integral part for feature extraction. Here, the pixel values from the input tensor are stored in a 2D array of numbers with a small grid of parameters known as kernel. At each point of the input tensor, an element-wise product between the input tensor and each element of the kernel is computed. Further, a feature map is created by summing the product from the previous step to generate the output tensor with corresponding output values. In such manner an arbitrary number of feature maps are created by repeatedly applying the process to multiple kernels. Thus, individual kernels act as a unique feature extractor. The extracted features hierarchically get more complex once a layer passes the output to the next layer

3.1.1. Padding

Padding is a technique used to address the problem of decentering of each kernel on the outer element of the input tensor. This involves adding rows and columns of zeros on all sides of the input tensor so that the center of the kernel falls on the outermost elements. This helps in retaining the in-plane dimensions while conducting the convolution operation and ensures that each succeeding feature map does not get smaller in size once the convolution operation takes place. Zero padding, which is rows and columns of all zeros, is a typical choice for most CNN architecture.

3.1.2. Activation Layer (ReLU)

The output of a convolution operations further passes through an activation function that transforms the summed weighted inputs from the node into the output of that node, or activation of the node. One common activation function is rectified linear activation function (ReLU), which is a piecewise linear function defined as follows:

$$f(x) = x, \ x > 0$$

$$f(x) = 0, \ x \leq 0$$

This means that if the pixel image has a negative value, then ReLU makes that pixel value to be zero. Overall, adding an activation layer like such helps in reducing the error and enables backpropagation as the weights and biases are updated by the gradients that are carried along with the error [12].

3.2. Pooling Layer

Pooling is a down-sampling operation that progressively reduces the in-plane dimensionality of the feature map to prevent over-fitting [13]. This introduces translation invariance to small deformations, which further reduces the learnable parameters, hence reducing the number of computations in the network. Max pooling is the most common technique used in the pooling layer which selects the maximum value out of the available data.

3.3. Fully Connected Layer

The final layer at the end of a CNN architecture in MLP network is a fully connected layer. All the neurons from the preceding activation layer will be connected in this layer. Transformation of the output feature maps from previous layer into a 1D array of vector, known as flattening, takes place followed by the connections between one or more fully connected layers, known as dense layers. These subsets of fully connected layers then map the extracted features from the previous layers to the final outputs of the network in terms of probabilities for each class in the classification task [14]. An activation function such as a Softmax function is used for such mapping.

4. Pre-Processing

Raw data, as is, is collection of images of Skin Cancer lesions that are of different shapes and sizes; possibly due to different origin or sources from which these images were collected. Keeping all the variations into consideration, preprocessing of some kind is essential to create a model that accurately accounts for the variations of the input data [7]. Some of these pre-processing is absolutely necessary such as making the images of the same size. This is mainly for planning the number of layers in the CNN architecture, as it is necessary to have a fixed input shape before defining the architecture of the model. On the contrary, some pre-processing such as normalization and augmentation is done to improve overall model performance. Although normalization may not be an absolute necessity, it is one of the most important techniques which improves learning speed, making the model drastically easier to train by achieving faster convergence. It mainly involves ensuring that the data distribution in each pixel (or input parameter in general) is similar. In other words, it involves rescaling the pixel values to make them lie within a specified range. In addition to, data augmentation is another creative pre-processing technical choice to ensure that the model is trained with enough variations in the training data. It involves creating perturbed versions of the images by flipping, rotating, contrasting, or adding gaussian noise to expose the model to a wide variety of images. Not only does this significantly improve the accuracy of the model, but it also tackles the increased invariance introduced during the pooling process [11].

All the aforementioned pre-processing techniques were used before developing the CNN architecture to handle the classification problem for skin cancer more effectively and to achieve better model performance by ensuring that there is not a significant gap between training and validation accuracy. Particularly, randomized flipping, rotating, contrasting, and adding gaussian noise were implemented on the data. The contrast scale that was used was between 0-1 to access the accuracy between training and validation set.

5. System Design

5.1. Proposed System Model

Table 1: Details of the proposed CNN model

Layers (Type)	Output Shape	Parameters
Input Image	(224, 224, 3)	0
Convolution	(224, 224, 16)	448
ReLU	(224, 224, 16)	0
Max Pooling	(112, 112, 16)	0
Convolution	(112, 112, 32)	4640
ReLU	(112, 112, 32)	0
Max Pooling	(56, 56, 32)	0
Convolution	(56, 56, 64)	18496
ReLU	(56, 56, 64)	0
Max Pooling	(28, 28, 64)	0
Dropout	(28, 28, 64)	0
Flatten	(50176)	0
Dense (ReLU)	(128)	6422656
Dense (Softmax)	(2)	258

Total Number of Parameters = 6,446,498

Trainable Parameters = 6,446,498

Non-trainable Parameters = 0

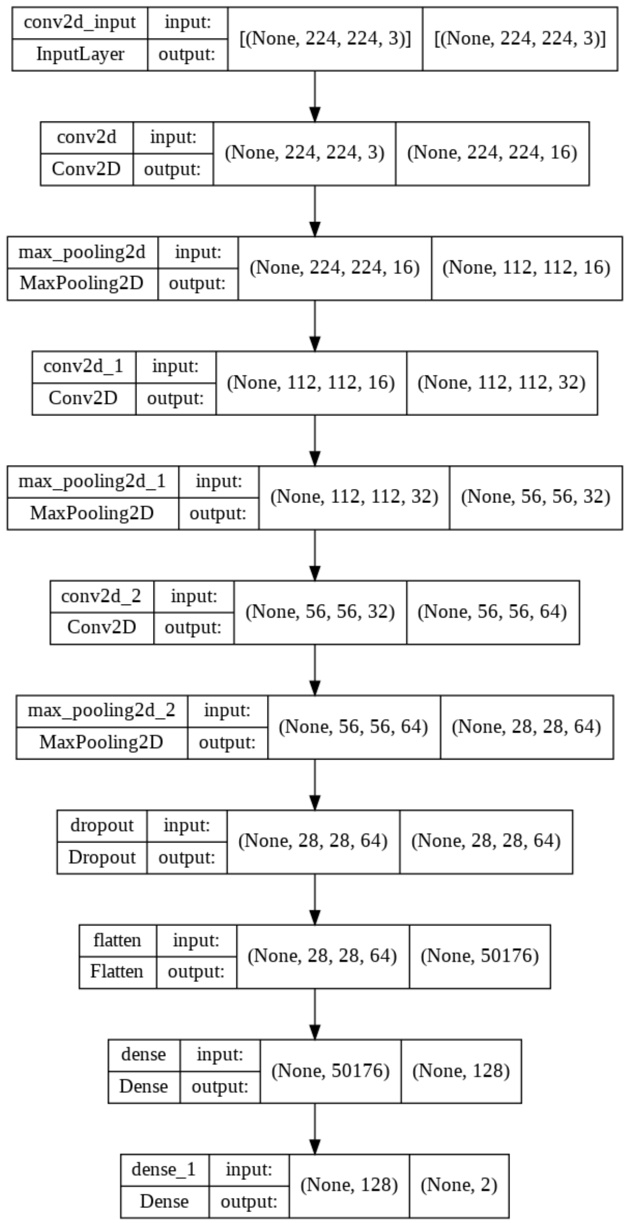

Figure 2: Summary of proposed CNN architecture

A summary of CNN architecture in terms of number of layers, output shape, and parameters is given in Table 1 and Figure 2. The input resolution of the images is reshaped to be 224 x 224 x 3. There are 3 hidden layers with a kernel size of 3 x 3 in each layer and a stride of 1. A ReLU activation along with max pooling is used at each hidden layer which reduces the image size as observed from the output shape in Table 1. This is followed by the flattening process to convert the 3D image into a 1D array of vectors. Finally, a Softmax activation function is used to classify the skin cancer images into benign or malignant (2 classes).

5.2. Sample Output

A sample output with the classified images from the test data is shown in figure 3.

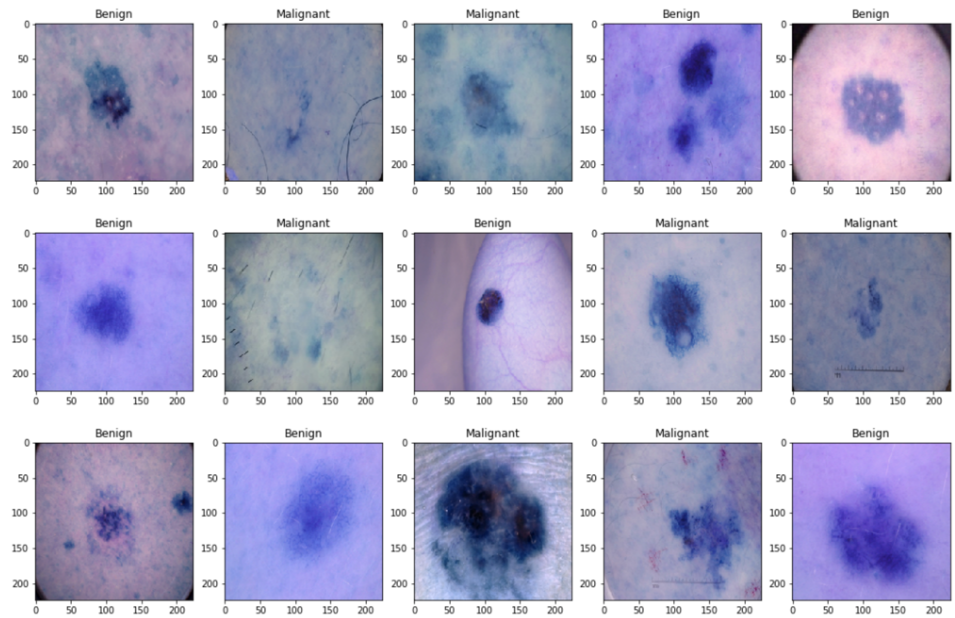

Figure 3: Sample classification from test data

5.3. System Performance

The system performance is measured in terms of accuracy, precision, recall, and F1 scores obtained using the confusion matrix. The following equations are used to calculate these values:

$$\text{A = Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

$$\text{P = Precision} = \frac{TP}{TP + FP}$$

$$\text{R = Recall} = \frac{TP}{TP + FN}$$

$$\text{F1 Score} = \frac{2 \cdot P \cdot R}{P + R}$$

Here, TP stands for True Positive, TN stands for True Negative, FP stands for False Positive, and FN stands for False Negative. TP is when the data is positive, and a correct prediction is made by CNN as positive. TN is when the data is negative, and a correct prediction is made by CNN as negative. On the contrary, FP is when the data is negative, but it is being predicted as positive, and FN is when the data is positive, but it is being predicted as negative. Altogether, these parameters are used to evaluate system performance for the classification problem in hand.

6. Results

In this study, a set of 1800 benign and 1497 malignant images are used for the training and validation. The data is split into 80% for training and 20% for validation. This makes a distribution of 2637 training images and 660 validation images. The data will be preprocessed with various combinations of color and image augmentation. The performance for each test is reported based on the training and validation accuracy. The first experiment passes only the normalized images through the baseline model and industry standard Convolutional Neural Networks. The comparison between well documented networks and the baseline model is needed to show the relative accuracy for the baseline model for the given dataset. Figure 4 shows the F1 accuracy of each model.

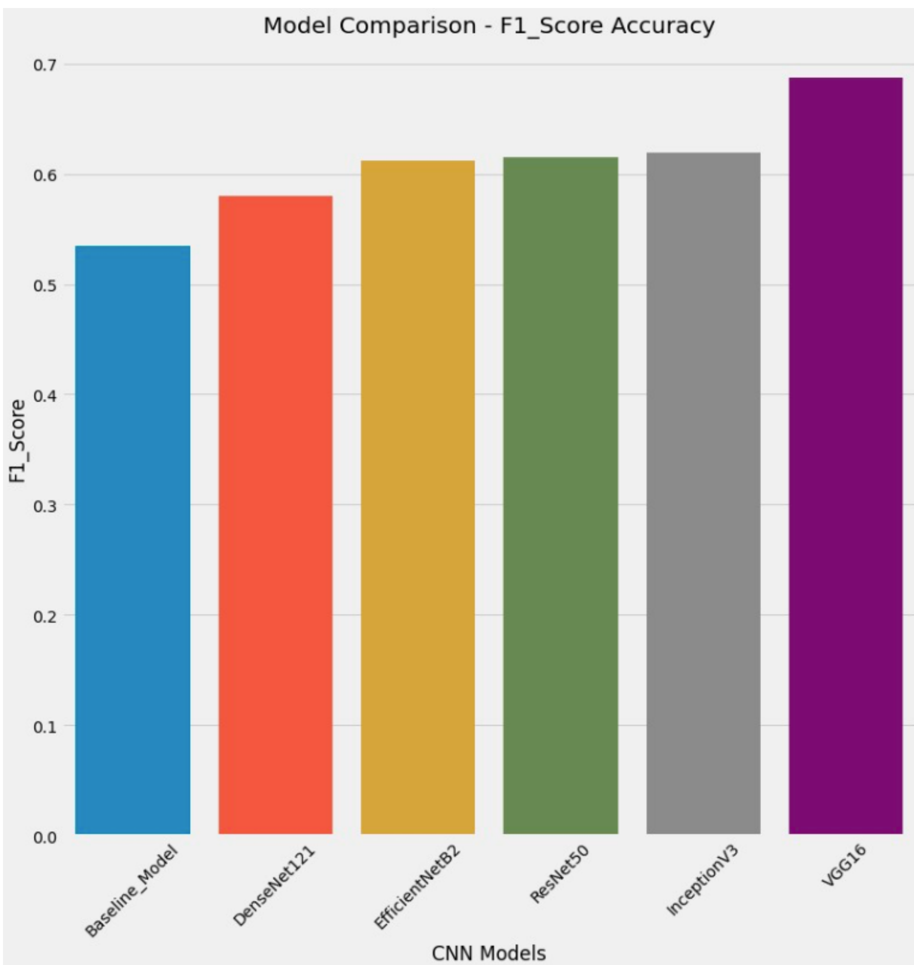

Figure **4**: F1 Accuracy for Baseline Model and Industry Standard CNNs

As shown in figure 4 the deeper networks outperform the shallower baseline model. However, the larger networks take longer to run and will not be used for the remaining experiments due to GPU limitations. Figure 5 depicts the training and testing accuracy for the baseline model.

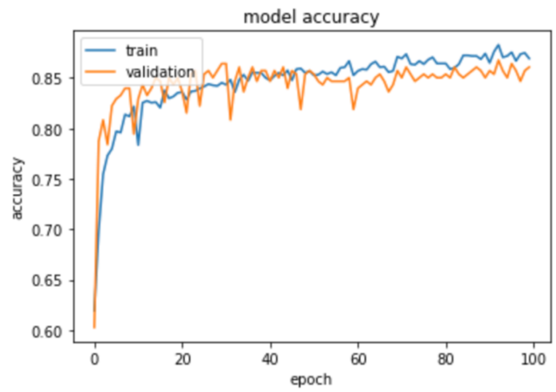

Figure 5: Baseline Model Training graph for Raw Data

In Figure 5 the validation and training accuracy are closely entwinned, but the accuracy only reaches ~86%. It is obvious the model could be made deeper like many premade models to increase the accuracy but that would cost computation time. Without changing the baseline model, the best way to improve the validation accuracy was to preprocess the images. The techniques used to preprocess the images were Flipping, Rotating, Contrasting and Gaussian Noise Addition.

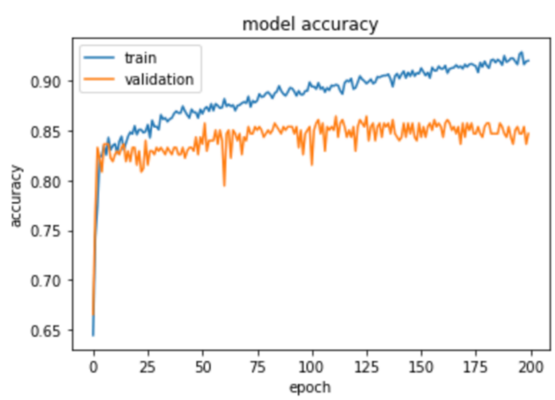

Figure 6: Flipped and Rotated Preprocessing on Baseline Model

Flipping and rotating the data increased the training data but decreased the validation data as shown in Figure 6. While the training accuracy did increase, the variance with the validation data is too great to deem flipping and rotating the data as a valuable preprocessing technique for this data set.

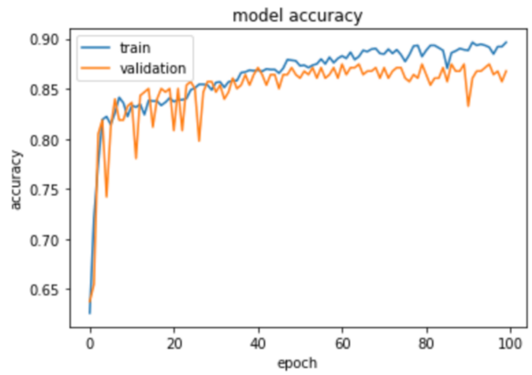

Figure 7: 0.2 Random Contrasting Preprocessing on Baseline Model

Adding the small contrast change to the input images resulted in an increased validation and training accuracy as shown in figure 7. The low contrasting acts to reduce the effect of external features in the view by softening the edges and creating a soft image [5].

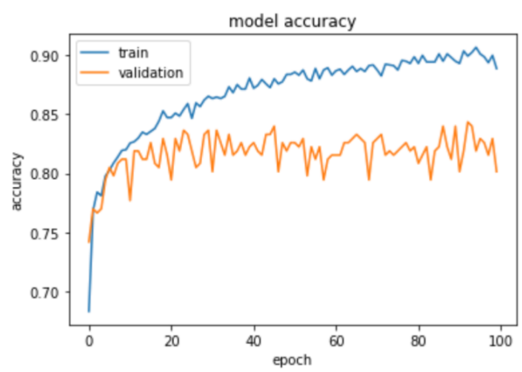

Figure 8: 0.5 Random Contrasting Preprocessing on Baseline Model

When a medium contrast is applied to the data, the validation and training accuracy diverge. During a medium contrast, the image boundaries will not be as clear and therefore would reduce the accuracy as seen in figure 8. This divergence is a sign of overfitting in the model.

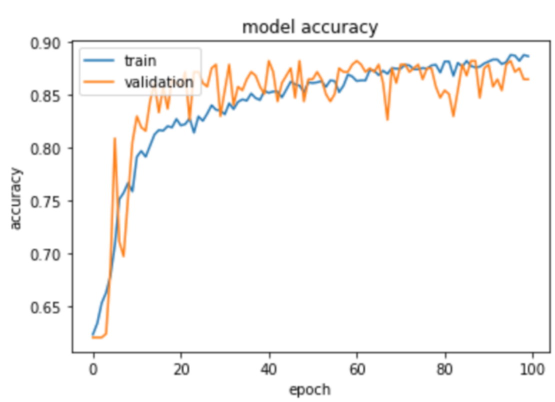

Figure 9: 0.9 Random Contrasting Preprocessing on Baseline Model

When the dataset is subject to a high contrast, the images will develop sharp boundaries around color differences. As seen in figure 9 the result is a validation curve that follows the training accuracy but has increased variation per epoch. The shaper boundaries likely improve the classification of lesions with few visual obstructions. If a hair of similar color interrupts the lesions boundary the model could misclassify it.

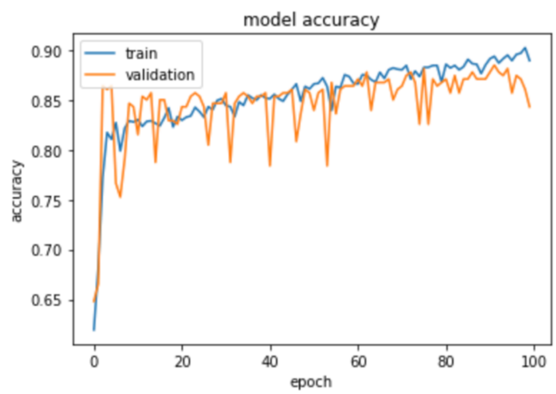

Figure 10: 0.01 Gaussian Noise Addition Preprocessing on Baseline Model

Figure 10 shows the addition of gaussian noise to the raw dataset. The addition of gaussian noise to the dataset added large variations over the training span and as such was disregarded as a valuable preprocessing technique.

Table 2: Training and Accuracy Results after 100 Epochs

	Training Accuracy (%)	Validation Accuracy (%)
No Preprocessing	86.92	86.06
Flip and Rotate	91.98	84.67
0.2 Contrast	89.64	86.76
0.5 Contrast	88.83	80.14
0.9 Contrast	90.25	83.62
0.01 Gaussian	88.96	84.32

As shown in table 1, the 0.2 contrast preprocessing raised the training and validation accuracy of the CNN compared to the base value. While the improvement is small, it is an improvement made without any additional layers of the CNN. Additional tests were performed with the preprocessing techniques paired together and the result was overtrained models.

7. Conclusion

Using convolutional neural networks to diagnose if a lesion is cancerous allows patients to rapidly test at home with the snap of a picture. This at home test would reduce the workload of doctors having to diagnose every patient. The CNN model used contains three convolutional layers a 3x3 kernel size, 3 maximum pooling layers, a dropout layer and two fully connected layers with ReLU and Softmax activation functions. The data used was from the ISIC public archive where it was experimentally preprocessed. Preprocessing the images is important to increase the accuracy of the model. By adding a low contrast to the data, the borders are softened to create a flat image capable of reducing the effect of other features like hair. Overall, preprocessing the data can lead to improved accuracy at no cost to the depth of the CNN model but tends to cause overfitting.

8. Bibliography

[1] “Skin Cancer.” *American Academy of Dermatology*, https://www.aad.org/media/stats-skin-cancer#:~:text=Mortality%20rates,cancer%20deaths%20are%20from%20melanoma.&text=Nearly%2020%20Americans%20die%20from,4%2C600%20men%20and%202%2C580%20women.

[2] Simon, Stacy. “How to Spot Skin Cancer.” *American Cancer Society*, American Cancer Society, 9 Apr. 2020, https://www.cancer.org/latest-news/how-to-spot-skin-cancer.html#:~:text=Redness%20or%20new%20swelling%20beyond,of%20a%20lump%20or%20bump.

[3] “Isic Archive.” ISIC Archive, https://www.isicarchive.com/#!/topWithHeader/wideContentTop/main.

[4] Esteva, Andre, et al. “Erratum: Corrigendum: Dermatologist-Level Classification of Skin Cancer with Deep Neural Networks.” *Nature*, vol. 546, no. 7660, 2017, pp. 686–686., https://doi.org/10.1038/nature22985.

[5] “Understanding Contract in Photography.” *Skylum Blog*, https://skylum.com/blog/understanding-contrast-in-photography.

[6] Beeravolu, Abhijith Reddy et al. "Preprocessing Of Breast Cancer Images To Create Datasets For Deep-CNN". IEEE Access, vol 9, 2021, pp. 33438-33463. Institute Of Electrical And Electronics Engineers (IEEE), https://doi.org/10.1109/access.2021.3058773. Accessed 25 Apr 2022.

[7] Beeravolu, Abhijith Reddy et al. "Preprocessing Of Breast Cancer Images To Create Datasets For Deep-CNN". IEEE Access, vol 9, 2021, pp. 33438-33463. Institute Of Electrical And Electronics Engineers (IEEE), https://doi.org/10.1109/access.2021.3058773. Accessed 25 Apr 2022.

[8] Hubel, D. H., and T. N. Wiesel. "Receptive Fields And Functional Architecture Of Monkey Striate Cortex". The Journal Of Physiology, vol 195, no. 1, 1968, pp. 215-243. Wiley, https://doi.org/10.1113/jphysiol.1968.sp008455.

[9] Fukushima K (1980) Neocognitron: a self organizing neural network model for a mechanism of pattern recognition unaffected by shift in position.

[10] Hidaka, Akinori, and Takio Kurita. "Consecutive Dimensionality Reduction By Canonical Correlation Analysis For Visualization Of Convolutional Neural Networks". Proceedings Of The ISCIE International Symposium On Stochastic Systems Theory And Its Applications, vol 2017, no. 0, 2017, pp. 160-167. The Institute Of Systems, Control And Information Engineers, https://doi.org/10.5687/sss.2017.160.

[11] Dayanand, Tanya. "Data Preprocessing And Network Building In CNN". Towards Data Science, , Medium, 2020, https://towardsdatascience.com/data-preprocessing-and-network-building-in-cnn-15624ef3a28b.

[12] Tiwari, Sakshi, and Akanksha Rai. "Activation Functions In Neural Networks". Geeksforgeeks, 2020, https://www.geeksforgeeks.org/activation-functions-neural-networks/.

[13] P.Kim. “Deep Learning: With Machine Learning, Neural Networks, and Artificial Intelligence.” MATLAB, 2017.

[14] Fu’adah, Yunendah Nur et al. "Convolutional Neural Network (CNN) For Automatic Skin Cancer Classification System". IOP Conference Series: Materials Science And Engineering, vol 982, no. 1, 2020, p. 012005. IOP Publishing, https://doi.org/10.1088/1757-899x/982/1/012005.

[15] Yamashita, Rikiya et al. "Convolutional Neural Networks: An Overview And Application In Radiology". Insights Into Imaging, vol 9, no. 4, 2018, pp. 611-629. Springer Science And Business Media LLC, https://doi.org/10.1007/s13244-018-0639-9. Accessed 25 Apr 2022.