Analysis of Skin Lesion Classification using Transfer Learning Models

Authors
Viswajith Menon (vm623)
Zeeshan Ahsan (za224)
Kevin Paul (kc1368)

Abstract

Skin cancer is one of the most common types of cancer, and early detection is critical for successful treatment. However, manual analysis of skin lesions by dermatologists can be time-consuming and subjective. Therefore, an automated classification system can assist dermatologists in their diagnosis and improve patient outcomes.

Our project aims to use the benchmark dataset Ham10000 which consists of skin lesion images to conduct a statistical analysis under conditions where the dataset contains corrupt/noisy images and build models that obtains accurate results of image classification.

Using noisy images in the implementation of Skin Lesion classification is important because it helps to improve the model's robustness and generalization performance. In the real world, images of skin lesions may contain various types of noise, such as blur, distortions, artifacts, and other types of interference.

By training the classification model on noisy images, we can make it more resistant to these types of noise and better able to generalize to unseen data. In this project, pre-trained models such as ResNet50, VGG16, DenseNet121, VGG19 and ResNet152 are used as classifiers on the benchmark HAM10000 dataset and are evaluated through various performance metrics. The models obtained the accuracies .87, .88, .89, .84 and .90 accordingly.

1 Data Preprocessing

1.1 Dataset

The dataset used is the publicly available benchmark dataset known as HAM10000(Human Against Machine with 10000 training images). This consists of dermatoscopic images from different populations, acquired and stored by different modalities. The final dataset consists of 10015 dermatoscopic images which can serve as a training set for deep learning purposes.

The images are a representative collection of all important diagnostic categories in the realm of pigmented lesions. These are classified into 7 types of skin lesion namely • Melanocytic nevus (nv) • Actinic keratosis (akiec) • Basal cell carcinoma (bcc) • Dermatofibroma (df) • Vascular lesion (vasc) • Malignant melanoma (mel) • Benign keratosis (bkl).

The images have been standardized to a 224x224x3 input size RGB image.

1.2 Dataset Augmentation

The dataset is highly imbalanced with Melanocytic nevus having the most images. In order to balance the dataset, several data augmentation techniques like flipping, rotating and tilting have been implemented.

Dataset before Augmentation -

Total Images	akiec	bcc	bkl	df	mel	nv	vas
10015	297	479	1067	107	1011	5822	129

Dataset After Augmentation –

Total Images	akiec	bcc	bkl	df	mel	nv	vas
35967	4455	4790	5055	5350	5335	5822	5160

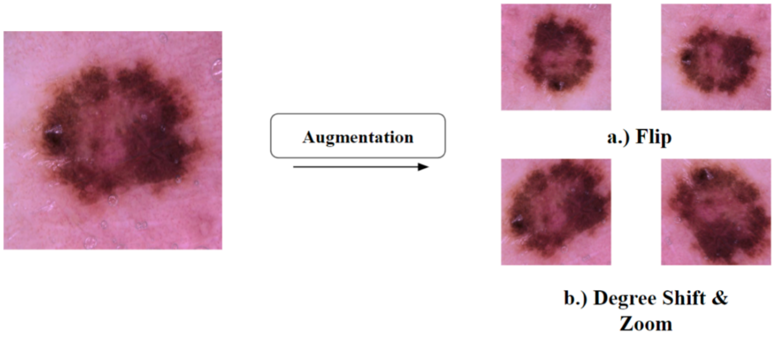


Figure 1: Augmentation

Flip (Mirroring) as seen in Fig 1(a), Flip top and bottom: In this operation, the image will be flipped along the horizontal axis. Probability of 0.5 is used which signifies 50% of the samples will be flipped randomly. Flip left and right: In this operation, the image will be flipped along the vertical axis. Here also probabilty of 0.5 is used for flipping the samples of dataset.

Degree Shift and Zoom is done as seen in Fig 1(b), This will rotate by arbitrary degrees, then a crop is taken from the centre of the newly rotated image. Image is rotated by an arbitrary angle in range -10 to 10 degree with a probability of 0.5 which signifies 50 percent of the sample images from the dataset will be rotated randomly. Zoom effect is not particularly drastic for smaller rotations of between - 10 and 10

1.3 Normalization of Input Images

In image processing, it is recommended to normalize image pixel values relative to the dataset mean and standard deviation. This helps to get consistent results when applying a model to new images and can also be useful for transfer learning. Normalization is done to ensure better performance of the neural network. Normalization is an important step to achieve faster convergence. With unnormalized data, numerical ranges of features may vary strongly. If this raw data is inputted in the transfer learning model, slow convergence will occur. The dataset is normalized in z-score normalization as per given formula:

where, Vi'= z-score normalized values

Vi is the value of row E of i'th column.

$$V_i' = \frac{V_i - \bar{E}}{std(E)} \qquad std\left(E\right) = \sqrt{\frac{1}{(n-1)} \sum_{i=1}^{n} \left(V_i - E\right)^2}$$

$$E = \frac{1}{(n)} \sum_{i=1}^{n} V_i$$

Figure 2: Formulae used during Normalization

The steepest gradient is searched, which is somewhat in the correct direction but also possesses quite a large oscillation. This can be explained by the value of the chosen learning rate. A relatively large learning rate is required for the features with larger standard deviation since its range is quite large. However, this large learning rate is too large for some smaller parameters. The optimizer overshoots each step, which results in oscillation and hence slow convergence. In this project, for the HAM10000 dataset, the Mean was found to be (0.49139968, 0.48215827, 0.44653124) and standard deviation was found to be (0.24703233, 0.24348505, 0.26158768) for the RGB Channel.

1.4 Corruption of Images

Dermoscopy enables magnified visualization of subcuticular features invisible to the naked eye. The quality of images, which are affected by light, angle and other distortions has a strong impact on the classification accuracy. Captured images are subject to various types of distortion, such as illumination variations, motion blur, and defocus aberrations. In technical terms, these are called “noise”. Photos are inevitably contaminated by noise, mainly a compound of Gaussian and Poisson noise. Additionally, there could be salt and pepper noise. Removing noise may increase the diagnostic accuracy. Thus, to achieve better diagnosis results it is necessary to address the effects of image noise on the deep Convolutional Neural Network (CNN) classification of skin lesions.

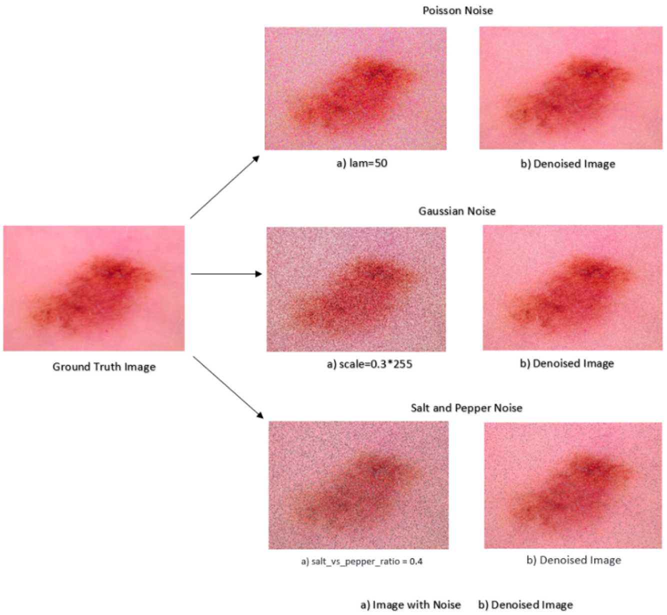


Figure 3: Image Corruption

1.4.1 Noising

In this project, gaussian noise is manually added to corrupt the images. Principal sources of Gaussian noise in digital images arise during acquisition e.g. sensor noise caused by poor illumination and/or high temperature, and/or transmission e.g. electronic circuit noise. To add gaussian noise to an image, sampled once per pixel from a normal distribution N(0, s), where s is sampled per image and varies between 0 and 0.2*255. In this project, the sampling scale is chosen as 0.1*255. The probability density function z is given by:

$$(z) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(z-\mu)^2}{2\sigma^2}}$$

Figure 4: Noising

where z represents the grey level, μ represents the mean grey value and sigma is standard deviation.

1.4.2 Denoising

Denoising of an image refers to the process of reconstruction of a signal from noisy images. Denoising is done to remove unwanted noise from image to analyze it in better form. It refers to one of the major pre-processing steps. There are metrics to test how well the denoising techniques work by measuring how much noise is remaining currently as well. They are included below.

PSNR and MSE The PSNR block computes the peak signal-to-noise ratio, in decibels, between two images. This ratio is used as a quality measurement between the original and a compressed image. The higher the PSNR, the better the quality of the compressed, or reconstructed image.

The mean-square error (MSE) as shown in the first Eq below and the peak signal-to-noise ratio (PSNR) as shown in the second Eq below are used to compare image compression quality. The MSE represents the cumulative squared error between the compressed and the original image, whereas PSNR represents a measure of the peak error. The lower the value of MSE, the lower the error.

$$MSE = \sum_{M,N} \frac{[I1(m,n) - I2(m,n)]^2}{M * N}$$

$$PSNR = 10 \log_{10} \frac{R^2}{MSE}$$

Figure 5: MSE & PSNR

SSIM The Structural Similarity Index (SSIM) is a perceptual metric that quantifies image quality degradation caused by processing such as data compression or by losses in data transmission. It is a full reference metric that requires two images from the same image capture— a reference image and a processed image. The processed image is typically compressed. It may, for example, be obtained by saving a reference image as a JPEG (at any quality level) then reading it back in. SSIM is best known in the video industry, but has strong applications for still photography. Any image may be used, including those of Imatest test patterns such as Spilled Coins or Log F Contrast.

SSIM as seen in Eq below actually measures the perceptual difference between two similar images. It cannot judge which of the two is better: that must be inferred from knowing which is the “original” and which has been subjected to additional processing such as data compression. Unlike PSNR (Peak Signal-to-Noise Ratio), SSIM is based on visible structures in the image.

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

Figure 6: SSIM

2 System Architecture

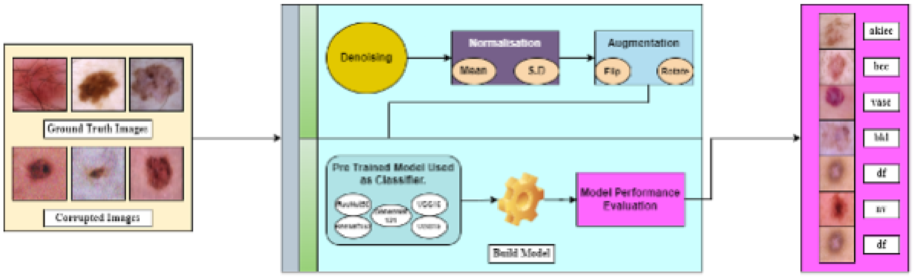


Figure 7: System Architecture

As seen in Fig 7, The benchmark dataset undergoes preprocessing methods such as denoising, normalization and augmentation to get balanced training samples in each class. Some images are corrupt, so denoising process is implemented. Augmentation is done to create balanced classes. Once images are preprocessed, training and testing sets are prepared accordingly and model is fit and used as a classifier.

3 Models Used

We have used transfer learning in this project. For this we have used pre-trained models from PyTorch as classifiers. The pre-trained models used are as follows.

3.1 RESNET 50

Residual Network or ResNet is a convolutional neural network that was designed to use hundreds or thousands of convolutional layers. ResNet was proposed by He et al[45]. ResNet solved the vanishing gradient problem which causes the degradation of results using ’skip connections’ and due to this connection, information from one layer can directly be fed to another layer by skipping some intermediate layers. In that way, a deeper layer can have information both from just it’s the previous layer and from other layers as well. ResNet also incorporates batch normalization. This network is 50 layers deep.

3.2 RESNET 152

ResNet 152 presents a residual learning framework to ease the training of networks that are substantially deeper than those used previously. It explicitly reformulates the layers as learning residual functions with reference to the layer inputs,instead of learning unreferenced functions. It provides comprehensive empirical evidence showing that these residual networks are easier to optimize, and can acheive considerable accuracy when the depth of the network is increased. This particular residual network is 152 layers in depth.

3.3 DENSENET 121

DenseNet is one of the top performing architectures on the ImageNet dataset. To improve the information flow between layers, there is a different connection pattern where each layer recieves inputs from all preceding layers as feature-maps. DenseNet concatenates features instead of summing like ResNet and in that way it keeps the difference between added information and preserved information. This concatenation of feature-maps raise variations in the input of different layers and thus enhance performance.

3.4 VGG 16

VGGNet is a well documented and commonly used architecture for convolutional neural networks. This ConvNet became popular by achieving excellent performance on the ImageNet dataset. The input layer of the network expects a 224×224 pixel RGB image. The input image is passed through five convolutional blocks. Small convolutional filters with a receptive field of 3×3 are used. Each convolutional block includes a 2D convolution layer operation (the number of filters changes between blocks). All hidden layers are equipped with a ReLU (Rectified Linear Unit) as the activation function layer (nonlinearity operation) and include spatial pooling through use of a max-pooling layer. The network is concluded with a classifier block consisting of three fully connected layers.The architecture has 13 convolutional layers followed by 3 fully connected layers, adding up to 16 layers to learn weights and bias parameters and hence the name VGG-16.

3.5 VGG 19

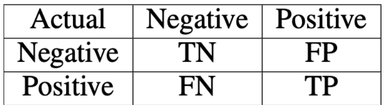
VGG-19 architecture is very much similar to VGG-16. It is a variant of VGG model which in short consists of 19 layers (16 convolution layers, 3 Fully connected layer, 5 MaxPool layers and 1 SoftMax layer). The rest of the architecture of this network follows the same norms as discussed in VGG-16.

4 Results and Performance Analysis

We have used metrics such as Confusion Matrix, Learning curves for loss and accuracy and Accuracy, Precision, Recall and F1 Score to evaluate the prformance of different networks under different conditions. A brief overview of this metrics is stated below.

4.1 Confusion Matrix

Confusion matrix is a very popular measure used while solving classification problems. It can be applied to binary classification as well as for multi-class classification problems. Confusion matrices represent counts from predicted and actual values.

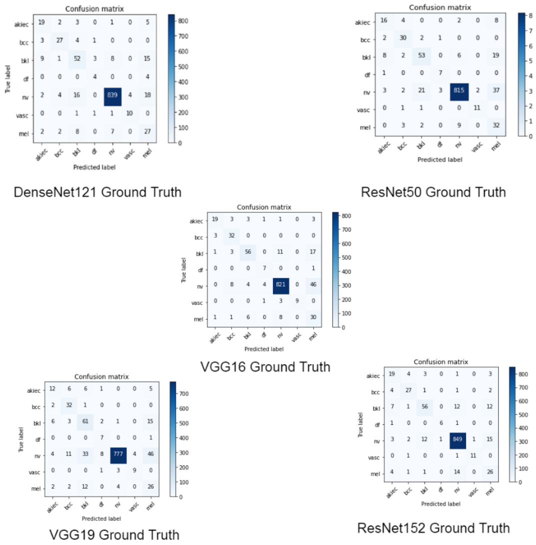


Actual	Negative	Positive
Negative	TN	FP
Positive	FN	TP

Figure 8: Confusion Matrix Classification

In the above figure "TN" stands for True Negative which shows the number of negative examples classified accurately. Similarly, "TP" stands for True Positive which indicates the number of positive examples classified accurately. The term "FP" shows False Positive value, i.e., the number of actual negative examples classified as positive; and "FN" means a False Negative value which is the number of actual positive examples classified as negative.

The results for the confusion matrix for ground truth images are as follows:



DenseNet121 Ground Truth

ResNet50 Ground Truth

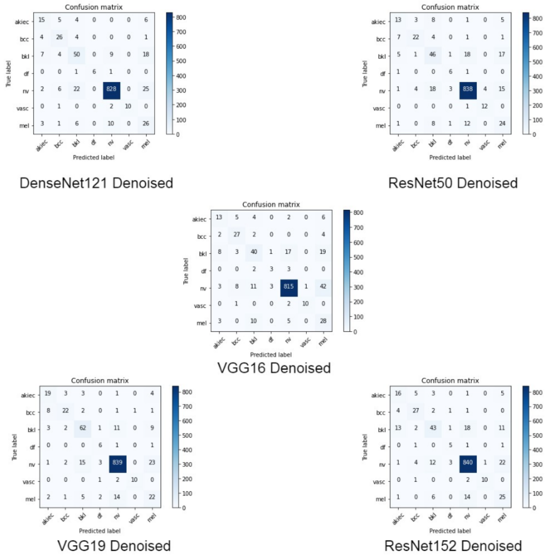
VGG16 Ground Truth

VGG19 Ground Truth

ResNet152 Ground Truth

Figure 9: Confusion Matrix of Ground Truth Images

The results for the confusion matrix for denoised images are as follows:



DenseNet121 Denoised

ResNet50 Denoised

VGG16 Denoised

VGG19 Denoised

ResNet152 Denoised

Figure 10: Confusion Matrix of Denoised Images

4.2 Learning Curves For Loss and Accuracy

A loss function is used to optimize a machine learning algorithm. The loss is calculated on training and validation data-sets and its interpretation is based on how well the model is doing in these two sets. It is the sum of errors made for each example in training or validation sets. Loss value implies how poorly or well a model behaves after each iteration of optimization.

An accuracy metric is used to measure the algorithm's performance. The accuracy of a model is usually determined after the model parameters are optimized and is expressed in the form of a percentage. It is the measure of how accurate the model's prediction is compared to the true data.

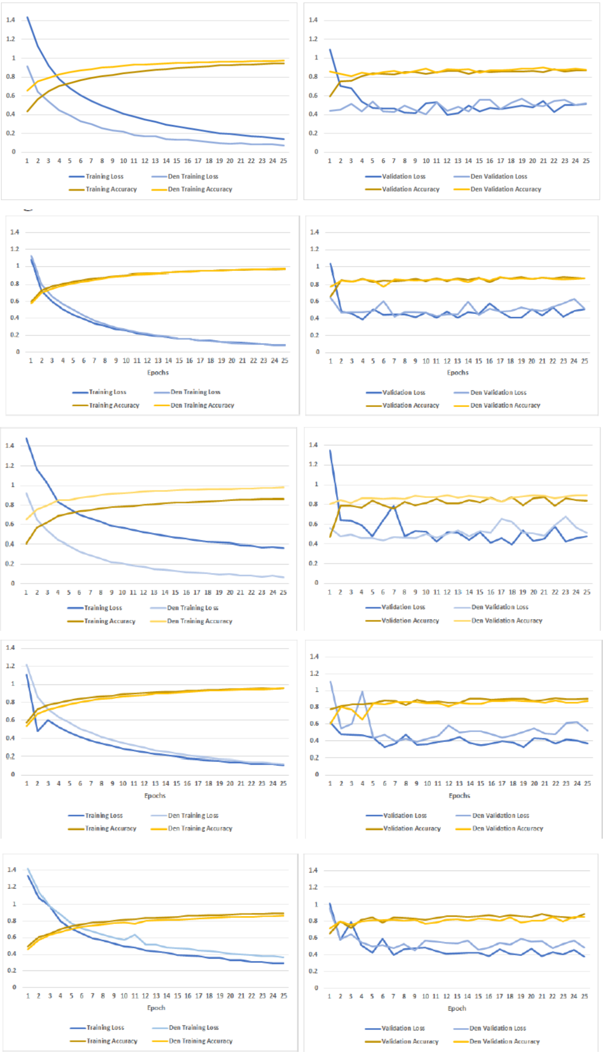


Figure 11: Ground Truth vs Denoised Curves for DensetNet121, ResNet50, VGG19, ResNet152, VGG16 Respecvtively

4.3 Accuracy, Precision, Recall and F1 Score

Performance metrics in the project's classifications report include accuracy, precision, recall, and F1 score, which are calculated on the basis of TP, TN, FP, and FN.

Accuracy of an algorithm is represented as the ratio of correctly classified patients (TP+TN) to the total number of patients (TP+TN+FP+FN).

Accuracy = (TP + TN) / (TP + TN + FP + FN)

Precision is defined as:

Precision = TP / (TP + FP)

Recall is defined as:

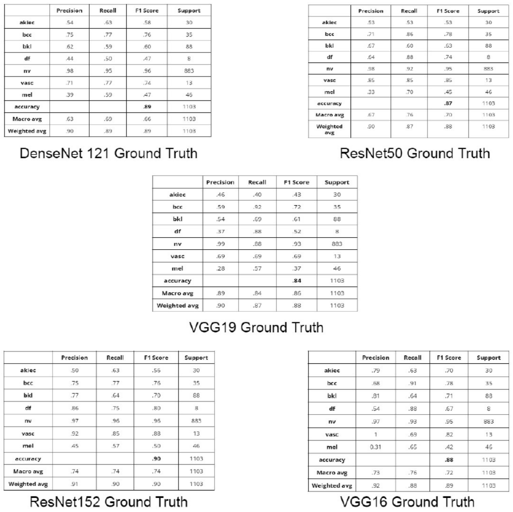
Recall = TP / (TP + FN)

F1 score is defined as:

F1 score = (2 * Precision * Recall) / (Precision + Recall)

The F1 score states the equilibrium between the precision and the recall.

The classification report for the ground truth obtained from the models are:



	Precision	Recall	F1 Score	Support
akiec	.54	.63	.58	30
bcc	.75	.77	.76	35
bkl	.62	.59	.60	88
df	.44	.50	.47	8
nv	.98	.95	.96	883
vasc	.71	.77	.74	13
mel	.39	.59	.47	46
accuracy			**.89**	1103
Macro avg	.63	.69	.66	1103
Weighted avg	.90	.89	.89	1103

DenseNet 121 Ground Truth

	Precision	Recall	F1 Score	Support
akiec	.53	.53	.53	30
bcc	.71	.86	.78	35
bkl	.67	.60	.63	88
df	.64	.88	.74	8
nv	.98	.92	.95	883
vasc	.85	.85	.85	13
mel	.33	.70	.45	46
accuracy			**.87**	1103
Macro avg	.67	.76	.70	1103
Weighted avg	.90	.87	.88	1103

ResNet50 Ground Truth

	Precision	Recall	F1 Score	Support
akiec	.46	.40	.43	30
bcc	.59	.92	.72	35
bkl	.54	.69	.61	88
df	.37	.88	.52	8
nv	.99	.88	.93	883
vasc	.69	.69	.69	13
mel	.28	.57	.37	46
accuracy			**.84**	1103
Macro avg	.89	.84	.86	1103
Weighted avg	.90	.87	.88	1103

VGG19 Ground Truth

	Precision	Recall	F1 Score	Support
akiec	.50	.63	.56	30
bcc	.75	.77	.76	35
bkl	.77	.64	.70	88
df	.86	.75	.80	8
nv	.97	.96	.96	883
vasc	.92	.85	.88	13
mel	.45	.57	.50	46
accuracy			**.90**	1103
Macro avg	.74	.74	.74	1103
Weighted avg	.91	.90	.90	1103

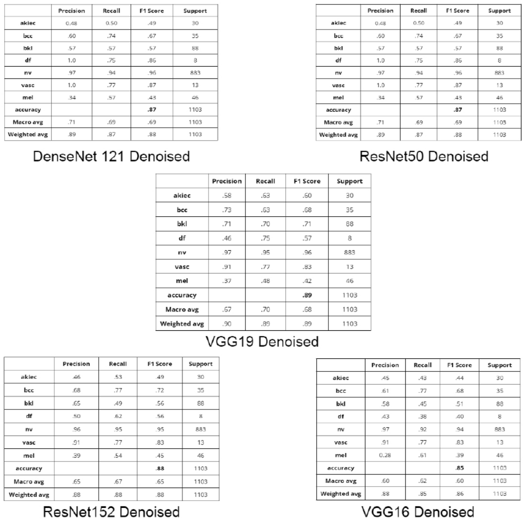
ResNet152 Ground Truth

	Precision	Recall	F1 Score	Support
akiec	.79	.63	.70	30
bcc	.68	.91	.78	35
bkl	.81	.64	.71	88
df	.54	.88	.67	8
nv	.97	.93	.95	883
vasc	1	.69	.82	13
mel	0.31	.65	.42	46
accuracy			**.88**	1103
Macro avg	.73	.76	.72	1103
Weighted avg	.92	.88	.89	1103

VGG16 Ground Truth

Figure 12: Classification Report for Ground Truth Images

The classification report for denoised images are:



	Precision	Recall	F1 Score	Support
akiec	0.48	0.50	.49	30
bcc	.60	.74	.67	35
bkl	.57	.57	.57	88
df	1.0	.75	.86	8
nv	.97	.94	.96	883
vasc	1.0	.77	.87	13
mel	.34	.57	.43	46
accuracy			**.87**	1103
Macro avg	.71	.69	.69	1103
Weighted avg	.89	.87	.88	1103

DenseNet 121 Denoised

	Precision	Recall	F1 Score	Support
akiec	0.48	0.50	.49	30
bcc	.60	.74	.67	35
bkl	.57	.57	.57	88
df	1.0	.75	.86	8
nv	.97	.94	.96	883
vasc	1.0	.77	.87	13
mel	.34	.57	.43	46
accuracy			**.87**	1103
Macro avg	.71	.69	.69	1103
Weighted avg	.89	.87	.88	1103

ResNet50 Denoised

	Precision	Recall	F1 Score	Support
akiec	.58	.63	.60	30
bcc	.73	.63	.68	35
bkl	.71	.70	.71	88
df	.46	.75	.57	8
nv	.97	.95	.96	883
vasc	.91	.77	.83	13
mel	.37	.48	.42	46
accuracy			**.89**	1103
Macro avg	.67	.70	.68	1103
Weighted avg	.90	.89	.89	1103

VGG19 Denoised

	Precision	Recall	F1 Score	Support
akiec	.46	.53	.49	30
bcc	.68	.77	.72	35
bkl	.65	.49	.56	88
df	.50	.62	.56	8
nv	.96	.95	.95	883
vasc	.91	.77	.83	13
mel	.39	.54	.45	46
accuracy			**.88**	1103
Macro avg	.65	.67	.65	1103
Weighted avg	.88	.88	.88	1103

ResNet152 Denoised

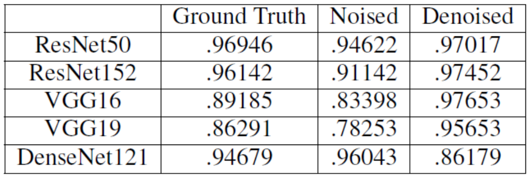
	Precision	Recall	F1 Score	Support
akiec	.45	.43	.44	30
bcc	.61	.77	.68	35
bkl	.58	.45	.51	88
df	.43	.38	.40	8
nv	.97	.92	.94	883
vasc	.91	.77	.83	13
mel	0.28	.61	.39	46
accuracy			**.85**	1103
Macro avg	.60	.62	.60	1103
Weighted avg	.88	.85	.86	1103

VGG16 Denoised

Figure 13: Classification Report for Denoised Images

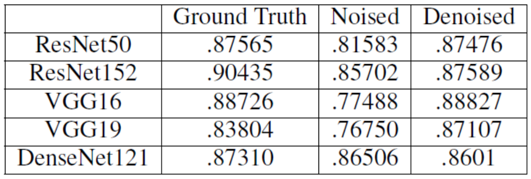
4.4 Accuracy of Models

The training and validation accuracy for different models under various conditions are as follows:



	Ground Truth	Noised	Denoised
ResNet50	.96946	.94622	.97017
ResNet152	.96142	.91142	.97452
VGG16	.89185	.83398	.97653
VGG19	.86291	.78253	.95653
DenseNet121	.94679	.96043	.86179

Figure 14: Training accuracy of models



	Ground Truth	Noised	Denoised
ResNet50	.87565	.81583	.87476
ResNet152	.90435	.85702	.87589
VGG16	.88726	.77488	.88827
VGG19	.83804	.76750	.87107
DenseNet121	.87310	.86506	.8601

Figure 15: Validation accuracy of models

5 Conclusion

The objectives of the project were to use transfer learning to train models on benchmark dataset with corruption. The corrupted dataset undergoes effectual preprocessing of denoising potential noisy images. To provide analysis with the help of graphs on model performance and help evaluate the best performing model is a key part of the project.

The conclusion of the project is that transfer learning is a useful technique to adopt when performing statistical analysis or comparison of pre trained model performance. Once images are preprocessed and dataset is balanced, Transfer learning proves to be extremely useful. In spite of corrupting dataset in terms of aberrations such as Gaussion Noise, The images have been successfully denoised using tools and functions present in various deep learning libraries.

For the future, there is scope to finetune CNN layers manually and customize it to produce even better results. There is also scope in testing models with dataset in occluded conditions. Different methods like feature extraction and image segmentation can also be used to yield results.

Hence, the conclusion for this project is that the results obtained give a definitive solution to what model works best under these standard conditions and how to successfully denoise corruption from benchmark dataset.

To summarize ResNet152 performs best under ground truth conditions, DenseNet121 has highest accuracy under noisy conditions and after denoising the image VGG16 has the best results. These conclusions are drawn from the validation set of the dataset discussed in this project.

6 References

1. O.R.P. P´erez V.d.J. A.Yabor A.M.Y. Palomo, M.d.J.D. P´erez and A. M. Fontaine. Melanoma maligno cut´aneo en pacientes de la provincia de lastunas. Revista Electr´onica Dr. Zoilo E. Marinello Vidaurret, 40, 2015.

2. A. C. Society. Cancer facts and figures. 2016.

3. H. Su J. Krause S. Satheesh S. Ma Z. Huang A. Karpathy A. Khosla M.Bernstein et al. O. Russakovsky, J. Deng. Imagenet large scale visual recognition challenge. International Journal of Computer Vision, 2016.

4. Ammara Masood and Adel Al-Jumaily. Semi advised learning and classification algorithm for partially labeled skin cancer data analysis. In 2017 12th International Conference on Intelligent Systems and Knowledge Engineering (ISKE), pages 1–4. IEEE, 2017.

5. U˘gur Fidan, ˙Ismail Sarı, and Raziye K¨ubra Kumrular. Classification of skin lesions using ann. In 2016 Medical Technologies National Congress (TIPTEKNO), pages 1–4. IEEE, 2016.

6. Mobeen ur Rehman, Sharzil Haris Khan, SM Danish Rizvi, Zeeshan Abbas, and Adil Zafar. Classification of skin lesion by interference of segmentation and convolotion neural network. In 2018 2nd International Conference on Engineering Innovation (ICEI), pages 81–85. IEEE, 2018.

7. M Monisha, Alex Suresh, BR Tapas Bapu, and MR Rashmi. Classification of malignant melanoma and benign skin lesion by using back propagation neural network and abcd rule. Cluster Computing, 22(5):12897–12907, 2019.

8. Marwan Ali Albahar. Skin lesion classification using convolutional neural network with novel regularizer. IEEE Access, 7:38306–38313, 2019.

9. Andre Esteva, Brett Kuprel, Roberto A Novoa, Justin Ko, Susan M Swetter, Helen M Blau, and Sebastian Thrun. Dermatologist-level classification of skin cancer with deep neural networks. nature, 542(7639):115–118, 2017.

10. Fangfang Han, Huafeng Wang, Guopeng Zhang, Hao Han, Bowen Song, Lihong Li, William Moore, Hongbing Lu, Hong Zhao, and Zhengrong Liang. Texture feature analysis for computer-aided diagnosis on pulmonary nodules. Journal of digital imaging, 28(1):99–115, 2015.