IMPROVING CLASSIFICATION PERFORMANCE OF LESION DERMOSCOPY BY EXPLORING GENERATIVE ADVERSARIAL NETWORKS BASED DATA AUGMENTATION APPROACHES

SAUGATA DAS PURKAYASTHA

Final Thesis Report

JUNE 2022

Table of Contents

Dedication 4

Abstract 5

List of tables 6

List of figures 7

List of abbreviations 9

Chapter1: Introduction 10

1.1 Background of study 10

1.2 Problem statement 11

1.3 Aim and objectives 11

1.4 Scope of study 12

1.5 Significance of study 12

1.6 Structure of study 12

Chapter2: Literature Review 13

2.1 Introduction 13

2.2 Classification in Lesion Analysis 13

2.3 Generative Adversarial Network (GAN) 16

2.3.1 Exploring GAN architectures 17

2.3.1.1 Deep Convolutional GAN 17

2.3.1.2 Wasserstein GAN 17

2.3.1.3 Progressive GAN 18

2.3.1.4 StyleGAN 19

2.3.1.5 Conditional GAN 19

2.3.1.6 Laplacian Pyramid of Adversarial Networks 20

2.3.2 GAN in Medical Images 21

2.3.2.1 Mode Collapse 24

2.3.3 Summary and Challenges of GAN based approaches in medical images 26

Chapter3: Research Methodology 27

3.1 Introduction 27

3.2 Dataset 27

3.2.1 Exploratory Data Analysis 28

3.3 Research Flow 29

3.3.1 Classification performance baseline 29

3.3.2 Explore GAN architecture 29

3.3.3 Train the GAN models 30

3.3.4 Augment the dataset 30

3.3.5 Training the classification model on augmented dataset 30

3.4 Evaluation Metrics 31

Chapter4: Implementation 32
4.1 Introduction 32
4.2 Machine Learning Pipeline 32
4.3 DCGAN with ISIC dataset 33
4.4 PGGAN with ISIC dataset 36
4.5 Classification with VGG16 38
4.5.1 Custom Data Loader 39
4.6 Model parameters 40
Chapter 5: Results and Evaluation 41
5.1 Introduction 41
5.2 VGG16 experiments on ISIC dataset 41
5.2.1 VGG16 performance on ISIC dataset 41
5.2.2 VGG16 performance on augmented ISIC dataset 42
5.3 GAN experiments 43
5.3.1 DCGAN 43
5.3.2 PGGAN 45
5.4 VGG16 experiments on ISIC dataset augmented with GAN 48
5.4.1 Augmentation with 64x64x3 image from DCGAN 48
5.4.2 Augmentation with 256x256x3 image from PGGAN 49
Chapter 6: Conclusion and Recommendations 50
6.1 Introduction 50
6.2 Discussion and Conclusion 50
6.3 Contributions 50
6.4 Future Work 51
Appendix A: Research Proposal 55

Dedication

To my family for their unconditional support during my master's journey and to my thesis advisor Bikash Santra for his guidance.

Abstract

Skin melanoma is a dangerous form of skin cancer since it can spread to other areas of body if left untreated. About 8000 people die of melanoma each year and it is highest among other types of skin cancer. Early detection and treatment of skin melanoma is very critical. Different machine learning based classification methods have been employed to identify whether skin cancer type is melanoma or not. ISIC dataset, which is the primary dataset of skin cancer images have large imbalance. The melanoma images are much less compared to benign type. As a result, the machine learning classification algorithm has poor recall and precision performance. Traditionally imbalance is resolved with data augmentation technique such as affine transformation. In this research work we explored GAN based architecture to augment ISIC dataset and evaluated if it improved the classification performance.

List of tables

Table 2-1: Summary of popular GAN models in image classification 26
Table 3-1: Train.csv of ISIC dataset 28
Table 3-2: Test csv of ISIC dataset 28
Table 3-3: Confusion Matrix 31
Table 3-4: Classification evaluation 31
Table 4-1: Convolution parameters for DCGAN generator for image of 64x64x3 resolution 33
Table 4-2: Convolution parameters for DCGAN generator for image of 224x224x3 resolution 34
Table 4-3: Convolution parameters for DCGAN discriminator for 64x64x3 image 35
Table 4-4: Convolution parameters for DCGAN discriminator for 224x224x3 image 35
Table 4-5: Convolution parameters for PGGAN generator 37
Table 4-6: Convolution parameters for PGGAN discriminator 37
Table 4-7: Model parameters for learning 40
Table 5-1: Confusion matrix of ISIC dataset 41
Table 5-2: Classification performance on ISIC dataset 41
Table 5-3: Confusion matrix on the augmented ISIC dataset 42
Table 5-4: Classification performance on augmented dataset 42
Table 5-5: Confusion matrix on the augmented ISIC dataset with 64x64x3 image from DCGAN 48
Table 5-6: Classification performance on augmented dataset with 64x64x3 image from DCGAN 48
Table 5-7: Confusion matrix on the augmented ISIC dataset with 256x256x3 image from PGGAN 49
Table 5-8: Classification performance on augmented dataset with 256x256x3 image from PGGAN 49

List of figures

Figure 1-1: Skin melanoma image (Mayoclinic, 2022) 10
Figure 1-2: Skin lesion images with label for melanoma classification (Codella et al., 2019) 10
Figure 1-3: GAN based model for ISIC dataset 11
Figure 2-1: Machine Learning Model for lesion classification by (Benedetti et al., 2021) 13
Figure 2-2: Soft attention for the lesion image (Datta et al., 2021) 14
Figure 2-3: (a) architecture with inception-resnet block; (b) architecture with soft attention block (Datta et al., 2021) 14
Figure 2-4: Training and test pipeline (Yao et al., n.d.) 15
Figure 2-5: GAN architecture 16
Figure 2-6: GAN learns a probability distribution of images for each class 16
Figure 2-7: DCGAN generator (Radford et al., n.d.) 17
Figure 2-8: PGGAN training process (Karras et al., 2017) 18
Figure 2-9: Upscaling and down-scaling of image (Karras et al., 2017) 18
Figure 2-10: StyleGAN generator (Karras et al., 2018) 19
Figure 2-11: Conditional adversarial network (Mirza and Osindero, n.d.) 20
Figure 2-12: Image generated by LAPGAN from right to left (Denton et al., n.d.) 20
Figure 2-13: LAPGAN training (Denton et al., n.d.) 21
Figure 2-14:CT images generated with PGGAN (Bowles et al., 2018, n.d.) 21
Figure 2-15 Brain Tumor image generated with PGGAN (Han et al., 2019a). Success and failure cases 22
Figure 2-16: Training flow for Brats Classification (Han et al., 2019a) 22
Figure 2-17: DCGAN model used to train liver lesion (Frid-Adar et al., 2018) 23
Figure 2-18: Liver lesion image. Left side: actual image. Right side: Synthetic image with GAN (Frid-Adar et al., 2018) 23
Figure 2-19: Skin lesion images generated with DCGAN (Pollastri et al., n.d.) 23
Figure 2-20: Checkerboard effect with generated images with DCGAN (Pollastri et al., n.d.) 24
Figure 2-21: Auto encoder training (Mariani et al., 2018) 24
Figure 2-22: GAN Initialization (Mariani et al., 2018) 25
Figure 2-23: GAN training (Mariani et al., 2018) 25
Figure 3-1: Automatic classification of lesion images 27
Figure 3-2: Training ML model with data augmentation from GAN 27
Figure 3-3: Class imbalance in ISIC Dataset 28
Figure 3-4: Research flow 29
Figure 3-5: Training GAN model on ISIC dataset 30
Figure 3-6: Data augmentation with GAN 30
Figure 4-1: Machine learning pipeline 32
Figure 4-2: DCGAN generator as proposed by (Radford et al., n.d.) to generate image of 64x64x3 resolution.. 33
Figure 4-3: DCGAN generator to generate image of 224x224x3 resolution 34
Figure 4-4: DCGAN discriminator as proposed by (Radford et al., n.d.) to discriminate image of 64x64x3 resolution 34
Figure 4-5: DCGAN discriminator for 224x224x3 image 35
Figure 4-6: PGGAN for ISIC dataset as proposed by (Karras et al., 2017) 36
Figure 4-7: Image classification for VGG16 model as proposed by (Simonyan and Zisserman, 2015) 38
Figure 4-8: Image classification for VGG16 with 64x64x3 image resolution 38
Figure 4-9: Custom data loader flowchart 39
Figure 5-1: Generated 224x224x3 resolution malignant melanoma images from DCGAN 43
Figure 5-2: Generated 64x64x3 resolution malignant melanoma images from DCGAN 44
Figure 5-3: Generator and discriminator loss during training DCGAN for 224x224x3 resolution image 44

Figure 5-4: Generator and discriminator loss during training DCGAN for 64x64x3 resolution image 45
Figure 5-5: Generated skin lesion images from PGGAN at 4x4x3 resolution..... 45
Figure 5-6: Generated skin lesion images from PGGAN 8x8x3 resolution 46
Figure 5-7: Generated skin lesion images from PGGAN 16x16x3 resolution..... 46
Figure 5-8: Generated skin lesion images from PGGAN 32x32x3 resolution..... 46
Figure 5-9: Generated skin lesion images from PGGAN 64x64x3 resolution..... 47
Figure 5-10: Generated skin lesion images from PGGAN 128x128x3 resolution 47
Figure 5-11: Generated skin lesion images from PGGAN 256x256x3 resolution 47
Figure 5-12: Generator and discriminator loss during training PGGAN for 256x256x3 resolution image..... 48
Figure 5-13: Classification performance summary 49

List of abbreviations

ML	Machine Learning
GAN	Generative Adversarial Network
ISIC	International Skin Imaging Collaboration
RMSProp	Root Mean Squared Propagation
BAGAN	Balanced GAN
Conv2D	2D convolution layer
DCGAN	Deep Convolutional GAN
LAPGAN	Laplacian Pyramid GAN
PGGAN	Progressive GAN
WGAN	Wasserstein GAN
LR	Learning Rates
BCELoss	Binary Cross Entropy Loss
BCEWithLogitsLoss	Binary Cross Entropy Loss with Sigmoid

Chapter1: Introduction

1.1 Background of study

Malignant melanoma is a dangerous form of skin cancer, mostly common among Caucasian (Apalla et al., 2017). The melanoma cases are increasing annually at a rate of 0.6% over 50 years. The number of cases of skin melanoma represents 4.5% of all cancer cases in 2016. Melanoma can spread deeper into the skin or other parts of the body. So, if melanoma is not treated early, it can become fatal (Stanley et al., 2007).

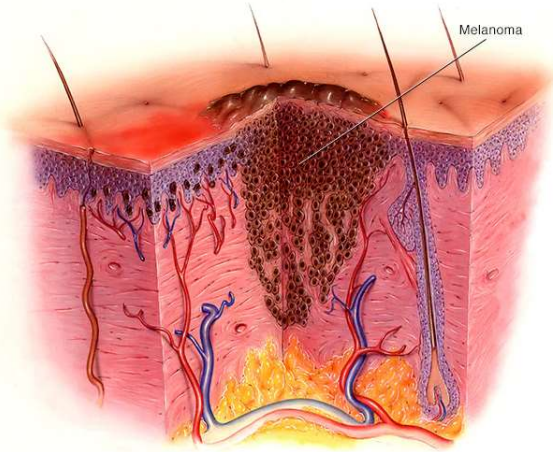

Figure 1-1: Skin melanoma image (Mayoclinic, 2022)

It is challenging to recognize benign and malignant skin cancer since they look very similar. A dermatologist's visual assessment is the initial phase in identifying and diagnosing a skin cancer, it is normally followed by dermoscopy imaging for additional investigation. At this point, machine learning classification algorithm is used. The input to the classification algorithm is an image of the skin cancer and the model needs to classify whether the image is melanoma or benign.

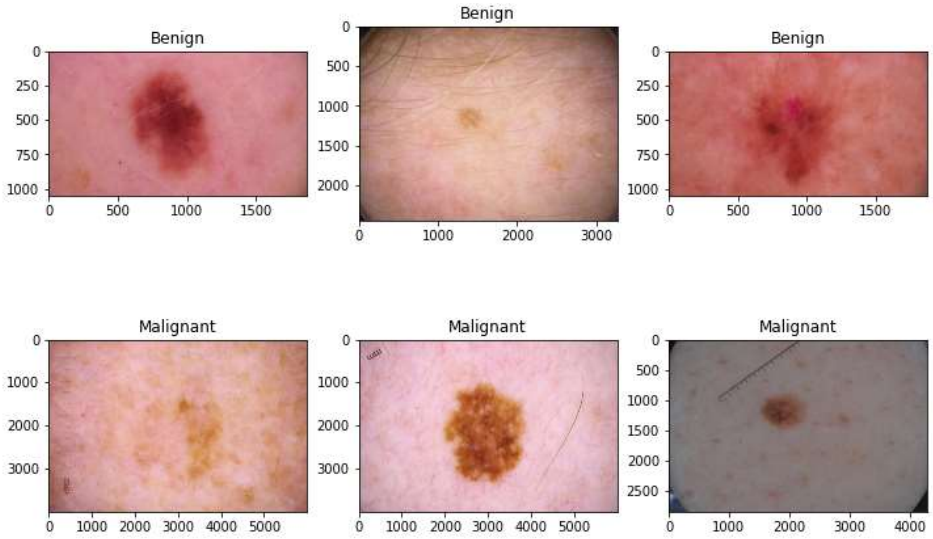

Figure 1-2: Skin lesion images with label for melanoma classification (Codella et al., 2019)

In order to help the researchers to come up with machine learning models, International Skin Imaging Collaboration (ISIC) conducts challenges for three types of tasks (Codella et al., 2019). Task 1 is lesion segmentation, task 2 is lesion attribute detection and task 3 is lesion disease classification. However, one of the challenges of using modern machine learning techniques such as deep neural networks for classification task is availability of large and balanced data. In ISIC dataset there is high imbalance among images of benign and malignant types. Such imbalance dataset is not suitable for classification with deep neural network. The model becomes biased and impacts classification scores such as precision and recall.

In our thesis work, we will like to improve the balance of ISIC dataset with augmentation methods. There are existing data augmentation approach like random crop, flips etc. which are often applied to improve the number of training data, but they are simply duplicating the existing training features. It does not add enough variety to the training data to learn the generic features of skin cancer images. Generative adversarial network or GAN (Goodfellow et al., 2014; Salimans et al., 2016) offers an alternative method to learn features from a dataset to generate synthetic images with appearance similar to the real images. In this research we will explore GAN based data augmentation method on ISIC dataset and evaluate if it improves the lesion classification performance.

1.2 Problem statement

GAN based data augmentation methods to improve classification and segmentation tasks are very common. However, it is not used widely in lesion classification. There are several state-of-art GAN models available. In this research we will explore the popular GAN models for data augmentation with ISIC dataset and evaluate their performance.

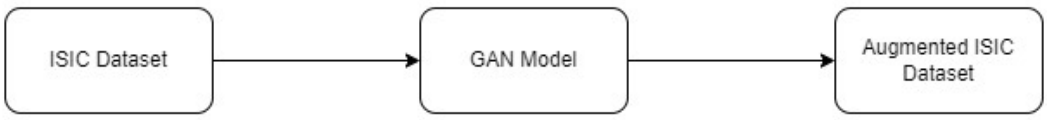

Figure 1-3: GAN based model for ISIC dataset

1.3 Aim and objectives

The aim of this research is to propose a GAN architecture to augment images in ISIC dataset. By augmenting the dataset, we will be able to improve the balance of the images between benign and malignant types. This will help to learn a generic deep neural network model for skin cancer image classification.

The research objectives are as follows:

- To explore state-of-art GAN architectures
- To propose a GAN architecture suitable for augmenting ISIC dataset
- To evaluate the performance of state-of-art classification model on ISIC dataset after GAN based data augmentation

1.4 Scope of study

The scope of the study will be to explore GAN based data augmentation techniques on ISIC dataset. We will limit our study to only ISIC dataset.

1.5 Significance of study

The study is important to augment the images in imbalanced medical image dataset such as ISIC. This will help to learn a generic deep neural network model to improve classification performance of skin cancer images and could be lifesaving.

1.6 Structure of study

The reminder of the thesis is organized as follows.

Chapter 2 reviews related work in the area of lesion image classification. It discusses the methods used to improve classification performance in lesion dermoscopy. It introduces generative adversarial network (GAN). It discusses the application of GAN in medical images.

Chapter 3 explains the research methodology. It explains the dataset and challenges in lesion image classification. It then explains the research flow and evaluation metrics of the research.

Chapter 4 explains the implementation details. It explains the machine learning pipeline. It shows how different models in this research are implemented. It discusses the parameters, optimization, loss functions used in each model.

Chapter 5 shows the result. It discusses the result of applying GAN on ISIC dataset. It then shows how the classification performance changes when we train the classification model on the augmented dataset.

Chapter 6 provides a summary of the observations made during the research. It highlights the recommendation and future work.

Chapter2: Literature Review

2.1 Introduction

This chapter discusses the state-of-art lesion classification methods. We show how data augmentation and ensemble-based models have been used to improve classification performance. Then we explain generative adversarial network (GAN) and current state-of-art GAN models. Next, we discuss how different GAN models have been used to perform data augmentation task in medical images. Finally, we summarize the GAN based approaches and challenges for using GAN in lesion classification task.

2.2 Classification in Lesion Analysis

It is challenging to train imbalanced dataset such as skin lesion dataset. (Benedetti et al., 2021) used transfer learning approach towards lesion classification in HAM10000 dataset (Tschandl et al., 2018). HAM10000 dataset is an imbalanced dataset among 7 classes.

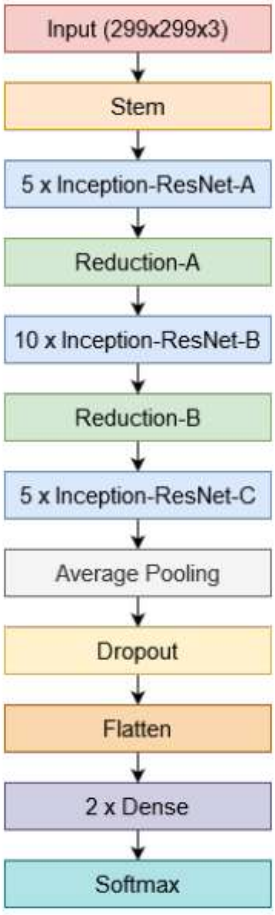

Figure 2-1: Machine Learning Model for lesion classification by (Benedetti et al., 2021)

The input image is of the resolution 299x299x3. The softmax layer gives a probability distribution among 7 classes of HAM10000 dataset. The inception-resetNet blocks are combination of inception and residual network model proposed by (Szegedy et al., 2016). (Benedetti et al., 2021) used these pretrained blocks to train the lesion classification model and improved accuracy from 73.4% to 78.9%.

The skin cancer images have irrelevant features like hair, veins. The parts of the image which are important for identifying skin cancer are limited. (Datta et al., 2021) introduced soft attention to identify

the important parts of the image to improve classification. The soft attention block attaches weight to each pixel in term of its importance.

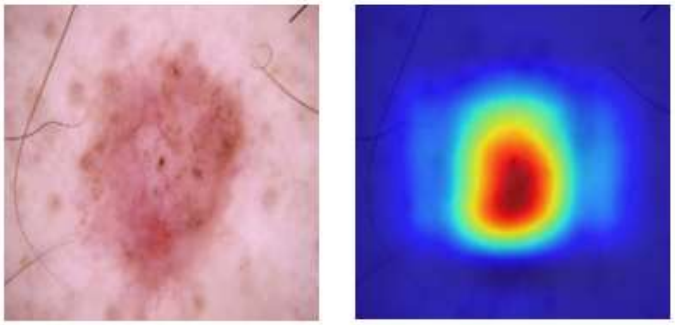

Figure 2-2: Soft attention for the lesion image (Datta et al., 2021)

Figure 2-3 shows the improvement of inception-resnet based architecture with soft attention block. Soft attention block multiplies with the feature tensor to give relative importance to the pixels which are more important to classify lesion images.

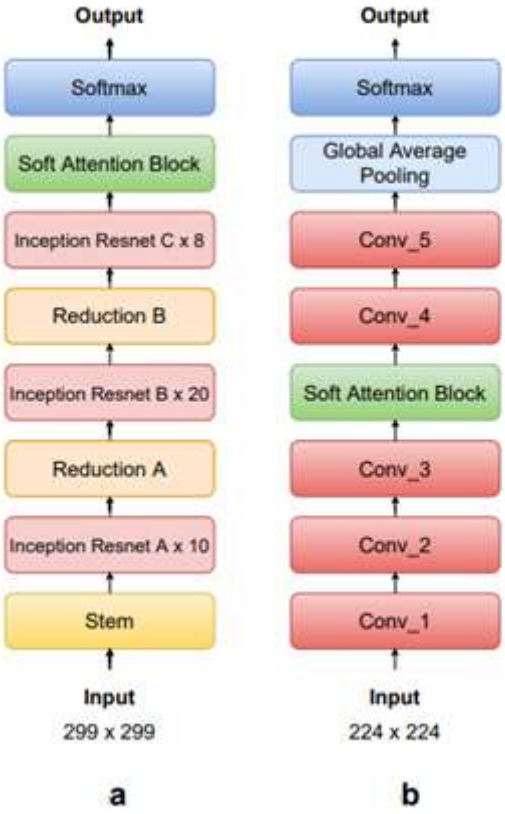

Figure 2-3: (a) architecture with inception-resnet block; (b) architecture with soft attention block (Datta et al., 2021)

Another improvement of lesion classification was proposed by (Gessert et al., 2019) with ensemble network. EfficientNet based 8 different neural networks are used in the ensemble. All the model are structurally similar. They differ in terms of number of feature maps per layer. (Gessert et al., 2019) also proposed image pre-processing where unimportant parts of the lesion image are automatically cropped before the training. The automatic cropping is done with a bounding box, which is identified by finding the box where the average intensity difference between the box and outside are sufficiently large. With ensemble the sensitivity improved from 68.8 to 72.5.

(Yao et al., n.d.) introduced multi-weighted loss (MWL) for training loss in class imbalance ISIC dataset. The usual cross entropy loss is weighted by a factor based on the size of the dataset of that class.

$$\mathrm{MWL}_{\mathrm{focal}}(z,y) = -C_y\left(\frac{1}{N_y}\right)^{\alpha}\sum_{i=1}^{C}(1-p_i^t)^{\gamma}\log(p_i^t). \qquad \text{Eq.2.1}$$

Figure 2-4 shows the training and test pipeline of the single deep learning model for lesion classification by (Yao et al., n.d.). It included data augmentation with random augmentation as proposed by (Cubuk et al., n.d.). Random data augmentation means randomly choose a transformation like translation, rotation, shear and apply on a minibatch. After random augmentation, (Yao et al., n.d.) performed random crop. The performance of this single deep learning model is comparable to ensemble method.

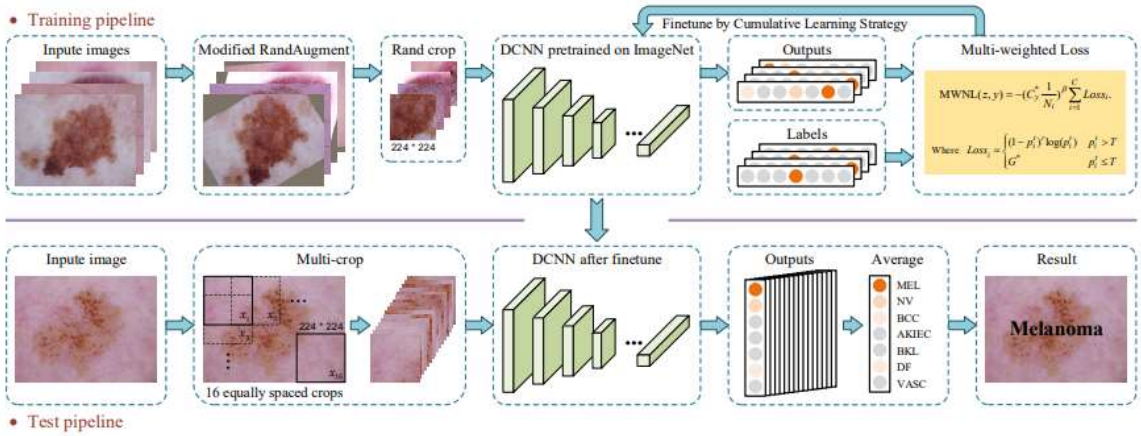

Figure 2-4: Training and test pipeline (Yao et al., n.d.)

In summary, the current state-of-art classification of lesion images involves transfer learning techniques, soft attention block to focus on the important part of the image and data augmentation using basic affine transformation. The data augmentation based on affine transformation tries to overcome the limited availability of training images.

2.3 Generative Adversarial Network (GAN)

GAN provides a way to learn deep representations of the training data. GAN consists of two deep neural network models, known as generator and discriminator, which compete with each other during learning. Generator generates fake images based on a noise vector as input. Discriminator tries to identify whether a given image is real or fake. Generator will try to create an image such that discriminator fails to identify it as fake image. The overall operation is shown in Figure 2-5

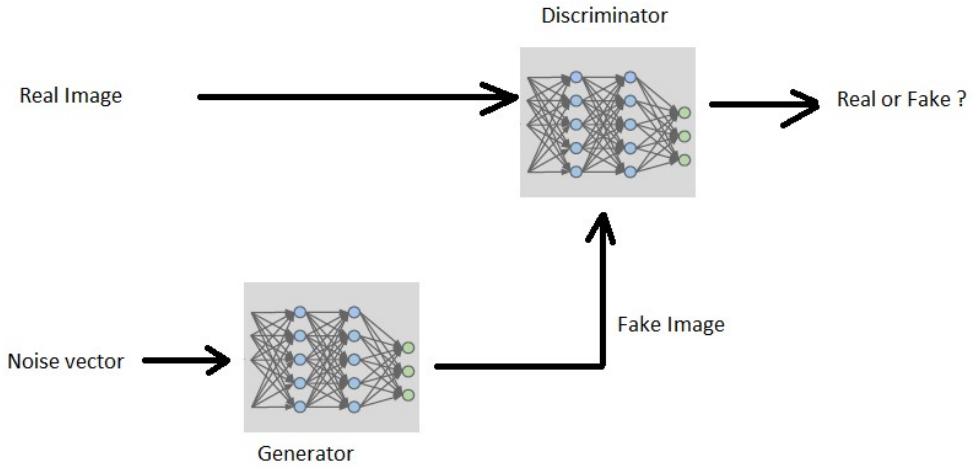

Figure 2-5: GAN architecture

The objective function of GAN can be expressed with the eq.2.2. G is the generator function which takes a random vector z as parameter and generates an image. D is the discriminator which can identify if the generated image is real (1) or fake (0). We train both G and D such that from D's perspective, log(1-D(G(z)) and log(D(x)) is maximized, i.e., D can identify the real and fake images and at the same time G is trained with perspective to minimize log(1-D(G(z)), i.e., D is unable to identify whether the generated image is fake

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))]. \quad \text{Eq.2.2}$$

Generator eventually learns a probability distribution P(X) of the training images in the dataset, where X is the image (Figure 2-6). So given any random vector, generator can sample an image.

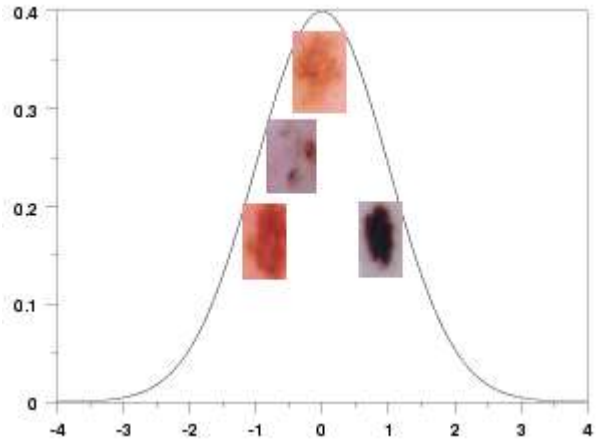

Figure 2-6: GAN learns a probability distribution of images for each class

2.3.1 Exploring GAN architectures

In this section we will explore different GAN architectures.

2.3.1.1 Deep Convolutional GAN

Deep convolutional GAN or DCGAN (Radford et al., n.d.) was the initial GAN architecture which was able to produce high resolution images. It takes a 100x1 noise vector (z) and converts it to a 64x64x3 image (G(z)) as shown in Figure 2-7. The discriminator model mirrors the generator model. It takes a 64x64x3 image and outputs whether the given image is real or fake.

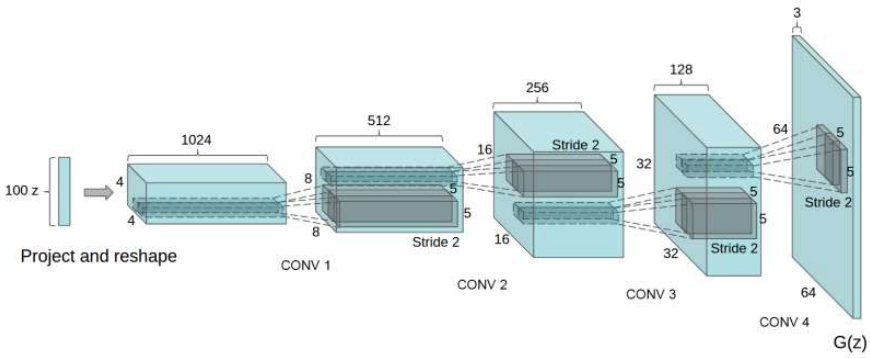

Figure 2-7: DCGAN generator (Radford et al., n.d.)

The key elements of DCGAN architectures are as follows,

a) On generator side it includes fractional strided convolutions, on discriminator side it includes strided convolutions instead of pooling layers.
b) It uses batchnorm in both generator and discriminator.
c) Removes fully connected hidden layers
d) On generator side, it uses Relu activation on all layers except output where it uses tanh. On discriminator it uses leaky Relu in all layers

2.3.1.2 Wasserstein GAN

(Arjovsky et al., 2017) introduced Wasserstein GAN (WGAN) as an enhancement to DCGAN. The main difference between DCGAN and WGAN is the introduction of Wasserstein loss function. Instead of just classifying a generated image as real or fake, Wasserstein loss calculates critic score which corresponds to the realness or fakeness of image. WGAN also uses gradient clipping. These measures ensure that WGAN is more stable than DCGAN. The discriminator is called critic in WGAN model.

The other major difference of WGAN and DCGAN are,

a) Use linear activation function in the output layer instead of sigmoid
b) Use -1 for labelling real image and 1 for fake image, instead of 1 and 0
c) Update critic model a greater number of times than generator in each iteration
d) Use RMSProp gradient descent with small learning rate and no momentum

2.3.1.3 Progressive GAN

(Karras et al., 2017) introduced progressive GAN (PGGAN) to progressively generate images from 4x4x3 resolution to 1024x1024x3 resolution. The rationale of doing progressively is that it is easier to learn how to generate small resolution image, and then gradually learn additional feature for higher resolution images. The initial layers of generator learn large scale features of the image and the later layers of the generator learn finer detail.

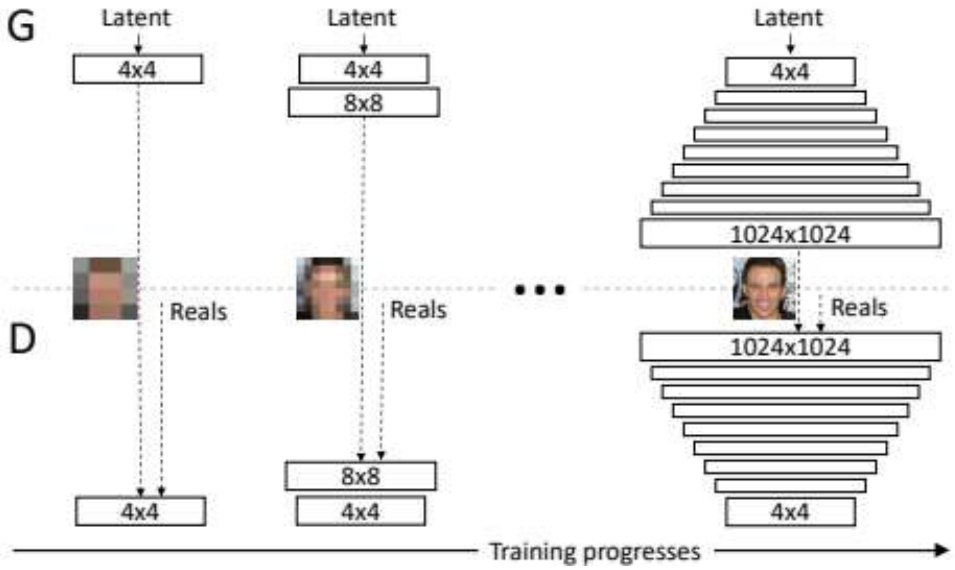

Figure 2-8: PGGAN training process (Karras et al., 2017)

The way the image is scaled up is shown in Figure 2-9. The Scale up by 2x is done with transpose convolution. The toRGB block is implemented with weighted scaled convolution. The alpha factor decides how the scaled image is faded in with output from toRGB block. The discriminator exactly mirrors the generator steps.

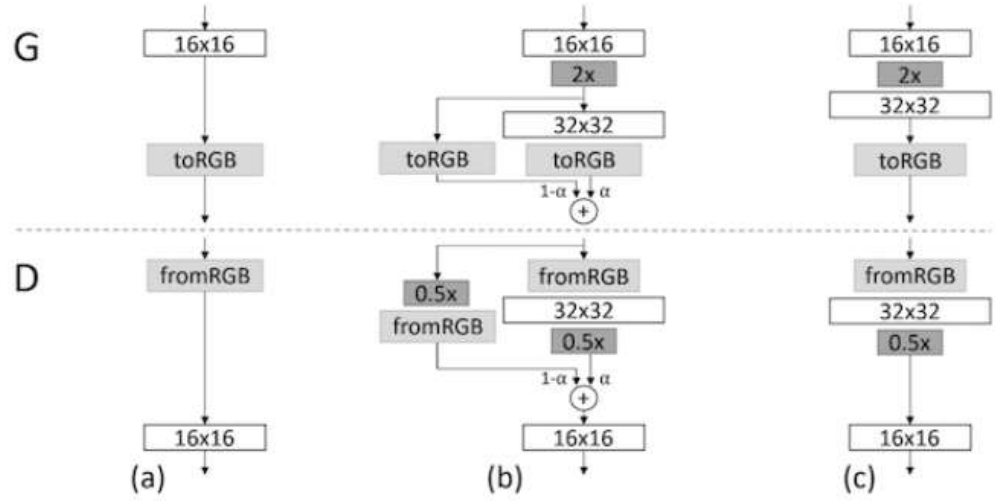

Figure 2-9: Upscaling and down-scaling of image (Karras et al., 2017)

2.3.1.4 StyleGAN

The base of StyleGAN (Karras et al., 2018) is PGGAN, i.e., images are progressively generated to higher resolution. StyleGAN (Figure 2-10) introduces a standalone mapping network which generates a style vector from a latent vector. This style vector is used to control style of the generated image at each stage.

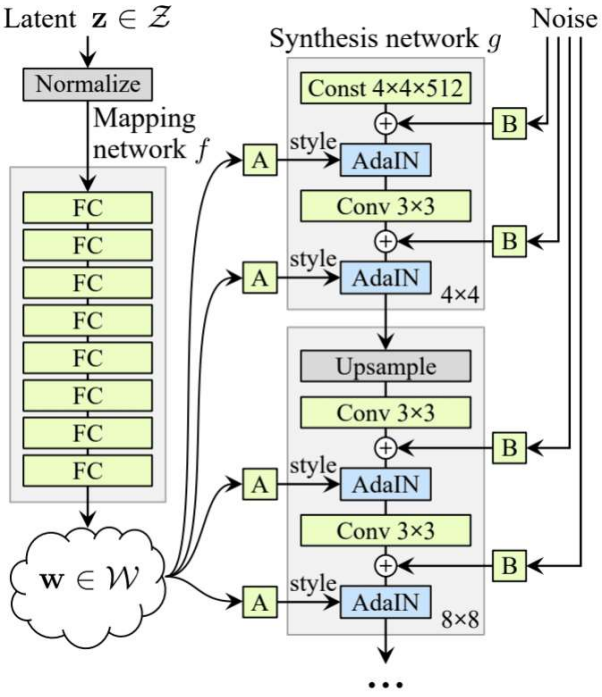

Figure 2-10: StyleGAN generator (Karras et al., 2018)

There is also a noise vector added at each stage of the generated image to control some variation of the generated image. The AdaIN layer standardizes the output of the feature map to a standard gaussian and then adds the style vector to it. The StyleGAN model is capable of generating high resolution images of faces but also control the style at different level of details.

2.3.1.5 Conditional GAN

(Mirza and Osindero, n.d.) introduced conditional GAN. Both the generator and discriminator are trained with condition of class labels. Here the objective function of GAN is modified as shown in Eq. 2.3.

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x}|\boldsymbol{y})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z}|\boldsymbol{y})))] \quad \text{Eq 2.3}$$

The generator function G, learns a conditional probability distribution given a class name. G(z|y) generates an image from an input vector z and class label y. The Discriminator D tries to identify whether the image from the given class label is real or fake.

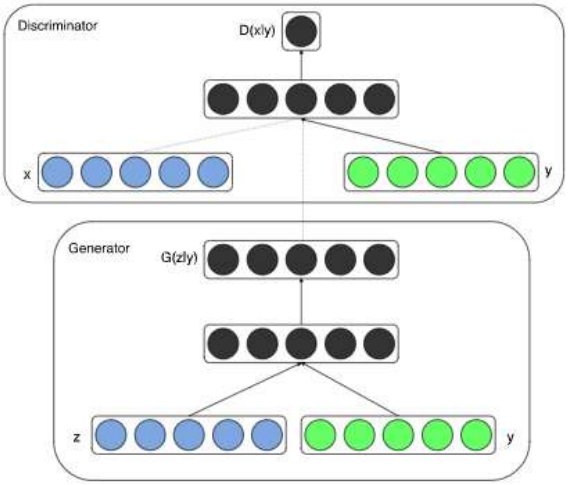

Figure 2-11: Conditional adversarial network (Mirza and Osindero, n.d.)

2.3.1.6 Laplacian Pyramid of Adversarial Networks

(Denton et al., n.d.) introduced Laplacian pyramid adversarial network or LAPGAN on top of conditional GAN. In LAPGAN, Laplacian pyramid framework is used to progressively generate the images. It has set of generative convnet models (shown as G0, G1, G2, G3) generating images at different resolutions. At each level, the image from previous level is up-sampled. The up-sampled image is added with a noise vector z and this is used to generate image. This generated image is added with the up-sampled image again. This image generation flow is explained with the equation 2.4

$$\tilde{I}_k = u(\tilde{I}_{k+1}) + \tilde{h}_k = u(\tilde{I}_{k+1}) + G_k(z_k, u(\tilde{I}_{k+1})) \qquad \text{Eq. 2.4}$$

This process continues as the resolution of the image keeps increasing.

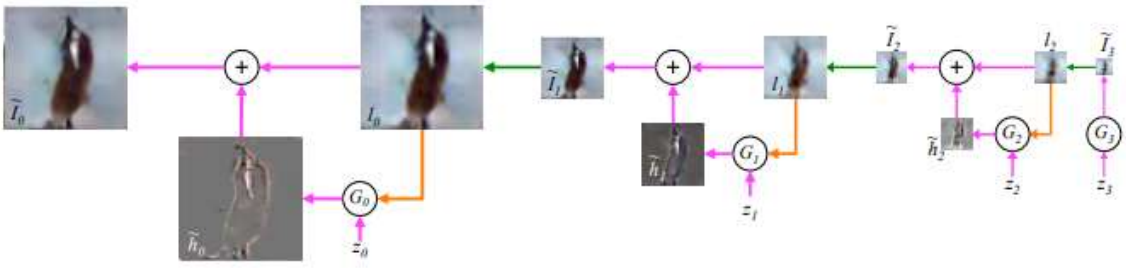

Figure 2-12: Image generated by LAPGAN from right to left (Denton et al., n.d.)

Each of the convnet models are trained separately at that resolution as a conditional GAN.

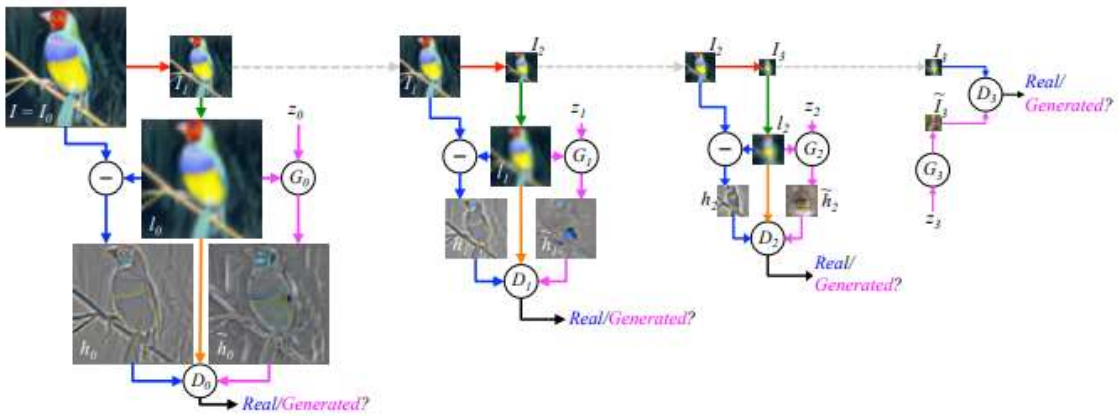

Figure 2-13: LAPGAN training (Denton et al., n.d.)

2.3.2 GAN in Medical Images

Different variations of GAN have been adopted for data augmentation task in various medical image datasets. (Bowles et al., n.d.) used PGGAN (Karras et al., n.d.) based augmentation on CT image dataset. (Bowles et al., n.d.) used 80,000 CT image samples from train dataset and trained the PGGAN. The only modification to original PGGAN model done was to concatenate a 32x32 gaussian noise at the start of fourth resolution level i.e., when the model scales up from 16x16 image to 32x32 image (Figure 2-9). This noise concatenation improves CT image with more realistic pattern. The generator model creates 128x128 CT image. Once PGGAN is trained, synthetic images are sampled randomly using the trained generator model and added to the CT images for training.

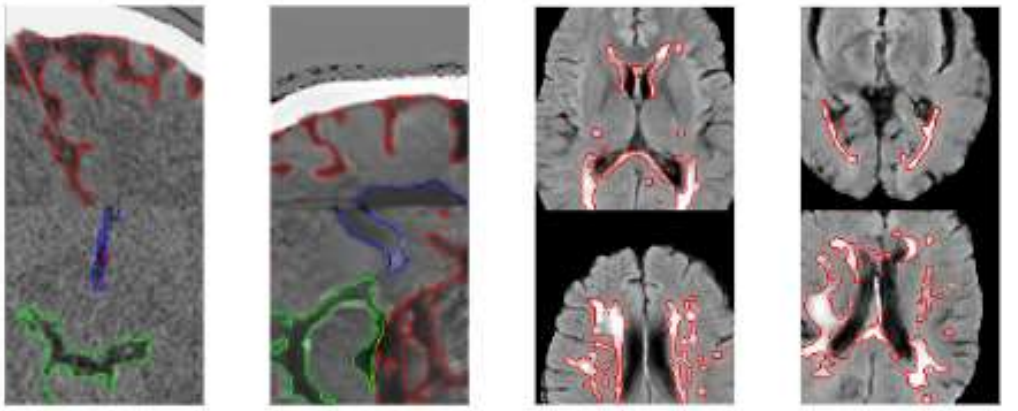

Figure 2-14:CT images generated with PGGAN (Bowles et al., 2018, n.d.)

(Han et al., 2019a) used PGGAN for data augmentation task in brain tumor dataset, BraTS (Sicas Medical Image Repository, 2016). They used Wasserstein loss function with gradient penalty. The PGGAN model was used to generate brain tumor with resolution of 256x256x3. Figure 2-15 shows the PGGAN generated brain tumor images. The successfully generated images are used for the classification training.

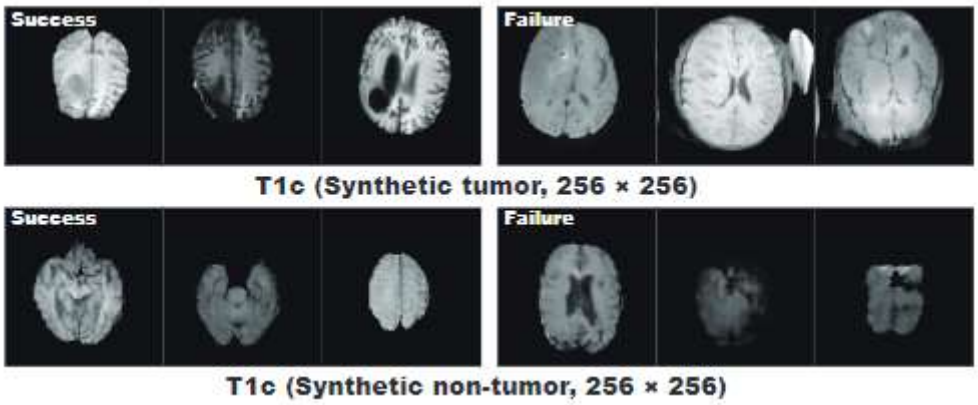

Figure 2-15 Brain Tumor image generated with PGGAN (Han et al., 2019a). Success and failure cases

Then these generated images are used to augment the training dataset of BraTS. (Han et al., 2019a) also used geometric transformation to generate more images. All these images are then used to train a ResNet model (Jian, 1996) for classification.

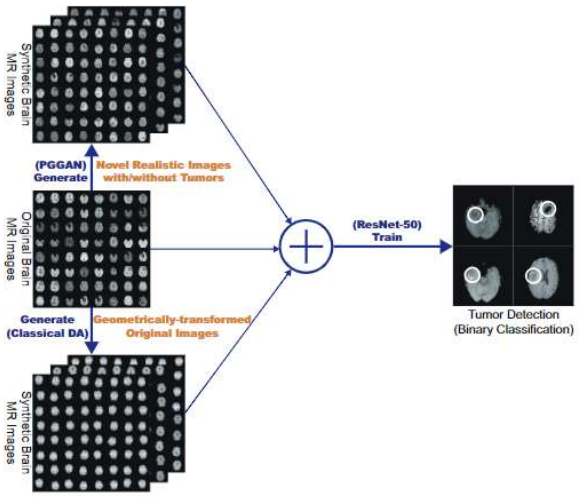

Figure 2-16: Training flow for Brats Classification (Han et al., 2019a)

(Frid-Adar et al., 2018) used DCGAN (Radford et al., n.d.) was for liver lesion classification. As shown in Figure 2-17, a random noise vector of 100x1 dimension is trained to generate lesion image of resolution 64x64x1.

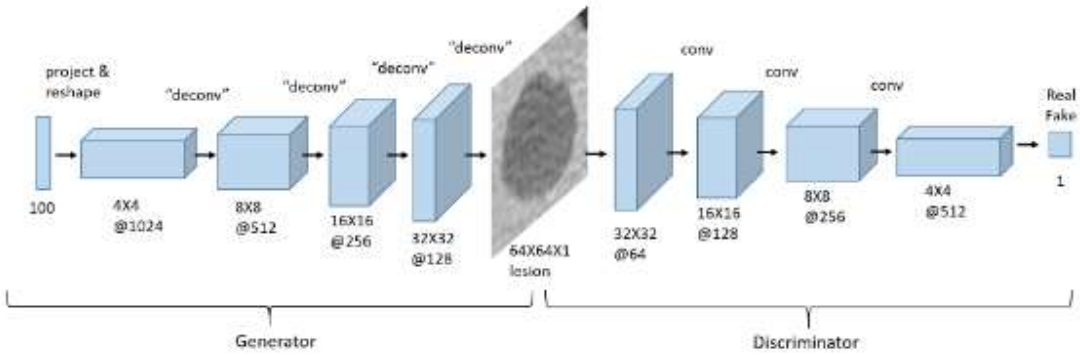

Figure 2-17: DCGAN model used to train liver lesion (Frid-Adar et al., 2018)

With the DCGAN model, (Frid-Adar et al., 2018) could generate very similar looking liver lesion images.

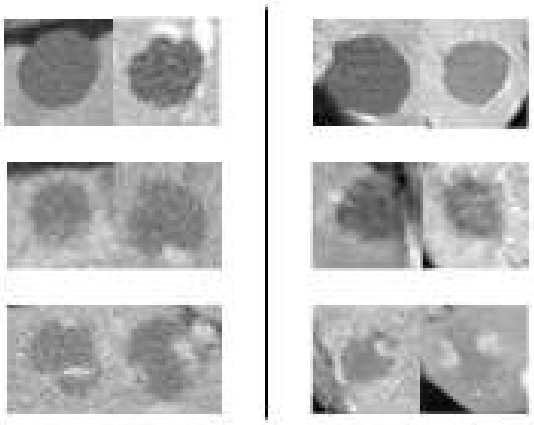

Figure 2-18: Liver lesion image. Left side: actual image. Right side: Synthetic image with GAN (Frid-Adar et al., 2018)

(Pollastri et al., n.d.) used DCGAN to generate skin lesion images for segmentation task. The model is used to take a random vector of 100x1 dimension and generate skin lesion images with resolution 192x256x3.

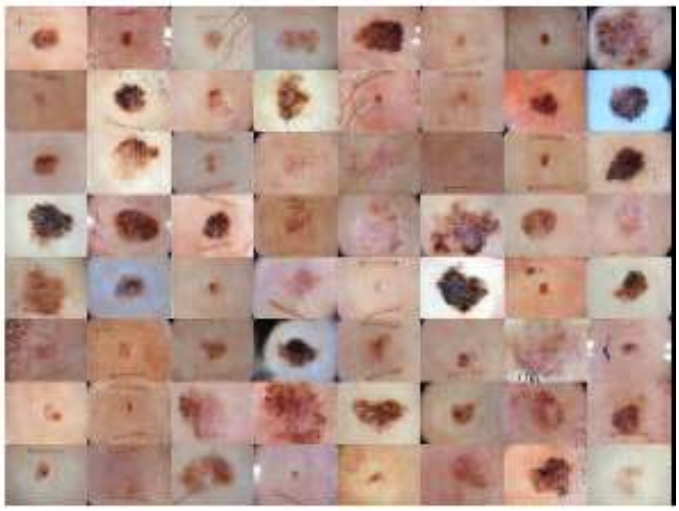

Figure 2-19: Skin lesion images generated with DCGAN (Pollastri et al., n.d.)

(Pollastri et al., n.d.) observed that the images generated with DCGAN, though had a checkerboard effect, but had good diversity and it was easy to train the segmentation network with the generated images.

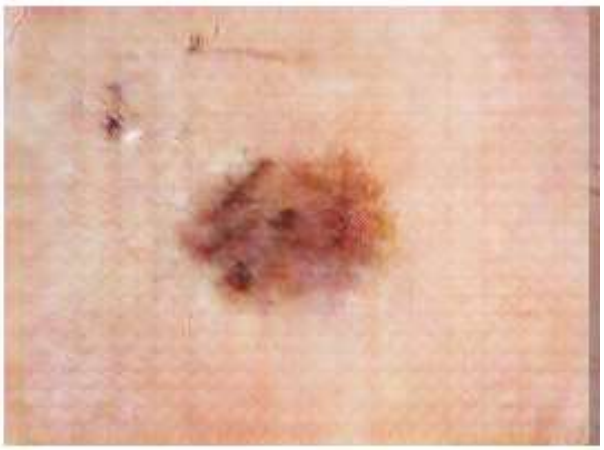

Figure 2-20: Checkerboard effect with generated images with DCGAN (Pollastri et al., n.d.)

2.3.2.1Mode Collapse

When training GAN with multiple classes and there is large imbalance across classes, then GAN suffers from mode collapse problem. Mode collapse is a problem where generator learns only to generate images from majority class. There are several research done in GAN to avoid mode collapse. One method is to train different generator models for each class. So, when training the model, it only sees the images from a specific class. That way, the learned generator can sample images of one class independent of other classes.

(Mariani et al., 2018) introduced BAGAN model to train GAN with class imbalance. The training is done in three stages. In the first stage, an auto encoder is trained on each class. So, auto encoder learns the distribution of latent vector for each class.

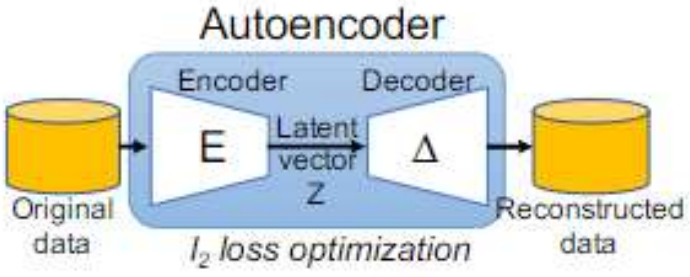

Figure 2-21: Auto encoder training (Mariani et al., 2018)

In the next stage, GAN module is initialized. The GAN network reuses the auto encoder model. The generator model reuses the decoder model and discriminator reuses the encoder model. The GAN model weights are initialized with the learned auto encoder model weights.

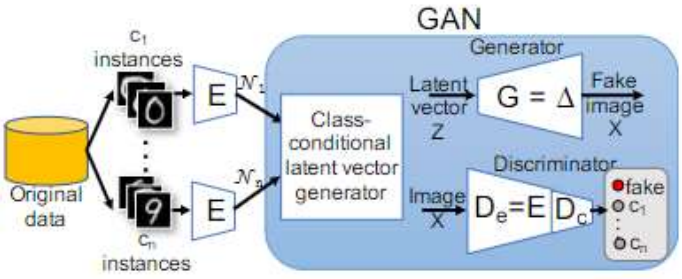

Figure 2-22: GAN Initialization (Mariani et al., 2018)

The final stage is GAN training with generator trying to create fake images and discriminator trying to find if the generated image is real or fake. Because of the use of autoencoders for initializing the GAN weights, GAN starts training from close to a good solution away from mode collapse.

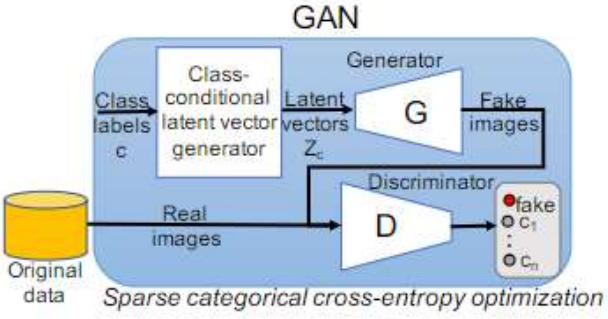

Figure 2-23: GAN training (Mariani et al., 2018)

2.3.3 Summary and Challenges of GAN based approaches in medical images

GAN models have been used in different medical image task

Table 2-1: Summary of popular GAN models in image classification

GAN Model	Author	Medical Image Task
PGGAN	(Bowles et al., n.d.)	CT image dataset classification task
PGGAN	(Han et al., 2019a)	BraTS classification task
DCGAN	(Frid-Adar et al., 2018)	Liver lesion classification task
DCGAN	(Pollastri et al., n.d.)	ISIC segmentation task

PGGAN and DCGAN are the popular choice of GAN model for medical images. However, all these models suffer from mode collapse problem in case the dataset is imbalance. For imbalance issue, we can either do autoencoder based GAN weight initialization as proposed in BAGAN model (Mariani et al., 2018). Alternatively, we can take care by training GAN for each class separately. We can do GAN only for the minority class. During our literature survey, we have not observed any major work of GAN towards ISIC classification task.

Chapter3: Research Methodology

3.1 Introduction

An accurate machine learning (ML) classification of skin melanoma is an important step towards early detection and treatment of the disease. Figure 3-1 shows an example of such ML classification model. It outputs a probability of input image being benign or malignant.

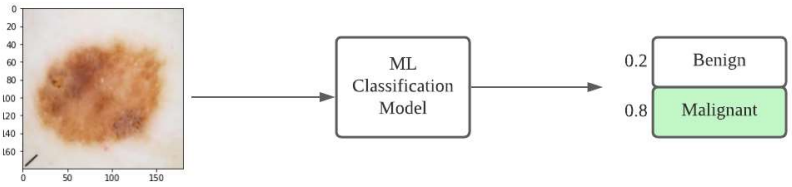

Figure 3-1: Automatic classification of lesion images

For improving classification performance on the imbalance ISIC dataset, we will use GAN to generate synthetic images for the imbalanced class and combine it with the rest of the ISIC image dataset. The overall flow of training is shown below in Figure 3-2. The transform step resizes the image for state-of-art classification model. In our research, we will adopt two different approaches for classification. At first, we will adopt transfer learning approach to train state-of-art VGG16 ML model for ISIC image classification. We will modify the final layer of VGG16 to classify between benign and malignant class. In the second approach, we will modify and train VGG16 ML model for smaller resolution ISIC images generated with GAN.

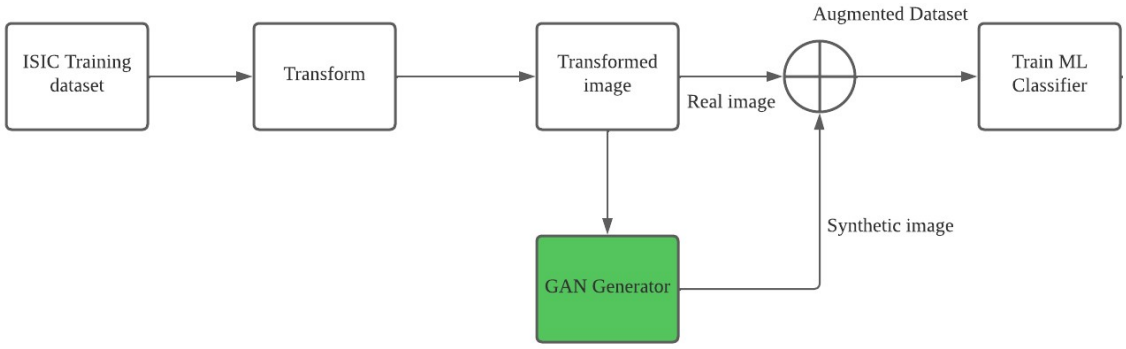

Figure 3-2: Training ML model with data augmentation from GAN

3.2 Dataset

For machine learning research on skin cancer images, international skin imaging collaboration (ISIC) conducts challenges for three types of tasks (Codella et al., 2019).

1. Task1: Lesion segmentation. Given a skin cancer image, identify the segment of the image with melanoma
2. Task2: Lesion attribute detection. Given a skin cancer image, identify the type of melanoma
3. Task3: Lesion classification. Given a skin cancer image, identify if the type is melanoma or benign

3.2.1 Exploratory Data Analysis

The ISIC dataset has a train and test csv file. The train csv files look like Table 3-1. The image_name column is the name of the image files which are available in the ISIC dataset.

Table 3-1: Train.csv of ISIC dataset

	image_name	patient_id	sex	age_approx	anatom_site_general_challenge	diagnosis	benign_malignant	target
0	ISIC_2637011	IP_7279968	male	45.0	head/neck	unknown	benign	0
1	ISIC_0015719	IP_3075186	female	45.0	upper extremity	unknown	benign	0
2	ISIC_0052212	IP_2842074	female	50.0	lower extremity	nevus	benign	0
3	ISIC_0068279	IP_6890425	female	45.0	head/neck	unknown	benign	0
4	ISIC_0074268	IP_8723313	female	55.0	upper extremity	unknown	benign	0

Target is 0 for benign and 1 for malignant melanoma. There are total of 33126 images in train dataset. Among 33126 images, 32542 images are benign and 584 are malignant melanoma, indicating large class imbalance.

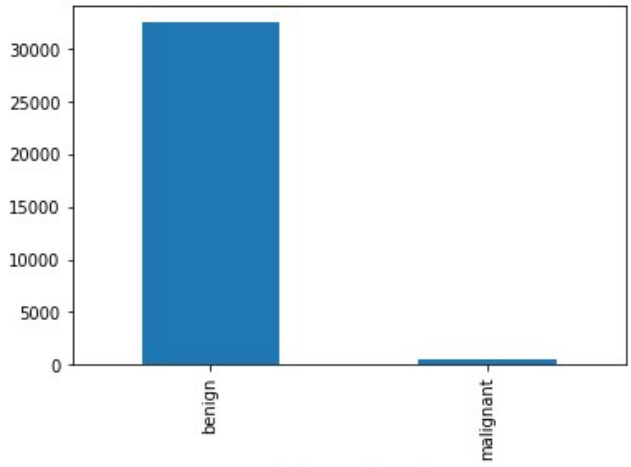

Figure 3-3: Class imbalance in ISIC Dataset

The test dataset has 6592 images. The test csv looks like Table 3-2. The ISIC challenge task 3 is to predict the target class of test dataset

Table 3-2: Test csv of ISIC dataset

	image_name	patient_id	sex	age_approx	anatom_site_general_challenge
0	ISIC_0052060	IP_3579794	male	70.0	NaN
1	ISIC_0052349	IP_7782715	male	40.0	lower extremity
2	ISIC_0058510	IP_7960270	female	55.0	torso
3	ISIC_0073313	IP_6375035	female	50.0	torso
4	ISIC_0073502	IP_0589375	female	45.0	lower extremity

In this research work, we split the train dataset into train and validation dataset in the ratio of 75:25. Once we train on train dataset, we verify on validation dataset.

3.3 Research Flow

A balanced training dataset is a key requirement towards training a machine learning classification model. So, the research work focuses on GAN architecture to create synthetic images for classes where less image is available. The research flow for this work is shown in Figure 3-4.

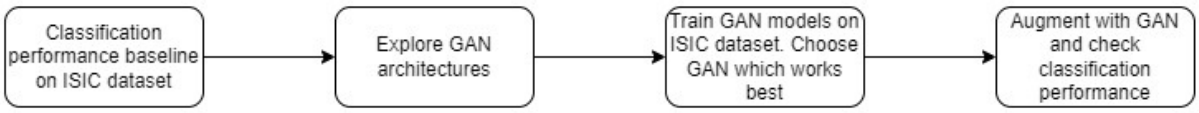

Figure 3-4: Research flow

3.3.1 Classification performance baseline

We will investigate the classification performance on ISIC dataset with a state-of-art classification model. In our research, we choose VGG16 (Simonyan and Zisserman, 2015). We will train VGG16 with transfer learning technique with pretrained weight on ISIC dataset in two steps,

- Step1: We will train VGG16 on ISIC dataset without any augmentation
- Step2: We will use data augmentation technique such as affine transformation to augment the dataset. Then we will train VGG16 on the augmented dataset

With the above two steps, we will baseline the VGG16 classification performance. This is important to baseline the classification performance to understand the gain due to GAN based data augmentation.

3.3.2 Explore GAN architecture

In section 2.3.1 we discussed many GAN architectures. The most popular GAN models in medical images are,

- DCGAN (Radford et al., n.d.): This has been used in augmentation of Liver lesion images
- PGGAN (Karras et al., n.d.): This has been used in augmentation of CT images, BraTS images

We will investigate these two GAN architectures on ISIC dataset.

3.3.3 Train the GAN models

In order to train GAN model, we will implement the start-of-art GAN generator and discriminators. We will train it with malignant melanoma images as shown in Figure 3-5. At the end of training, the GAN model will learn the distribution of the malignant melanoma images.

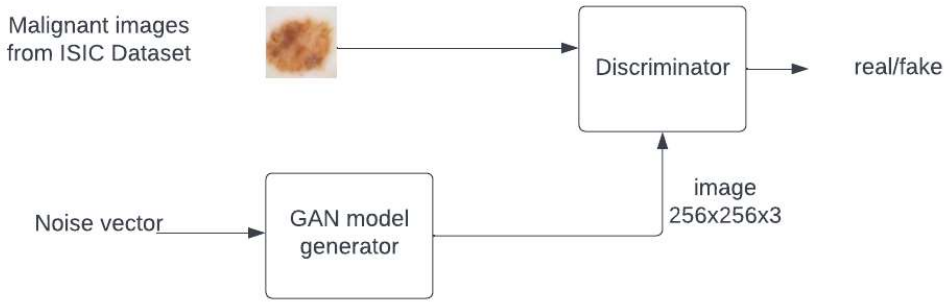

Figure 3-5: Training GAN model on ISIC dataset

3.3.4 Augment the dataset

Once GAN model is trained, the model is used to generate malignant melanoma images. Given a noise vector, GAN model generates an image and the image is added to the image dataset. We repeat the process with next noise vector till we have generated sufficient number of malignant melanoma images to match the number of benign images in the ISIC dataset.

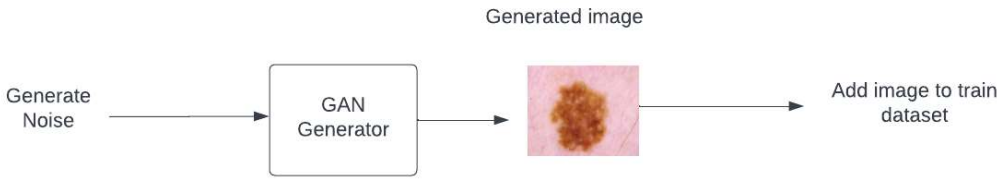

Figure 3-6: Data augmentation with GAN

3.3.5 Training the classification model on augmented dataset

We will have two separate classification model for different resolution of input image. The state-of-art VGG16 is requires input image of the resolution 224x224x3. For GAN models, where we can generate larger resolution image, we will use transfer learning-based approach to train VGG16 model. However, for GAN models which generate images at lower resolution than 224x224x3, we will modify the VGG16 model to fit in the lower input size, while keeping the number of layers of VGG16 same.

3.4 Evaluation Metrics

Classification task performance is evaluated with a confusion matrix (Table 3-3) which depends on how many predictions we made correctly and incorrectly,

Table 3-3: Confusion Matrix

	Predicted Truth	Predicted False
Actual Truth	True Positive (TP)	False Negative (FN)
Actual False	False Positive (FP)	True Negative (TN)

Based on confusion matrix, following evaluation criteria has been defined,

Table 3-4: Classification evaluation

Sensitivity or Recall	TP / (TP + FN)
Specificity	TN / (FP + TN)
Precision	TP / (TP + FP)
F1	2 x (Precision * Recall) / (Precision + Recall)
Accuracy	(TP + TN) / (TP + FN + FP + TN)

We will evaluate these parameters, i.e., sensitivity, specificity, precision, F1 and accuracy in our classification model. We will prioritize recall score to reduce false negative case.

Chapter4: Implementation

4.1 Introduction

In this chapter we will explain the implementation of GAN models for malignant lesion images. We will explain the machine learning pipeline that is used to augment and classify ISIC dataset of lesion images.

4.2 Machine Learning Pipeline

The machine learning pipeline as shown in Figure 4-1. It is split into three phases.

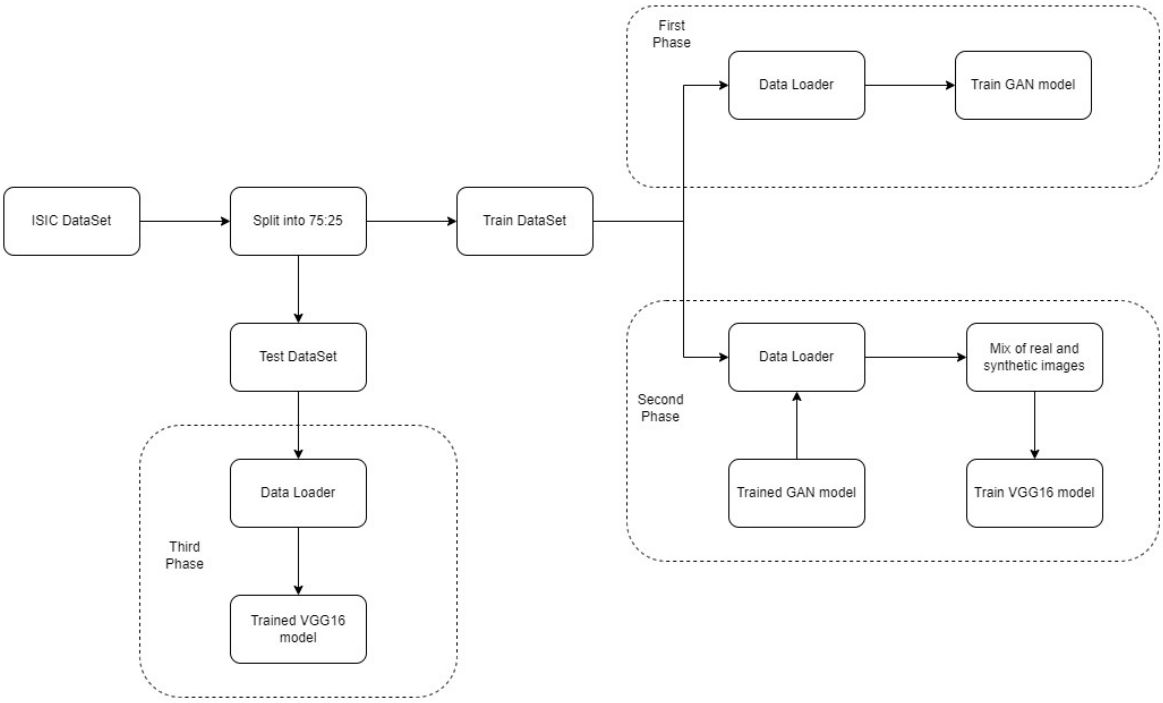

Figure 4-1: Machine learning pipeline

We split the ISIC dataset in the ratio 75:25 for train and test dataset. In the first phase, the train dataset passes through a data loader loads only the malignant images. The data loader applies random rotation and rescales the image. The rescaled malignant images are used to train the GAN model. The DCGAN model is trained twice. Once with image of 224x224x3 resolution and second time with resolution of 64x64x3. We use only malignant images to train GAN model to avoid mode collapse issue when training with imbalanced dataset with dominant images from benign class. Once GAN model is trained, first phase is completed.

In second phase, we train the classification model. In this phase, we train two classification models. One with input image size of 224x224x3 and second with input image size of 64x64x3. The data loader selects images of both malignant and benign images from the train dataset, rescales it properly. For generating malignant images, data loader will first use all the malignant images from ISIC dataset and it will use the GAN model to generate synthetic images. These images are used to train VGG16 model. Once both the VGG16 models are trained, the second phase is completed.

In the third and final phase, the validation dataset is used to the test the VGG16 model. For the two VGG16 models, data loader will rescale the validation image and give input to the model. Based on the prediction of the model, we find the classification parameters like sensitivity, specificity, precision, accuracy and F1 score.

4.3 DCGAN with ISIC dataset

DCGAN as proposed by (Radford et al., n.d.) generates images of resolution 64x64x3. We train DCGAN twice. Once we train DCGAN in its original form to generate 64x64x3 resolution image. Next, we modify DCGAN for generating synthetic image of resolution 224x224x3. Figure 4-2 shows the generator model generating 64x64x3 resolution image. The generator model has series of transpose convolution and scaling up the generated image at every step. Table 4-1 shows the parameters of different transpose convolutions of this generator model.

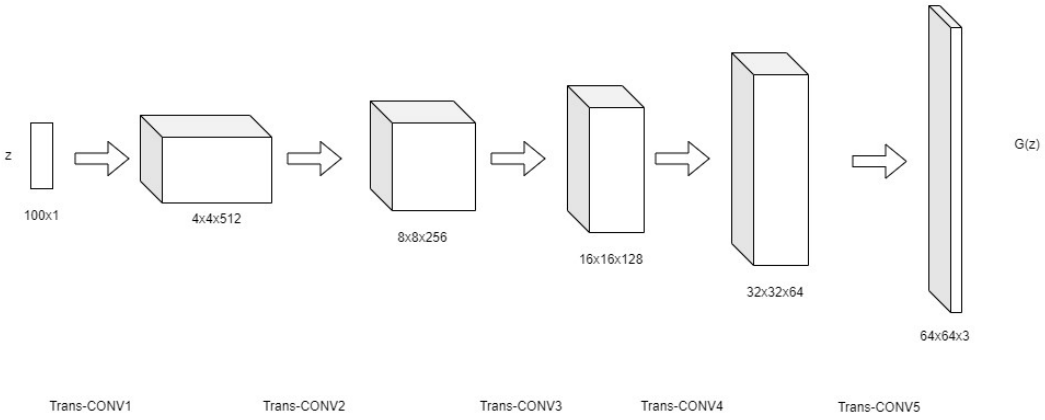

Figure 4-2: DCGAN generator as proposed by (Radford et al., n.d.) to generate image of 64x64x3 resolution

Table 4-1: Convolution parameters for DCGAN generator for image of 64x64x3 resolution

Transpose Convolution	Number of channels	Kernel size	Stride	Padding
Trans-Conv1	512	(4, 4)	(1, 1)	(0, 0)
Trans-Conv2	256	(4, 4)	(2, 2)	(1, 1)
Trans-Conv3	128	(4, 4)	(2, 2)	(1, 1)
Trans-Conv4	64	(4, 4)	(2, 2)	(1, 1)
Trans-Conv5	3	(4, 4)	(2, 2)	(1, 1)

Figure 4-3 shows the generator model generating 224x224x3 resolution image. The parameters of different transpose convolutions of this generator are shown below in Table 4-2,

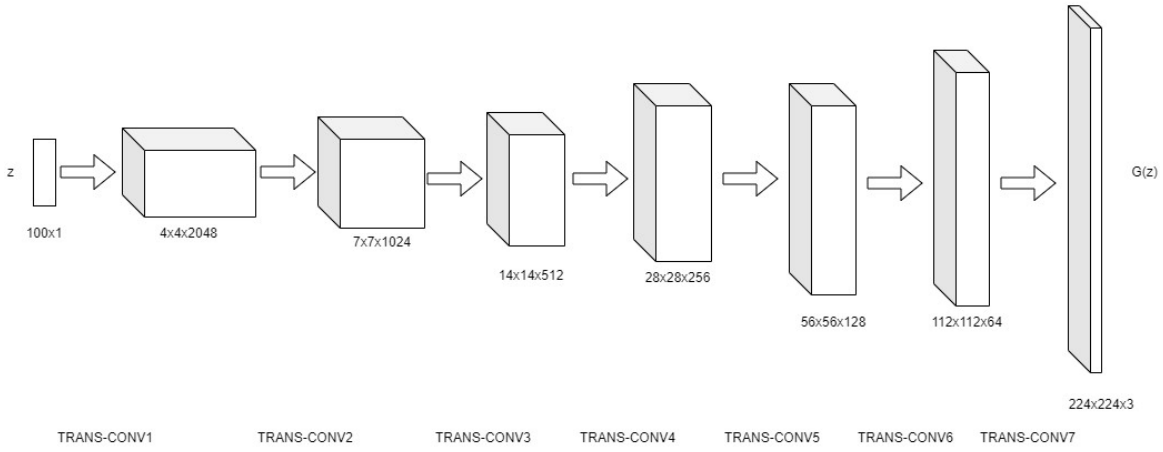

Figure 4-3: DCGAN generator to generate image of 224x224x3 resolution

Table 4-2: Convolution parameters for DCGAN generator for image of 224x224x3 resolution

Transpose Convolution	Number of channels	Kernel size	Stride	Padding
Trans-Conv1	2048	(4, 4)	(1, 1)	(0, 0)
Trans-Conv2	1024	(4, 4)	(1, 1)	(0, 0)
Trans-Conv3	512	(4, 4)	(2, 2)	(1, 1)
Trans-Conv4	256	(4, 4)	(2, 2)	(1, 1)
Trans-Conv5	128	(4, 4)	(2, 2)	(1, 1)
Trans-Conv6	64	(4, 4)	(2, 2)	(1, 1)
Trans-Conv7	3	(4, 4)	(2, 2)	(1, 1)

The discriminator model mirrors the generator model with series of convolution, scaling down the image at every step and finally ending with a 1x1x1 value. This is passed to a sigmoid activation function to predict if the image is real or fake. Figure 4-4 shows the discriminator model as proposed by (Radford et al., n.d.) for image of 64x64x3 resolution. It consists of series of convolution layers ending with a sigmoid activation function. Table 4-3 shows parameters of different convolution layers of this discriminator.

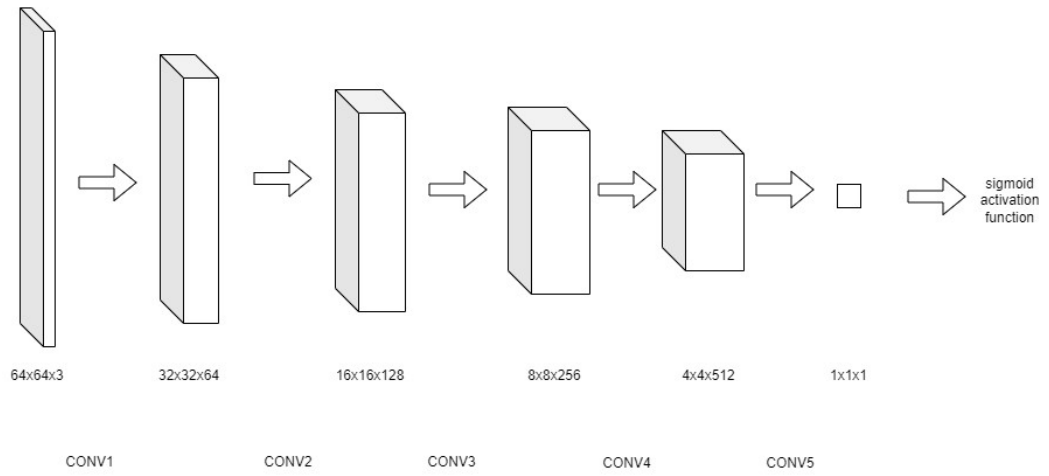

Figure 4-4: DCGAN discriminator as proposed by (Radford et al., n.d.) to discriminate image of 64x64x3 resolution

Table 4-3: Convolution parameters for DCGAN discriminator for 64x64x3 image

Convolution	Number of channels	Kernel size	Stride	Padding
Conv1	64	(4, 4)	(2, 2)	(1, 1)
Conv2	128	(4, 4)	(2, 2)	(1, 1)
Conv3	256	(4, 4)	(2, 2)	(1, 1)
Conv4	512	(4, 4)	(2, 2)	(1, 1)
Conv5	1	(4, 4)	(1, 1)	(0, 0)

Figure 4-5 shows the discriminator model for image of 224x224x3 resolution. Table 4-4 shows parameters of different convolution layers of this discriminator.

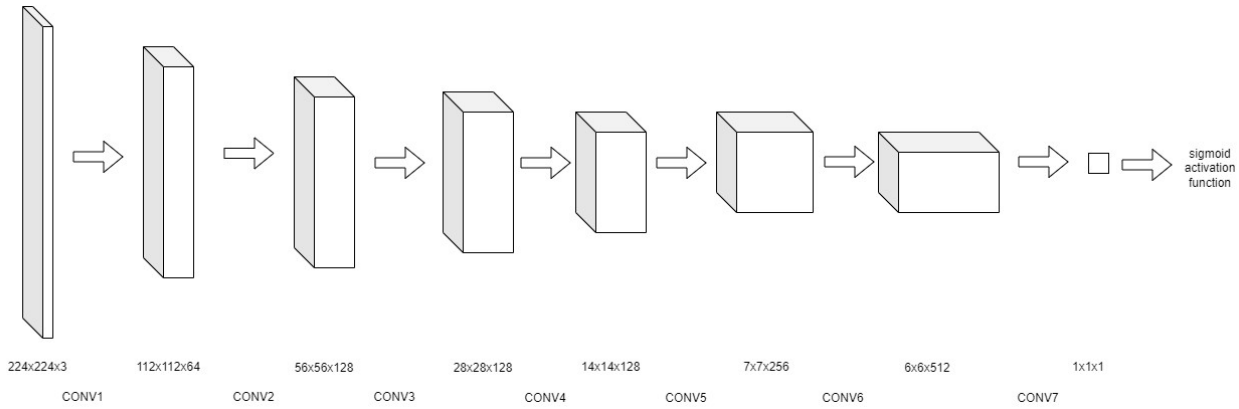

Figure 4-5: DCGAN discriminator for 224x224x3 image

Table 4-4: Convolution parameters for DCGAN discriminator for 224x224x3 image

Convolution	Number of channels	Kernel size	Stride	Padding
Conv1	64	(4, 4)	(2, 2)	(1, 1)
Conv2	128	(4, 4)	(2, 2)	(1, 1)
Conv3	128	(4, 4)	(2, 2)	(1, 1)
Conv4	128	(4, 4)	(2, 2)	(1, 1)
Conv5	256	(4, 4)	(2, 2)	(1, 1)
Conv6	512	(4, 4)	(1, 1)	(0, 0)
Conv7	1	(4, 4)	(1, 1)	(0, 0)

4.4 PGGAN with ISIC dataset

The PGGAN model (Karras et al., 2017) implemented is shown in Figure 4-6. The idea of PGGAN is to progressively learn the model to generate higher resolution images. Initially, the model learns to generate a malignant lesion image of resolution 4x4x3. The parameters for the convolution blocks are shown in Table 4-5. Next for 8x8x3 resolution image, the image output from previous model which generated 4x4x3 resolution image, is scaled up. The 2x scale up is achieved with interpolation feature of pytorch module. Subsequently the scaled-up image is passed through a toRGB block in one path and a convolution block followed by toRGB block in another path. The toRGB blocks reduces the number of channel and generate an image of resolution 8x8x3. The images from the two paths are faded in or combined with parameter α. This repeats for each image resolution, i.e., the image generated from model of previous resolution is scaled up with interpolation, it passes through a toRGB block and convolution block before it gets combined together.

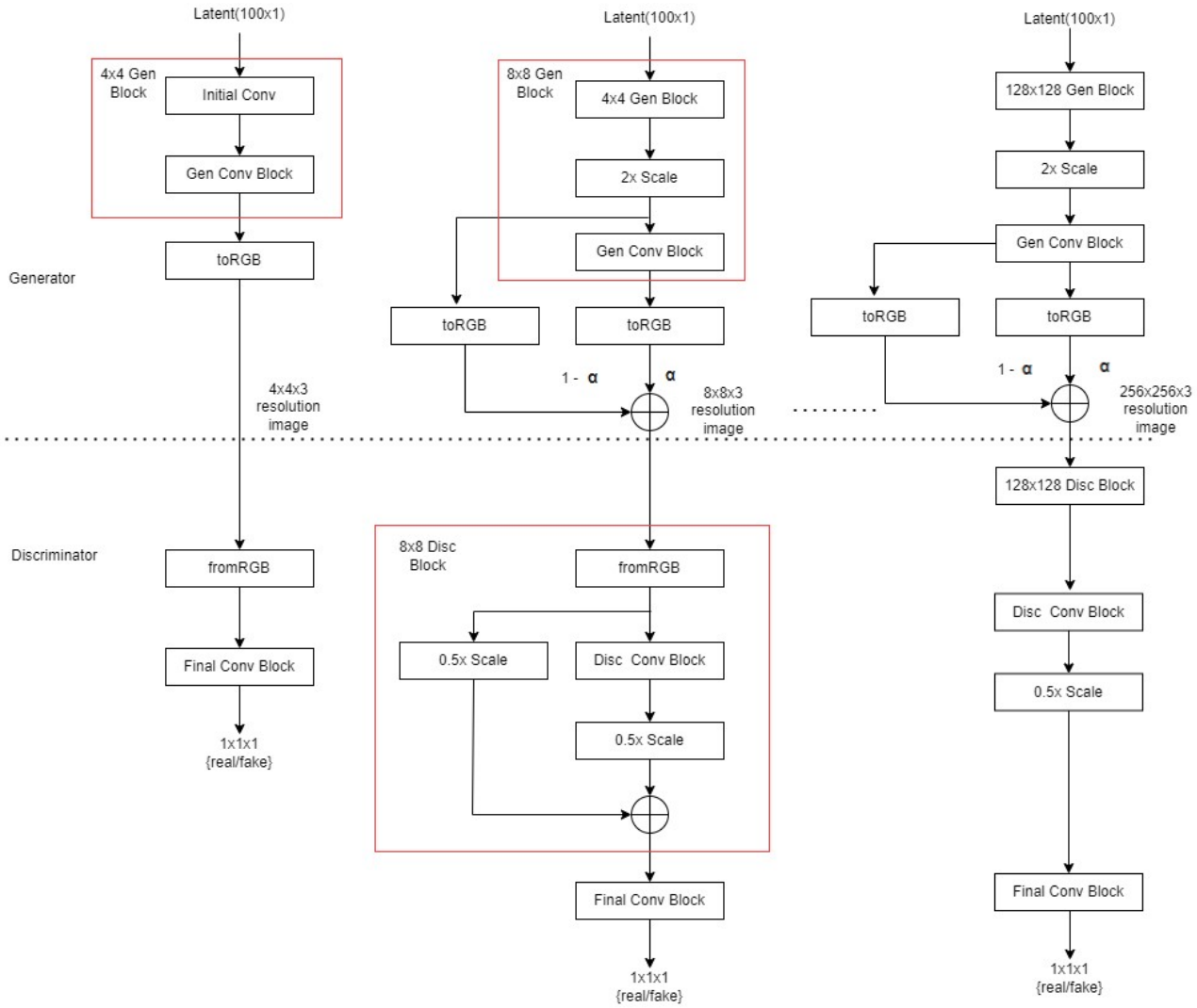

Figure 4-6: PGGAN for ISIC dataset as proposed by (Karras et al., 2017)

The parameters of different blocks of generator are shown below in Table 4-5,

Table 4-5: Convolution parameters for PGGAN generator

Block Name	Convolution	Number of channels	Kernel size	Stride	Padding
Initial Conv	Transpose convolution	256	(4, 4)	(1, 1)	(0, 0)
	Conv2d	256	(3, 3)	(1, 1)	(1, 1)
Gen Conv Block	Conv2d	256	(3, 3)	(1, 1)	(1, 1)
	Conv2d	256	(3, 3)	(1, 1)	(1, 1)
toRGB	Conv2d	3	(1, 1)	(1, 1)	(0, 0)

On discriminator side the image first goes through a fromRGB block where it is convolved to higher number of channels. The parameters of the convolutions are shown in Table 4-6. For 4x4x3 image, it goes through a final level of convolution operation which reduces the output to 1x1x1 to indicate where the image is real or fake. For higher resolution images, based on the stage of the PGGAN, the discriminator has a convolution block, shown as Disc Conv Block in Figure 4-6. The output of the fromRGB block is scaled down by half in one path and in other path it goes through the Disc Conv Block and then scales down. The scaling down is done with AvgPool2d operation with kernel size (2, 2) and stride of (2, 2). Then the output of the two paths is faded in or combined before it passed to the final level of convolution. The process is repeated for each level of the image resolution.

The parameters of different blocks of generator are shown below in Table 4-6,

Table 4-6: Convolution parameters for PGGAN discriminator

Block Name	Convolution	Number of channels	Kernel size	Stride	Padding
fromRGB	Conv2d	256	(1, 1)	(1, 1)	(0, 0)
Disc Conv Block - n	Conv2d	16 * n	(3, 3)	(1, 1)	(1, 1)
	Conv2d	16 * n	(3, 3)	(1, 1)	(1, 1)
Final Conv Block	Conv2d	256	(4, 4)	(1, 1)	(0, 0)

In PGGAN, we use gradient penalty as the loss function. Gradient penalty enforces a constraint that gradients of the discriminator output with respect to input will be unit norm. This is calculated by squaring the distance of each discriminator output from the ideal norm of 1 and then taking the mean of all the squared distances.

4.5 Classification with VGG16

There are many well-known state-of-art machine learning models for image classification. In this research work, we worked with VGG16 (Simonyan and Zisserman, 2015). We train VGG16 model twice for two input image resolution, 224x224x3 and 64x64x3. The pretrained VGG16 model works with 224x224x3 resolution image. So, for the image of resolution higher than 224x224x3, we can rescale the image to 224x224x3 and use the existing pretrained VGG16 model in pytorch with transfer learning technique.

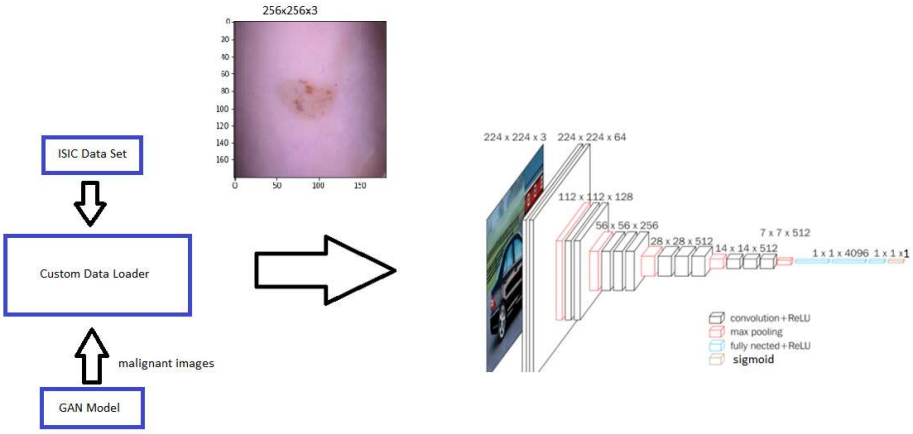

Figure 4-7: Image classification for VGG16 model as proposed by (Simonyan and Zisserman, 2015)

We set the required_grad parameter of all VGG16 parameter to false so that it does not modify the weight during training. Then the final layer is removed. We add a linear layer with 1x1x1 output (Figure 4-7). During training, a custom data loader module generates a batch of images either from ISIC dataset or from GAN mode. Then the batch of images are passed to the VGG16 model. The output of VGG16 is passed through a sigmoid activation function to indicate where the given image is melanoma or benign.

For 64x64x3 image, we modify the VGG16 implementation as shown in Figure 4-7. The number of layers is same. But because the input image size is smaller, the successive convolution layer reduces the size of the convolution output more. The fully connected layer is of dimension 1x1x2048 instead of 1x1x4096. Eventually we have a sigmoid activation function at the final layer to predict if the image is malignant or benign.

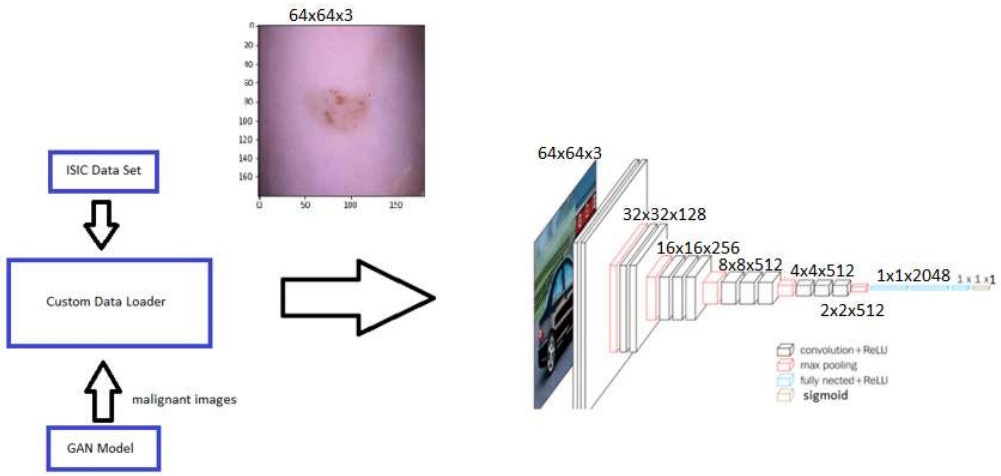

Figure 4-8: Image classification for VGG16 with 64x64x3 image resolution

4.5.1 Custom Data Loader

The custom data loader is an important module in the VGG16 training pipeline. It is responsible for feeding skin lesion images to the VGG16 model during training. For benign images, the custom data loader will pick images from the ISIC dataset. However, for malignant images, custom data loader will pick images initially from ISIC dataset and then generate images from GAN model. Custom data loader ensures that there is a balance of both classes of images during training.

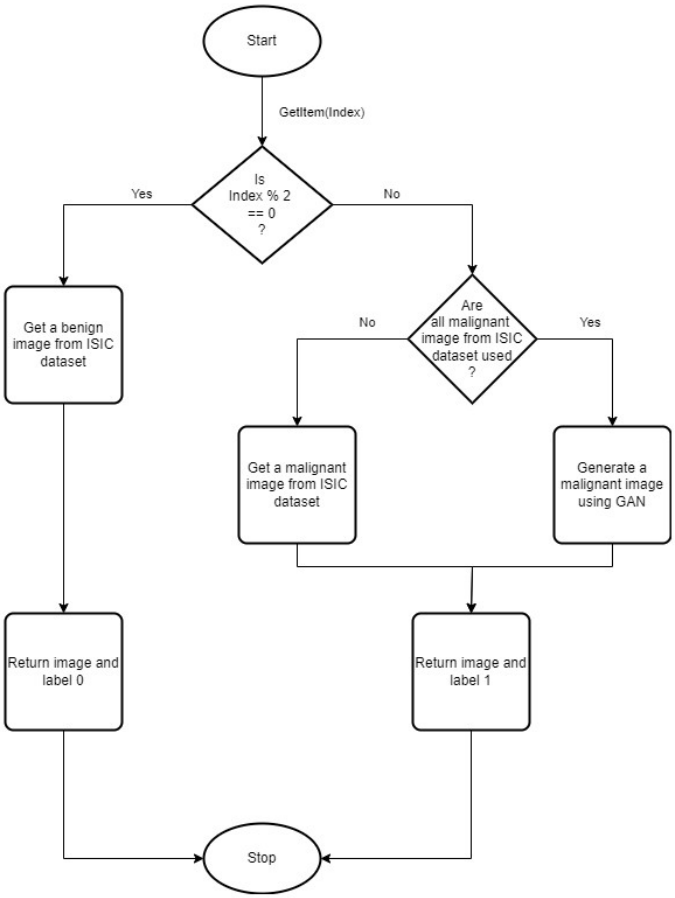

Figure 4-9: Custom data loader flowchart

Figure 4-9 shows the flowchart of the custom data loader. While for benign images, it will always pick image from ISIC dataset, for malignant images, it will pick malignant images from ISIC dataset till all the malignant images get exhausted. Then it will use GAN model to synthetically generate malignment images. Custom data loader will provide the images and label to VGG16 model. For benign image, it will provide label as 0 while for malignment images, it will provide label as 1.

4.6 Model parameters

In Table 4-7 summarizes the hyper-parameters, loss functions, optimizer used in all the machine learning models training as part of this research work.

Table 4-7: Model parameters for learning

Models trained	LR	Epochs	Loss function	Optimizer
DCGAN for 64x64x3 image	0.0002	1500	BCELoss	Adam
DCGAN for 224x224x3 image	0.0002	1500	BCELoss	Adam
PGGAN for 256x256x3 image	0.001	1000	Gradient penalty	Adam
VGG for 64x64x3 image	0.0001	100	BCEWithLogitsLoss	Adam
VGG for 224x224x3 image	0.0001	100	BCEWithLogitsLoss	Adam

Chapter 5: Results and Evaluation

5.1 Introduction

In this chapter we will discuss the results of our GAN experiments on ISIC dataset. We will first discuss the performance of VGG16 model on ISIC dataset without any modification to the model. This defines the baseline for our work. We will then discuss performance with data augmentation done with affine transformation. This defines the baseline of performance with augmentation. We then discuss GAN performance in generating synthetic images and discuss the classification performance after the dataset is augmented with GAN generated images.

5.2 VGG16 experiments on ISIC dataset

In the first experiment of our research, we baseline VGG16 performance on ISIC dataset. In one experiment we train VGG16 on ISIC dataset without any augmentation. In second experiment we train VGG16 on an augmented ISIC dataset where basic affine transformation such as rotation and scale were performed.

5.2.1 VGG16 performance on ISIC dataset

In our experiment, we used a pretrained VGG16 model and modified the final layer for two class. We train the model for 10 epochs with the weights for VGG16 model fixed, i.e., we only train weights for the final layer. After the model is validated on the validation dataset, the confusion matrix of the predicted class is shown in Table 5-1

Table 5-1: Confusion matrix of ISIC dataset

		Actual	
		Malignant	Benign
Prediction	Malignant	2 (TP)	144 (FN)
	Benign	3 (FP)	8132 (TN)

The classification performance in medical data is measured with parameters like precision, recall, F1 score, accuracy etc as explained in chapter 3.4. Precision represents how many of the predicted positive cases are actually positive. Recall presents how many of the actual malignant cases are predicted positive.

Table 5-2: Classification performance on ISIC dataset

Sensitivity or Recall	0.013
Specificity	0.99
Precision	0.4
F1	0.02
Accuracy	0.98

We observe that both precision and recall are low. Low recall is a concern because it implies that model is not able to detect true malignant melanoma images.

5.2.2 VGG16 performance on augmented ISIC dataset

In our next experiment, we augment the malignant class of ISIC dataset with affine transformation such as rotate and zoom. Table 5-3 shows the confusion matrix of VGG16 prediction after it is trained on the augmented dataset.

Table 5-3: Confusion matrix on the augmented ISIC dataset

		Actual	
		Malignant	Benign
Prediction	Malignant	5 (TP)	141 (FN)
	Benign	2 (FP)	8133 (TN)

Table 5-4 shows the classification performance on the predicted data. We observe that precision has improved significantly with augmented data. But recall improved only marginally.

Table 5-4: Classification performance on augmented dataset

Sensitivity or Recall	0.034
Specificity	0.999
Precision	0.714
F1	0.065
Accuracy	0.982

In summary, we observe that on plain ISIC dataset the classification performance is very poor. It does not seem to learn meaningful features of the malignant melanoma images. Affine data augmentation to ISIC dataset shows improvement but recall score is not good.

5.3 GAN experiments

We experimented in two GAN models, DCGAN and PGGAN. With DCGAN, we explored two models to create 64x64x3 and 224x224x3 resolution images of malignant melanoma. With PGGAN, we made model to create 256x256x3 resolution images of malignant melanoma. In this chapter we share the results of these experiments.

5.3.1 DCGAN

Figure 5-1 shows the 224x224x3 resolution melanoma images generated with DCGAN model as described in chapter 4.3. We observe that the generated images have a checker board pattern and the images are not very clear. The checker board pattern was also observed by (Pollastri et al., n.d.).

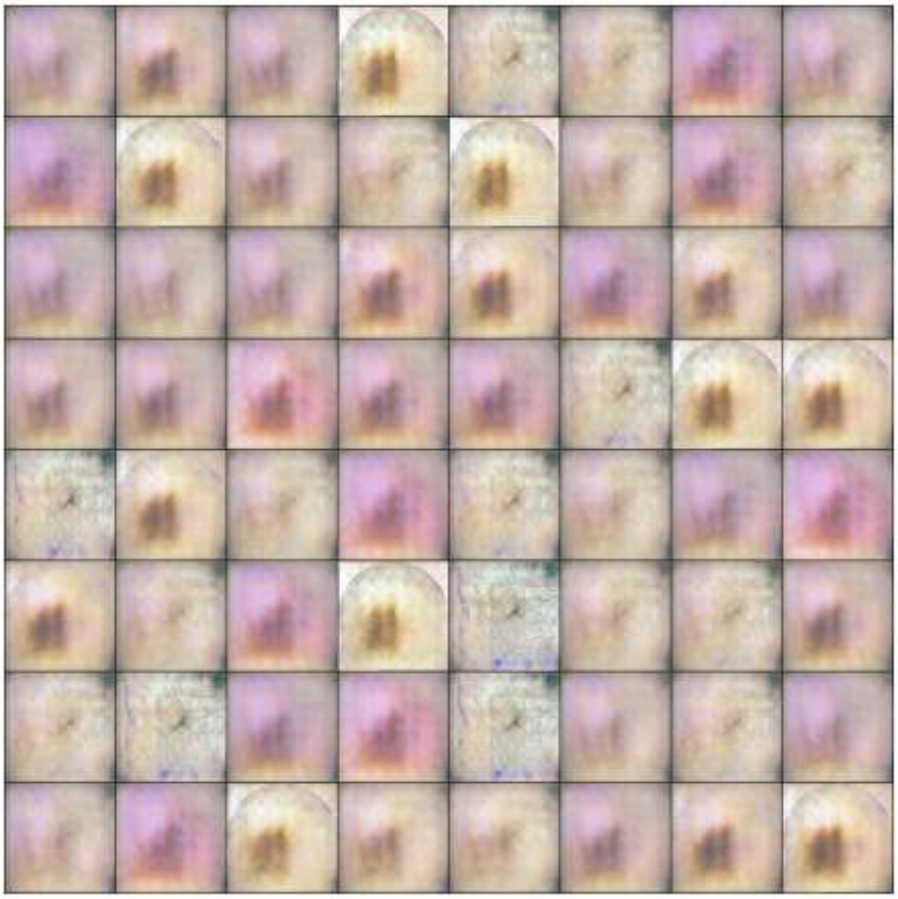

Figure 5-1: Generated 224x224x3 resolution malignant melanoma images from DCGAN

Figure 5-2 shows the 64x64x3 resolution melanoma images generated with DCGAN model. We observe that the generated images have a checker board pattern like in 224x224x3 resolution image. However, the images are clearer than 224x224x3 resolution.

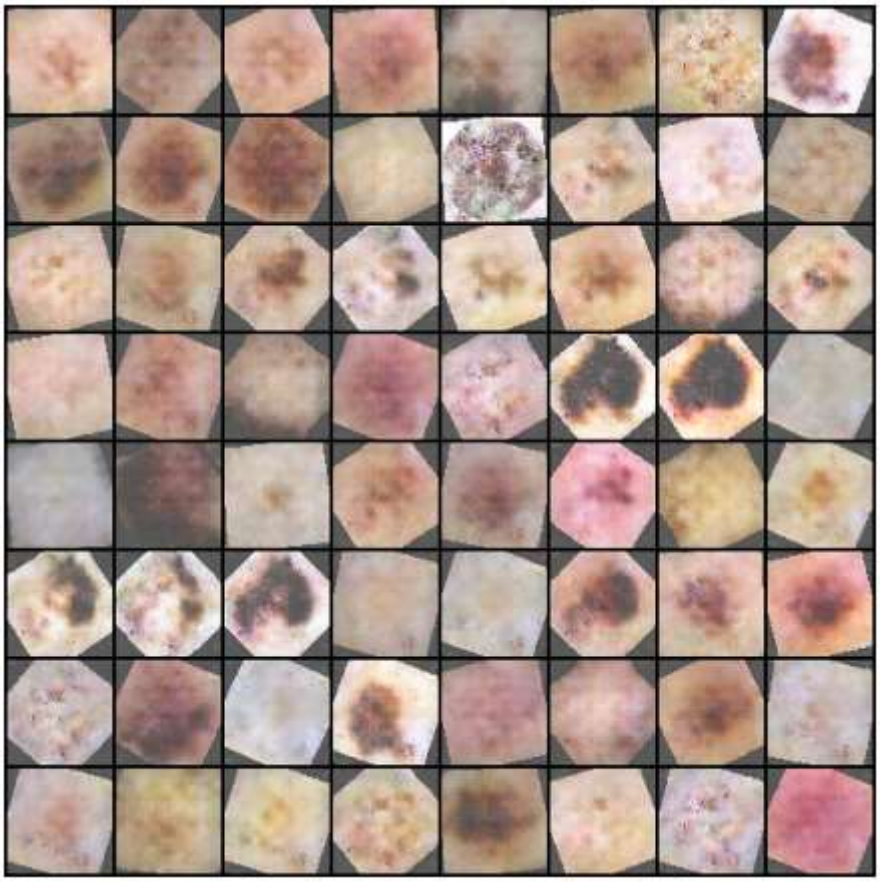

Figure 5-2: Generated 64x64x3 resolution malignant melanoma images from DCGAN

We captured the generator and discriminator loss during the training in both DCGAN models. Figure 5-3 shows the generator and discriminator loss during training DCGAN for 224x224x3 resolution image.

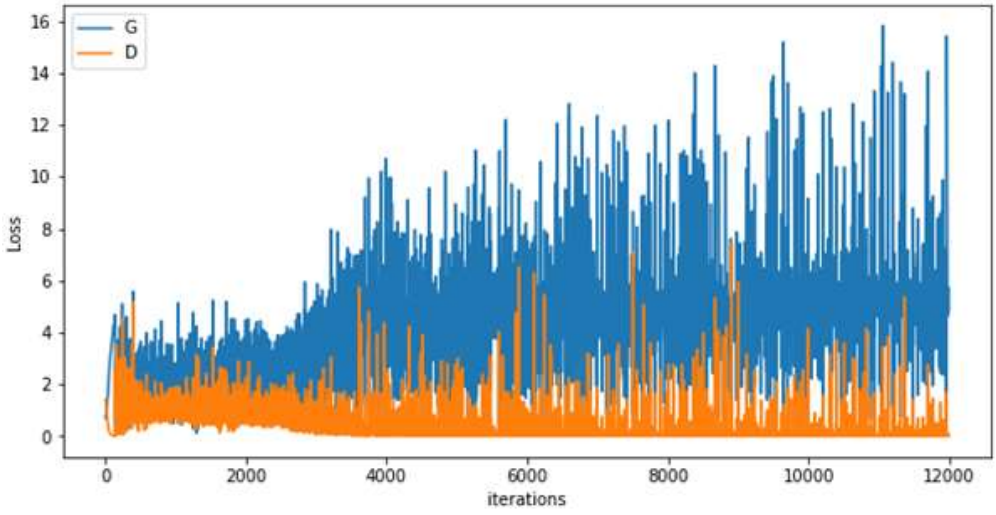

Figure 5-3: Generator and discriminator loss during training DCGAN for 224x224x3 resolution image

Figure 5-4 shows the generator and discriminator loss during training DCGAN for 64x64x3 resolution image.

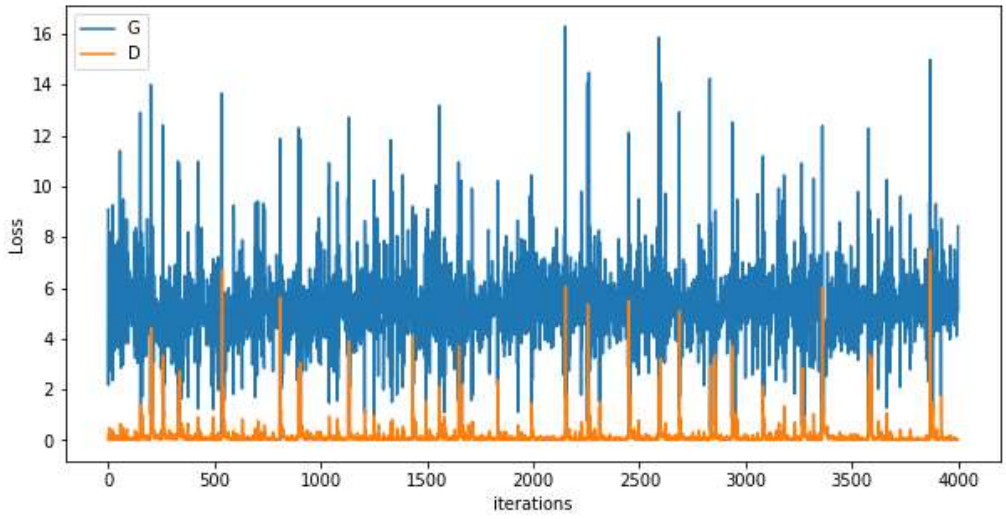

Figure 5-4: Generator and discriminator loss during training DCGAN for 64x64x3 resolution image

From both Figure 5-3 and Figure 5-4, it is evident that loss is not stable. Ideally in machine learning algorithm, we expect the loss to gradually reduce. But in this case, it fluctuates a lot between high and low value. This indicates that we need to work on the loss and optimization function of DCGAN.

5.3.2 PGGAN

From Figure 5-5 to Figure 5-11, show the progressive generation of melanoma images with PGGAN model as described in section 4.4. We observe that PGGAN does not learn to generate good melanoma images till it reaches resolution of 256x256x3. We also observe that compared to DCGAN, PGGAN generated much better quality of the high-resolution images.

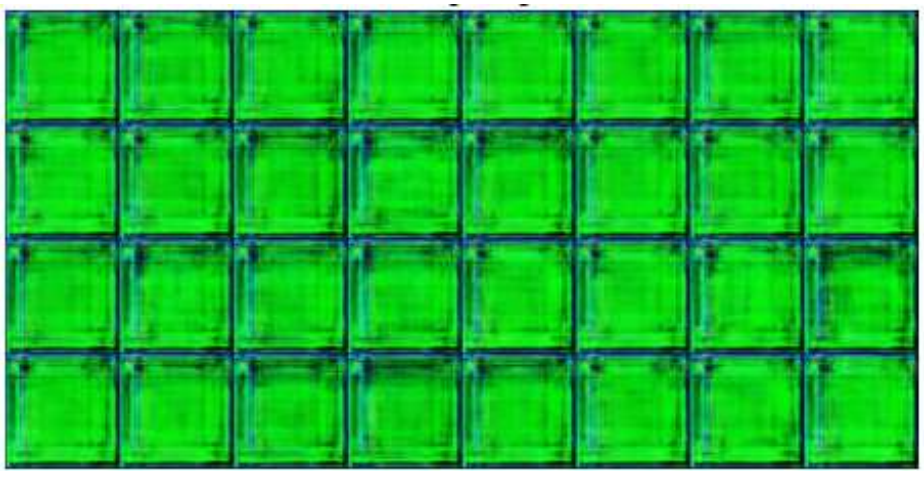

Figure 5-5: Generated skin lesion images from PGGAN at 4x4x3 resolution

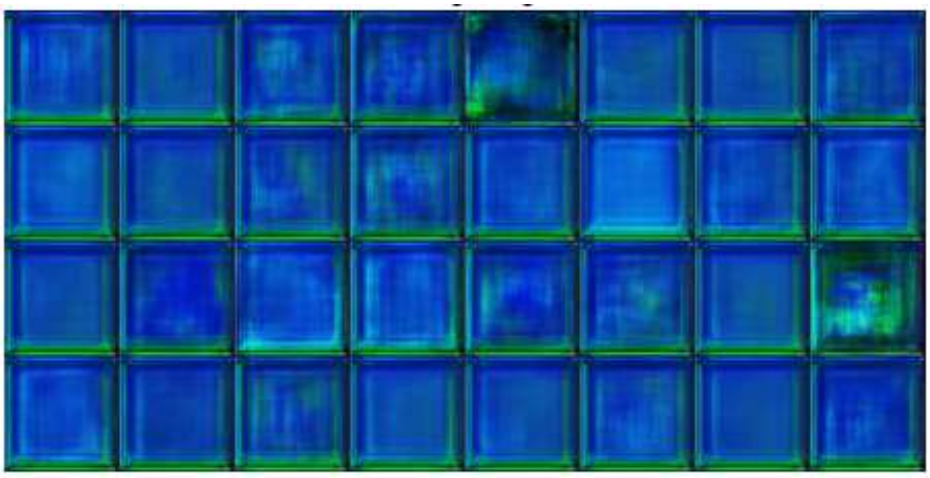

Figure 5-6: Generated skin lesion images from PGGAN 8x8x3 resolution

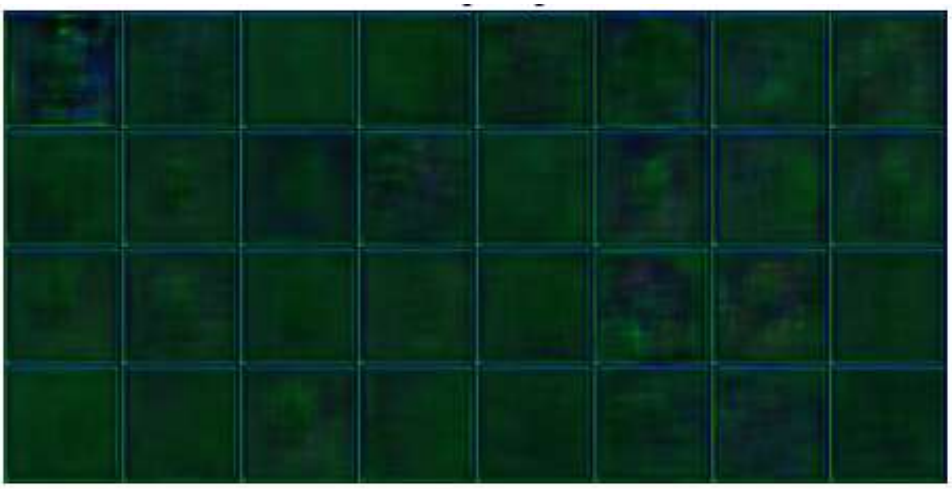

Figure 5-7: Generated skin lesion images from PGGAN 16x16x3 resolution

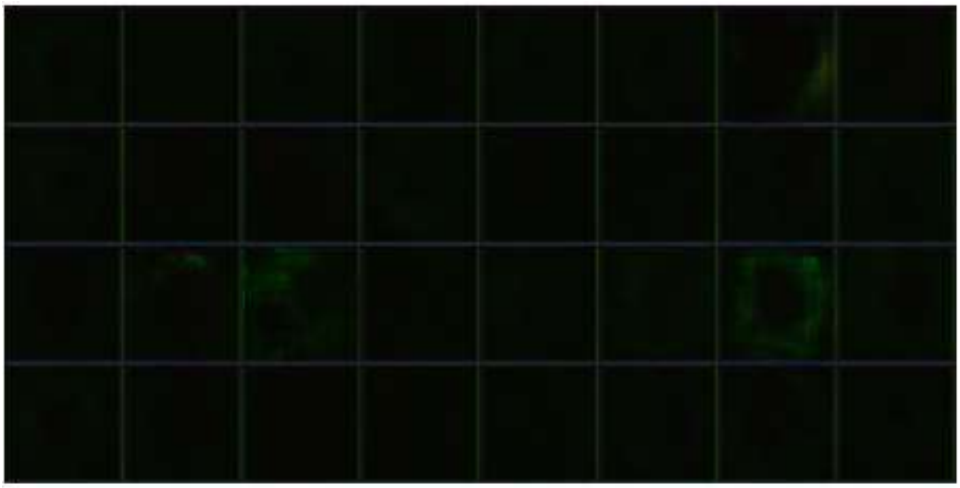

Figure 5-8: Generated skin lesion images from PGGAN 32x32x3 resolution

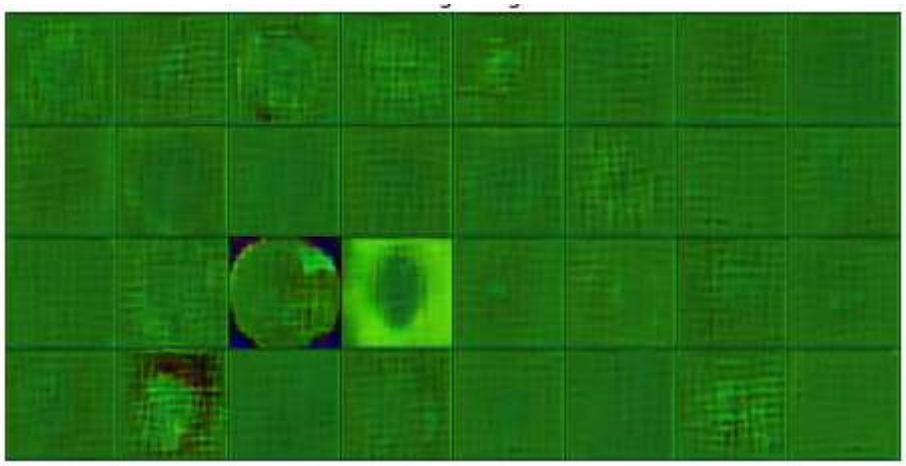

Figure 5-9: Generated skin lesion images from PGGAN 64x64x3 resolution

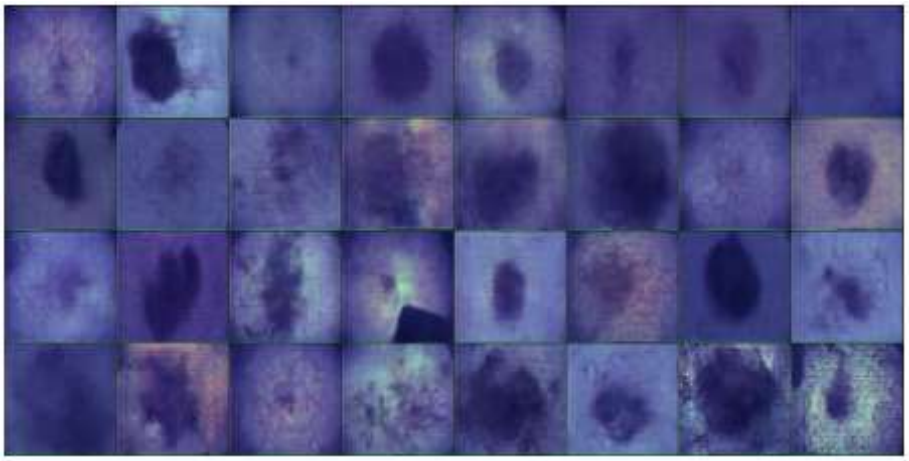

Figure 5-10: Generated skin lesion images from PGGAN 128x128x3 resolution

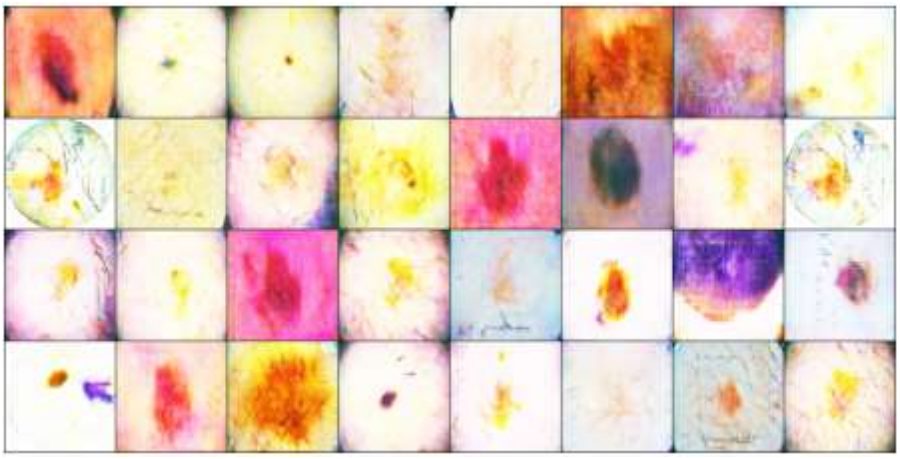

Figure 5-11: Generated skin lesion images from PGGAN 256x256x3 resolution

The PGGAN model has been used to create lesion images up to 256x256x3 resolution. Further attempt to create higher resolution image such as 512x512x3 failed due to lack of CUDA memory, even with very small batch size.

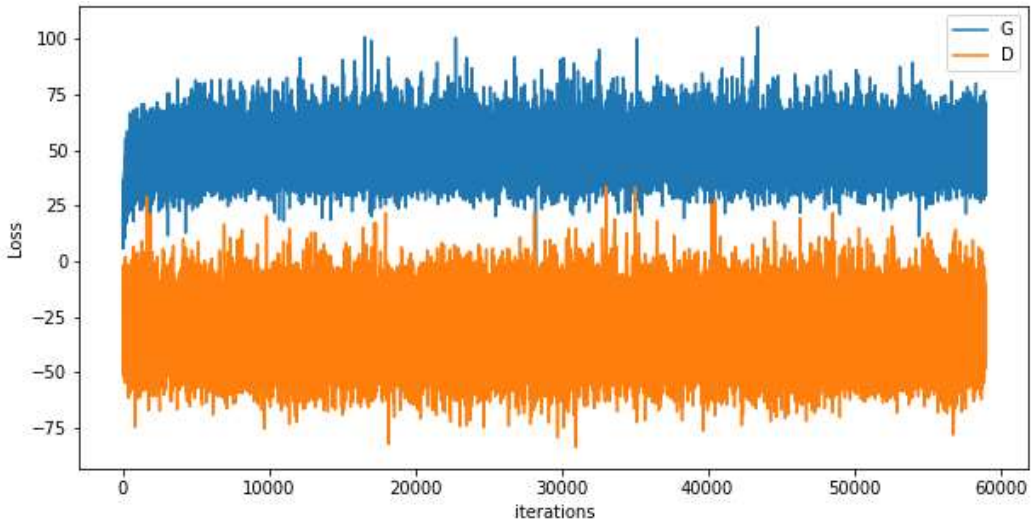

Figure 5-12: Generator and discriminator loss during training PGGAN for 256x256x3 resolution image

Figure 5-12 shows the generator and discriminator loss during PGGAN training. Like in DCGAN, in PGGAN also we observe fluctuations in loss of both generator and discriminator.

5.4 VGG16 experiments on ISIC dataset augmented with GAN

We now discuss the classification result when the ISIC dataset is augmented with DCGAN and PGGAN. We used 64x64x3 resolution image from DCGAN and 256x256x3 resolution image from PGGAN.

5.4.1 Augmentation with 64x64x3 image from DCGAN

Table 5-5 shows the confusion matrix and Table 5-6 shows the classification performance on ISIC dataset augmented with 64x64x3 image from DCGAN. For classification, we modify and train VGG16 model as explained in section 4.5.

Table 5-5: Confusion matrix on the augmented ISIC dataset with 64x64x3 image from DCGAN

		Actual	
		Malignant	Benign
Prediction	Malignant	4 (TP)	142 (FN)
	Benign	2 (FP)	8133 (TN)

Table 5-6: Classification performance on augmented dataset with 64x64x3 image from DCGAN

Sensitivity or Recall	0.027
Specificity	0.999
Precision	0.666
F1	0.052
Accuracy	0.982

We observe no significant improvement in classification performance compared to augmentation with basic affine transformation.

5.4.2 Augmentation with 256x256x3 image from PGGAN

Table 5-7 shows the confusion matrix and Table 5-8 shows the classification performance on ISIC dataset augmented with 256x256x3 images from PGGAN. For classification, we use the pre-trained VGG16 model as explained in chapter 4.5.

Table 5-7: Confusion matrix on the augmented ISIC dataset with 256x256x3 image from PGGAN

		Actual	
		Malignant	Benign
Prediction	Malignant	5 (TP)	141 (FN)
	Benign	3 (FP)	8132 (TN)

Table 5-8: Classification performance on augmented dataset with 256x256x3 image from PGGAN

Sensitivity or Recall	0.034
Specificity	0.999
Precision	0.714
F1	0.065
Accuracy	0.982

In this experiment too, we observe no significant improvement in classification performance compared to augmentation with basic affine transformation. The overall summary of the classification summary is shown in Figure 5-13. There is some improvement in the performance after basic data augmentation. But no significant change is seen with GAN based augmentation.

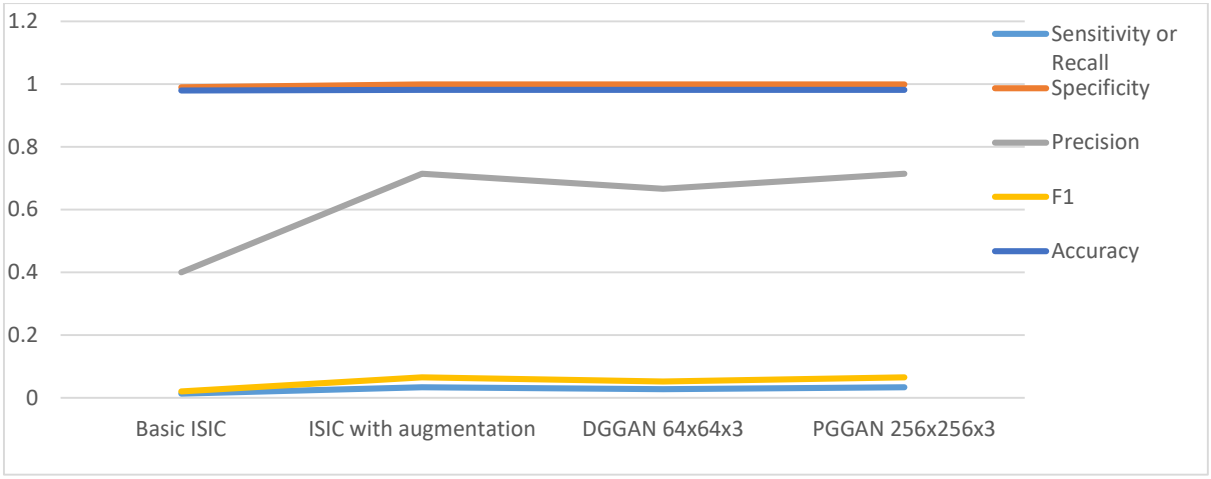

Figure 5-13: Classification performance summary

Chapter 6: Conclusion and Recommendations

6.1 Introduction

In this research we explored different methods with GAN to improve classification performance of skin lesion images. In this chapter we summarize the findings.

6.2 Discussion and Conclusion

The challenge with ISIC dataset was high data imbalance with a smaller number of malignant melanoma images. Without any data augmentation, we observed that precision and recall score is very poor. With data augmentation with affine transformation, the precision and recall score improved marginally. In this research, we wanted to explore more data augmentation approach with GAN. We started this research work with following three objectives,

- To explore state-of-art GAN architectures
- To propose a GAN architecture suitable for augmenting ISIC dataset
- To evaluate the performance of the state-of-art classification model after GAN based data augmentation

In this research, we have explored evolution of various GAN architectures starting from DCGAN, WGAN, to PGGAN and StyleGAN. We explained how these GAN architectures are used in medical images. We found that DCGAN and PGGAN are most popular GAN architectures in medical image augmentation task. We also discussed that mode collapse is the most common issue faced by GAN based data augmentation.

In our implementation discussion, we have showed how we implemented the DCGAN and PGGAN models for different lesion image resolution. DCGAN was able to generate good lesion images at lower resolution but at higher resolution, DCGAN performed very poorly. PGGAN performed very poorly for lower resolution images. At 256x256x3 resolution, PGGAN started to generate good melanoma lesion images. We showed the unstable training errors in both GANs. This pointed that further work is needed towards finding a better loss functions in both DCGAN and PGGAN when generating images from ISIC dataset.

In the result discussion, we have shown the classification performance of VGG16 model with ISIC dataset augmented with both DCGAN and PGGAN. We did not find any remarkable improvement with the GAN model implemented for lesion images augmentation.

6.3 Contributions

In this research work, we implemented state-of-art GAN models such as DCGAN and PGGAN for ISIC dataset. We also modified VGG16 classification to work on low resolution image.

6.4 Future Work

We need to explore more GAN models for lesion images to create higher resolution images. We can investigate whether adding soft attention to the GAN generator model helps to create more realistic lesion images. We also need to investigate the loss function in GAN model.

References

Apalla, Z., Nashan, D., Weller, R.B. and Castellsagué, X., (2017) *Skin Cancer: Epidemiology, Disease Burden, Pathophysiology, Diagnosis, and Therapeutic Approaches*. *Dermatology and Therapy*,

Arjovsky, M., Chintala, S. and Bottou, L., (2017) Wasserstein GAN. [online] Available at: http://arxiv.org/abs/1701.07875.

Benedetti, P., Perri, D., Simonetti, M., Gervasi, O., Reali, G. and Femminella, M., (2021) Skin Cancer Classification using Inception Network and Transfer Learning. [online] Available at: http://arxiv.org/abs/2111.02402.

Bowles, C., Chen, L., Guerrero, R., Bentley, P., Gunn, R., Hammers, A., Dickie, D.A., Hernández, M.V., Wardlaw, J. and Rueckert, D., (2018) GAN Augmentation: Augmenting Training Data using Generative Adversarial Networks. [online] Available at: http://arxiv.org/abs/1810.10863.

Bowles, C., Chen, L., Guerrero, R., Bentley, P., Gunn, R., Hammers, A., Dickie, D.A., Valdés Hernández, M., Wardlaw, J. and Rueckert, D., (n.d.) *GAN Augmentation: Augmenting Training Data using Generative Adversarial Networks*.

Codella, N., Rotemberg, V., Tschandl, P., Celebi, M.E., Dusza, S., Gutman, D., Helba, B., Kalloo, A., Liopyris, K., Marchetti, M., Kittler, H. and Halpern, A., (2019) Skin Lesion Analysis Toward Melanoma Detection 2018: A Challenge Hosted by the International Skin Imaging Collaboration (ISIC). [online] Available at: http://arxiv.org/abs/1902.03368.

Cubuk, E.D., Zoph, B., Vasudevan, V. and le Google Brain, Q. v, (n.d.) *AutoAugment: Learning Augmentation Strategies from Data*. [online] Available at: https://pillow.readthedocs.io/en/5.1.x/.

Datta, S.K., Shaikh, M.A., Srihari, S.N. and Gao, M., (2021) Soft-Attention Improves Skin Cancer Classification Performance. [online] Available at: http://arxiv.org/abs/2105.03358.

Deng, L., (2012) The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*.

Denton, E., Chintala, S., Szlam, A. and Fergus, R., (n.d.) *Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks*.

Frid-Adar, M., Klang, E., Amitai, M., Goldberger, J. and Greenspan, H., (2018) Synthetic Data Augmentation using GAN for Improved Liver Lesion Classification. [online] Available at: http://arxiv.org/abs/1801.02385.

Gessert, N., Nielsen, M., Shaikh, M., Werner, R. and Schlaefer, A., (2019) Skin Lesion Classification Using Ensembles of Multi-Resolution EfficientNets with Meta Data. [online] Available at: http://arxiv.org/abs/1910.03910.

Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y., (2014) Generative Adversarial Networks. [online] Available at: http://arxiv.org/abs/1406.2661.

Han, C., Rundo, L., Araki, R., Furukawa, Y., Mauri, G., Nakayama, H. and Hayashi, H., (2019a) Infinite Brain MR Images: PGGAN-based Data Augmentation for Tumor Detection. [online] Available at: http://arxiv.org/abs/1903.12564.

Han, C., Rundo, L., Araki, R., Nagano, Y., Furukawa, Y., Mauri, G., Nakayama, H. and Hayashi, H., (2019b) Combining Noise-to-Image and Image-to-Image GANs: Brain MR Image Augmentation for Tumor Detection. [online] Available at: http://arxiv.org/abs/1905.13456.

Jian, K.H.X.Z.S.R., (1996) Deep Residual Learning for Image Recognition arXiv:1512.03385v1. *Enzyme and Microbial Technology*, [online] 192, pp.107–117. Available at: http://image-net.org/challenges/LSVRC/2015/ [Accessed 1 Apr. 2022].

Karras, T., Aila, T., Laine, S. and Lehtinen, J., (2017) Progressive Growing of GANs for Improved Quality, Stability, and Variation. [online] Available at: http://arxiv.org/abs/1710.10196.

Karras, T., Aila, T., Laine, S. and Lehtinen, J., (n.d.) *PROGRESSIVE GROWING OF GANS FOR IMPROVED QUALITY, STABILITY, AND VARIATION*. [online] Available at: https://youtu.be/G06dEcZ-QTg.

Karras, T., Laine, S. and Aila, T., (2018) A Style-Based Generator Architecture for Generative Adversarial Networks. [online] Available at: http://arxiv.org/abs/1812.04948.

Mariani, G., Scheidegger, F., Istrate, R., Bekas, C. and Malossi, C., (2018) BAGAN: Data Augmentation with Balancing GAN. [online] Available at: http://arxiv.org/abs/1803.09655.

Mayoclinic, (2022) *Melanoma*. [online] Available at: https://www.mayoclinic.org/diseases-conditions/melanoma/symptoms-causes/syc-20374884 [Accessed 20 Mar. 2022].

Mazzia, V., Salvetti, F., Chiaberge, M. and Data and, B., (123AD) Efficient-CapsNet: capsule network with self-attention routing. *Scientific Reports* |, [online] 11, p.14634. Available at: https://doi.org/10.1038/s41598-021-93977-0.

Mirza, M. and Osindero, S., (n.d.) *Conditional Generative Adversarial Nets*.
Pollastri, F., Bolelli, F., Paredes, R. and Grana, C., (n.d.) *Augmenting data with GANs to segment melanoma skin lesions*.

Radford, A., Metz, L. and Chintala, S., (n.d.) *UNSUPERVISED REPRESENTATION LEARNING WITH DEEP CONVOLUTIONAL GENERATIVE ADVERSARIAL NETWORKS*.

Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A. and Chen, X., (2016) Improved Techniques for Training GANs. [online] Available at: http://arxiv.org/abs/1606.03498.

Shamsolmoali, P., Zareapoor, M., Shen, L., Sadka, A.H. and Yang, J., (2020) Imbalanced Data Learning by Minority Class Augmentation using Capsule Adversarial Networks. [online] Available at: http://arxiv.org/abs/2004.02182.

Sicas Medical Image Repository, (2016) *The Brain Tumor Image Segmentation Challenge*. [online] Available at: https://www.smir.ch/BRATS/Start2015 [Accessed 1 Apr. 2022].

Simonyan, K. and Zisserman, A., (2015) *VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION*. [online] Available at: http://www.robots.ox.ac.uk/.

Stanley, R.J., Stoecker, W. v and Moss, R.H., (2007) A relative color approach to color discrimination for malignant melanoma detection in dermoscopy images.

Szegedy, C., Ioffe, S., Vanhoucke, V. and Alemi, A., (2016) Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. [online] Available at: http://arxiv.org/abs/1602.07261.

Tanaka, F.H.K. dos S. and Aranha, C., (2019) Data Augmentation Using GANs. [online] Available at: http://arxiv.org/abs/1904.09135.

Tschandl, P., Rosendahl, C. and Kittler, H., (2018) Data descriptor: The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Scientific Data*, 5.

Yao, P., Shen, S., Xu, M., Liu, P., Zhang, F., Xing, J., Shao, P., Kaffenberger, B. and Xu, R.X., (n.d.) *Single Model Deep Learning on Imbalanced Small Datasets for Skin Lesion Classification*. [online] Available at: https://github.com/yaopengUSTC/mbit-skin-cancer.git.

Appendix A: Research Proposal

1. Background

Malignant melanoma (MM) and non-melanoma skin cancer (NMSC) are the common form of malignancy among Caucasian (Apalla et al., 2017). Both NMSC and MM cases are increasing annually at a rate of 0.6% over 50 years. The number of cases of skin melanoma represents 4.5% of all cancer cases in 2016. Melanoma could spread deeper into the skin or other parts of the body and could become fetal. So early detection of malignant melanoma is important and it is achieved with machine learning techniques (Stanley et al., 2007).

Towards machine learning research on skin cancer, international skin imaging collaboration (ISIC) conducts challenges for three types of tasks (Codella et al., 2019). Task 1 is lesion segmentation, task 2 is lesion attribute detection and task 3 is lesion disease classification. One of the challenges of using modern machine learning techniques such as deep neural networks for classification task is availability of large and balance data. In ISIC dataset there is high imbalance among images of benign and malignant types. Such imbalance dataset is not suitable for classification with deep neural network. There are existing data augmentation approach like random crop, flips etc. which are often applied to improve the number of training data, but they are simply duplicating the existing training features. It does not add enough variety to the training data to learn the generic features of skin cancer images.

Generative adversarial network or GAN (Goodfellow et al., 2014; Salimans et al., 2016) offers a method to learn features from a dataset to generate synthetic images with appearance similar to the real images. In this research proposal we will explore GAN based data augmentation method to improve skin cancer classification performance of ISIC dataset.

2. Related Work

GAN provides a way to learn deep representations of the training data. GAN consists of two deep neural network model, known as generator and discriminator, which contest with each other during learning. Generator generates fake images based on a noise vector as input. Discriminator tries to identify whether a given image is real or fake. Generator will try to create an image such that discriminator fails to identify it as fake image. The overall operation is shown in Figure A-1

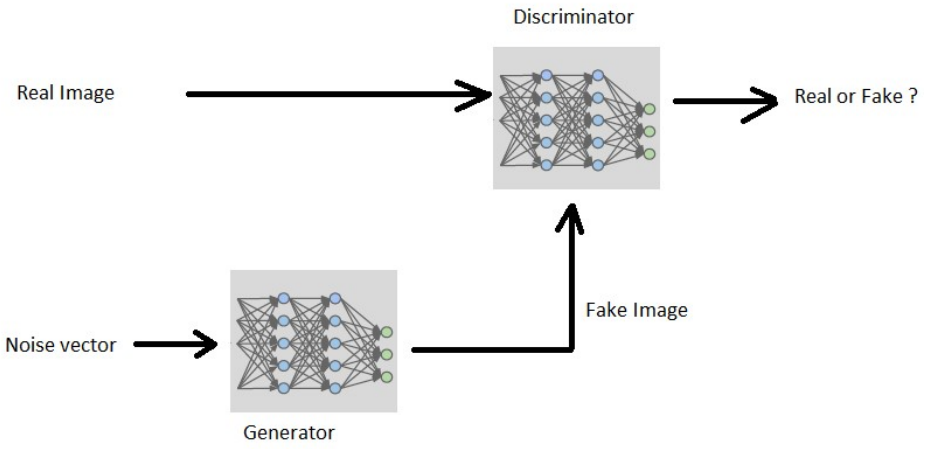

Figure A-1: GAN architecture

Generator eventually learns a conditional probability distribution P(X) of the training images in the dataset, where X is the image (Figure A-2).

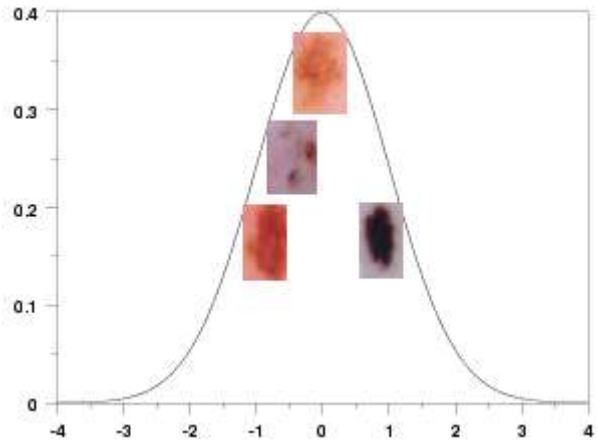

Figure A-2: GAN learns a probability distribution of images for each class

At the end of GAN training, the generator model can generate realistic looking images from a random noise vector.

(Tanaka and Aranha, 2019) did two types of experiment with GAN. In first experiment, they used only GAN generated synthetic images in place of original images during model training on breast cancer Wisconsin dataset. In second experiment, they used GAN generated synthetic data for imbalance data on credit card fraud detection database and compared the performance with other data balancing technique such as SMOTE, ADASYN. In first experiment, they observed that accuracy of the model was retained when the model was trained on only synthetic images. In the second experiment, they observed that model trained on GAN generated balanced dataset, had better accuracy and precision compared to balanced dataset with SMOTE and ADASYN.

Different variations of GAN have been adopted for data augmentation task in various medical image datasets. (Bowles et al., n.d.) showed PGGAN (Karras et al., n.d.) based augmentation can be used on CT image dataset. PGGAN progressively grows the generator and discriminator, from low resolution to high resolution image. Using PGGAN (Bowles et al., n.d.) found that there is an improvement in dice similarity coefficient in the CT image dataset segmentation task. (Han et al., 2019b) also used PGGAN based augmentation to boost sensitivity on MR dataset classification.

(Frid-Adar et al., 2018) used DCGAN (Radford et al., n.d.) for data augmentation and improved both sensitivity and specificity of the liver lesion classification task. (Pollastri et al., n.d.) used LAPGAN (Denton et al., n.d.) to augment the ISIC dataset and observed improvement in Jaccard similarity index for the segmentation task. LAPGAN uses a series of convolutional neural networks with Laplacian pyramid framework to generate images. (Mirza and Osindero, n.d.) introduced conditional GAN to produce a class specific image from a random noise vector and class label.

Training GAN for data augmentation has its challenges. With imbalance dataset, traditional GAN may end up producing images of the dominant class. This problem is known as mode collapse. (Mariani et al., 2018) used BAGAN to address mode collapse in GAN during data augmentation. (Shamsolmoali et al., 2020) combined GAN with capsule network (Mazzia et al., 123AD) to avoid mode collapse problem in data augmentation task. It also achieves performance with much less parameters compared to GAN based on convolutional networks.

3. Aim and Objectives

The aim of this research is to propose a GAN architecture to augment images in ISIC dataset. By augmenting the dataset, we will be able to improve the balance of the images between benign and malignant types. This will help to learn a generic deep neural network model for skin cancer image classification.

The research objectives are as follows:

- To explore state-of-art GAN architectures
- To propose a GAN architecture suitable for augmenting ISIC dataset
- To evaluate the classification performance of state-of-art convolution network on ISIC dataset with GAN based data augmentation

4. Significance of the Study

The study is important to augment the images in imbalanced medical image dataset such as ISIC. This will help to learn a generic deep neural network model for improving classification performance of skin cancer images and could be lifesaving.

5. Scope of the Study

The scope will be to explore the GAN models for data augmentation. Modification of the state-of-art classification model is out of scope.

6. Research Methodology

Accurate classification of skin melanoma is an important step towards early detection and treatment of the disease. Figure A-3 shows an example of such ML classification model. It outputs a probability of input image being benign or malignant.

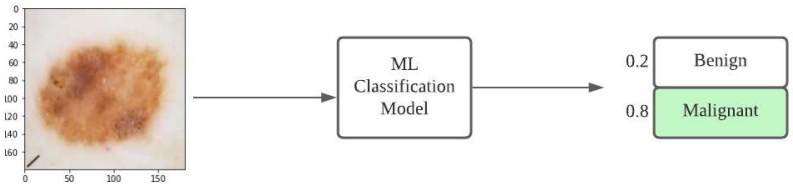

Figure A-3: Automatic classification of lesion images

For improving classification performance on the imbalance ISIC dataset, we will use GAN to generate synthetic images for the imbalanced class and combine it with the rest of the ISIC image dataset. The overall flow of training is shown below in Figure A-4. The aim of this training is to create a very generic ML classification model, which can classify lesion image with higher accuracy.

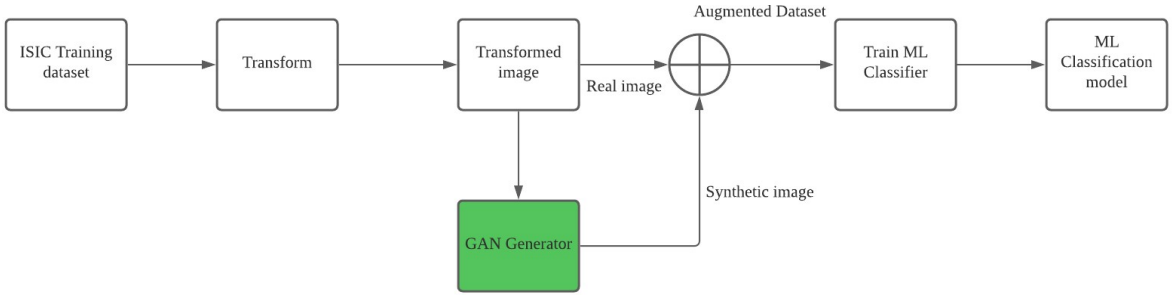

Figure A-4: Training ML model with data augmentation from GAN

A balanced training dataset is a key requirement towards training a generic ML classification model. So, the research proposal focuses on GAN architecture to create synthetic images for classes where less image is available and intends to balance this dataset. The research flow for this study is given below in Figure A-5.

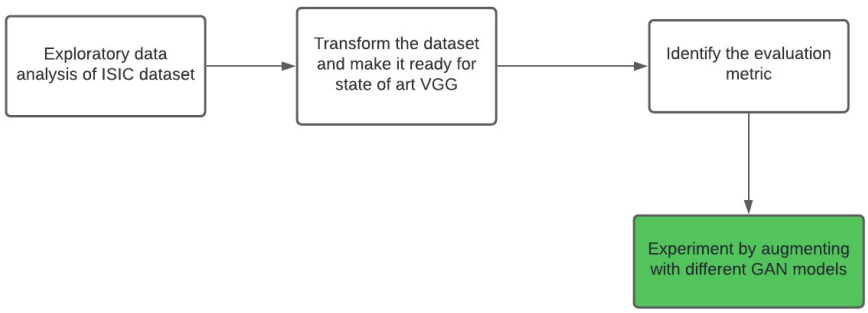

Figure A-5: research flow

We now describe each phase of the research flow

ISIC dataset Exploration

For the classification task, ISIC dataset has 2 ground truth labels,

1. Malignant
2. Benign

The train dataset has 33126 images. Images are of different shapes 4000x6000x3, 2448x3264x3, 1052x1872x3 etc. A sample of ISIC dataset images are shown in Figure A-6.

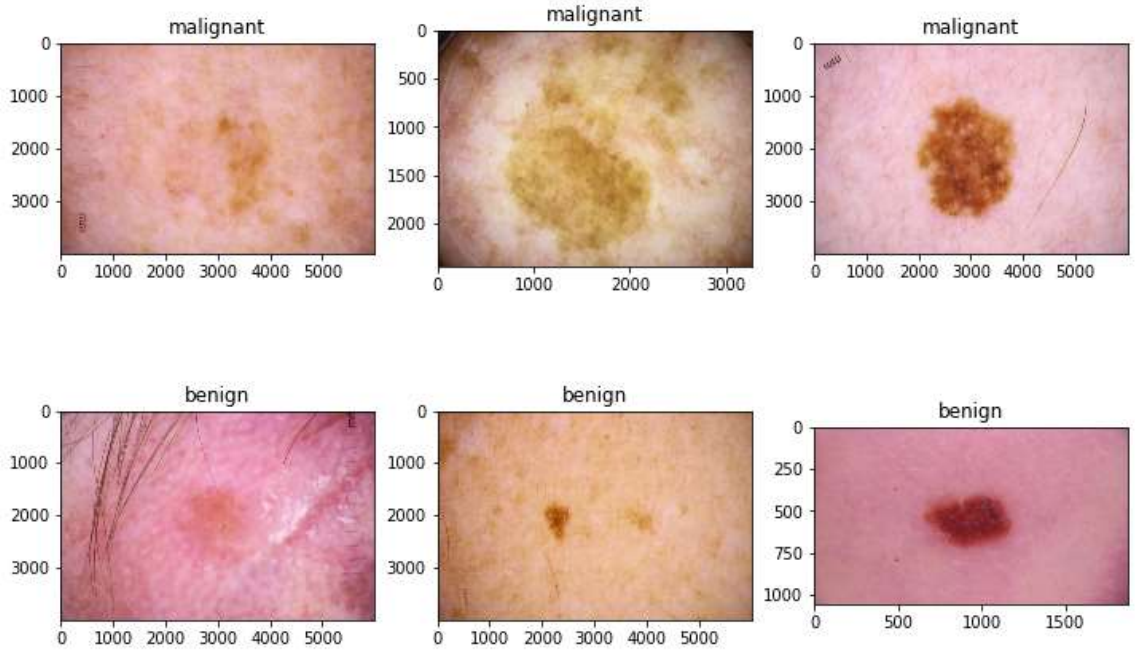

Figure A-6: Sample images from ISIC dataset

We observe large data imbalance in ISIC dataset. The bar chart of Figure shows the number of images in the ISIC dataset among the benign and malignment types

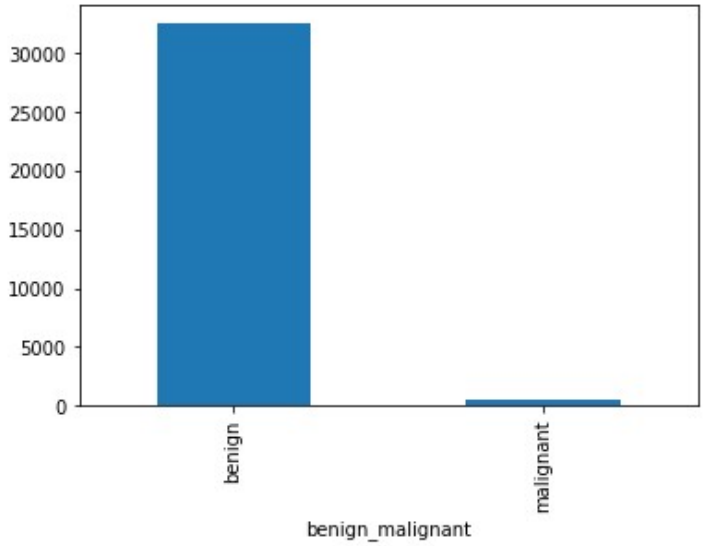

Figure A-7: Distribution of classes in ISIC dataset

The type benign dominates and it is above 98% in the dataset.

Image transformation

We propose to use state-of-art VGG architecture (Simonyan and Zisserman, 2015) for classification of ISIC dataset.

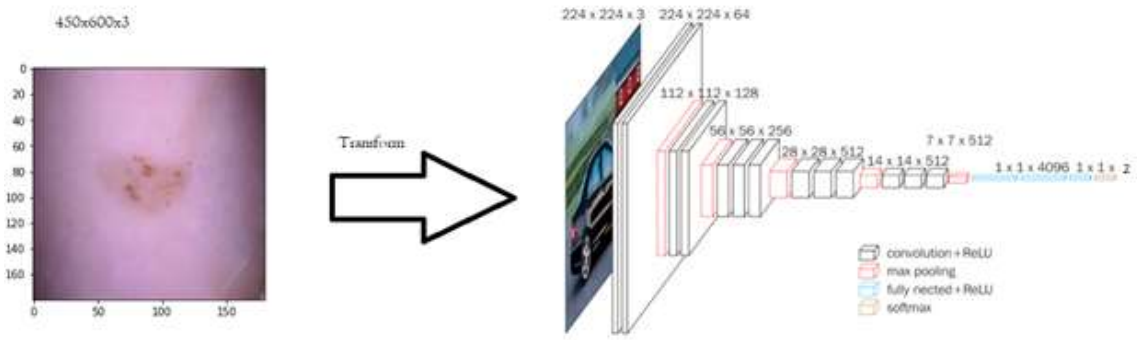

Figure A-8: Image transformation for VGG

The input image to VGG is of shape 224x224x3. The ISIC dataset is of different shapes. We will transform the image when loading the dataset. The output of VGG16 is a classification among 1000 image classes. We will modify the final layer to map to classification of 2 types.

Evaluation Metrics

Classification task performance is evaluated with a confusion matrix (Table A-1) which depends on how many predictions we made correctly and incorrectly,

	Predicted True	Predicted False
Actual truth	True Positive (TP)	False Negative (FN)
Actual false	False Positive (FP)	True Negative (TN)

Table A-1: Confusion Matrix

Based on confusion matrix, following evaluation criteria has been defined,

Sensitivity / Recall	TP / (TP + FN)
Specificity	TN / (FP + TN)
Precision	TP / (TP + FP)
F1	2 x (Precision * Recall) / (Precision + Recall)
Accuracy	(TP + TN) / (TP + FN + FP + TN)

Table A-2: Classification evaluation

We will evaluate these parameters, i.e., sensitivity, specificity, precision, F1 and accuracy in our classification model. We will prioritize sensitivity score to reduce false negative case.

GAN architecture exploration

We will explore following state-of-art GAN models,

1. DCGAN: DCGAN (Radford et al., n.d.) was the initial GAN architecture which was able to produce high resolution images. It achieves this with following changes in the deep convolution model,
    a. On generator side it includes fractional strided convolutions, on discriminator side it includes strided convolutions instead of pooling layers.
    b. It uses batchnorm in both generator and discriminator.
    c. Removed fully connected layer in discriminator
    d. On generator side, it used Relu activation on all layers except output where it used tanh. On discriminator it used leaky Relu in all layers

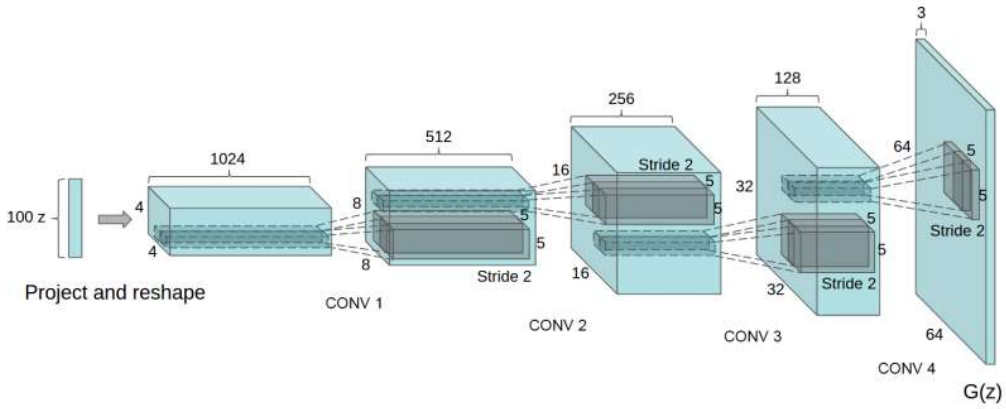

Figure A-9: DCGAN generator

2. LAPGAN: LAPGAN is another model to generate high resolution images. It uses conditional GAN generating images at different resolutions, (shown as G0, G1, G2, G3 etc. in Figure A-10). The generator starts with a noise vector z3 as shown Figure and it is passed through generator G3 to generate an image I3. It is then passed to generator G2, which has another noise vector z2 as input. Together with these two input it generates another image I2. This continues as the resolution of the image keeps increasing.

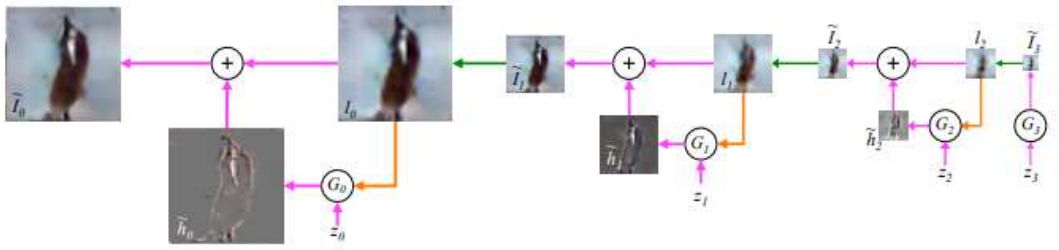

Figure A-10: Image generated in LAPGAN from right to left

3. PGGAN: In PGGAN, layers of neural network generate an image progressively from low resolution images to high resolution images. The initial layers of generator learn large scale features of the image and the later layers of the generator learns finer detail. Discriminator network layers mirrors the generator network.

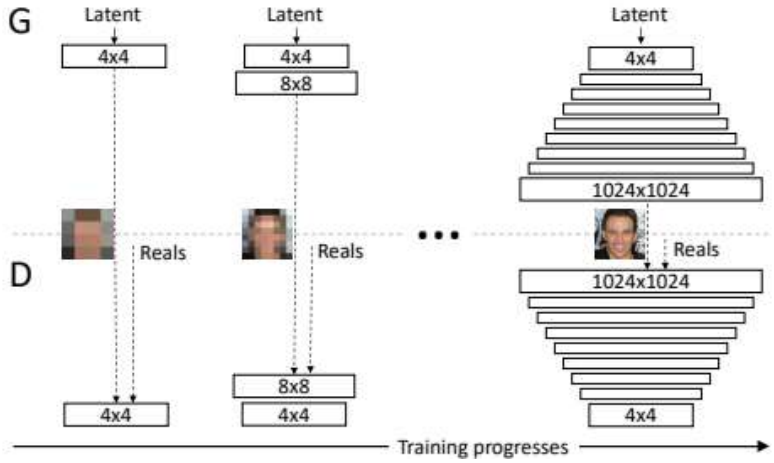

Figure A-11: PGGAN training process

We will use these GAN models to augment the ISIC dataset and evaluate the classification performance. We will summarize the observation from different GAN on classification performance and compare with baseline VGG performance.

7. Requirements Resources

We categorize the requirements as follows:

a) Hardware resource: We will require GPU. We will utilize GPU framework from google colaboratory

b) Software resource: We will require machine learning framework. We will use pytorch.

8. Research Plan

The GANT chart for the project plan is shown in Figure A-12.

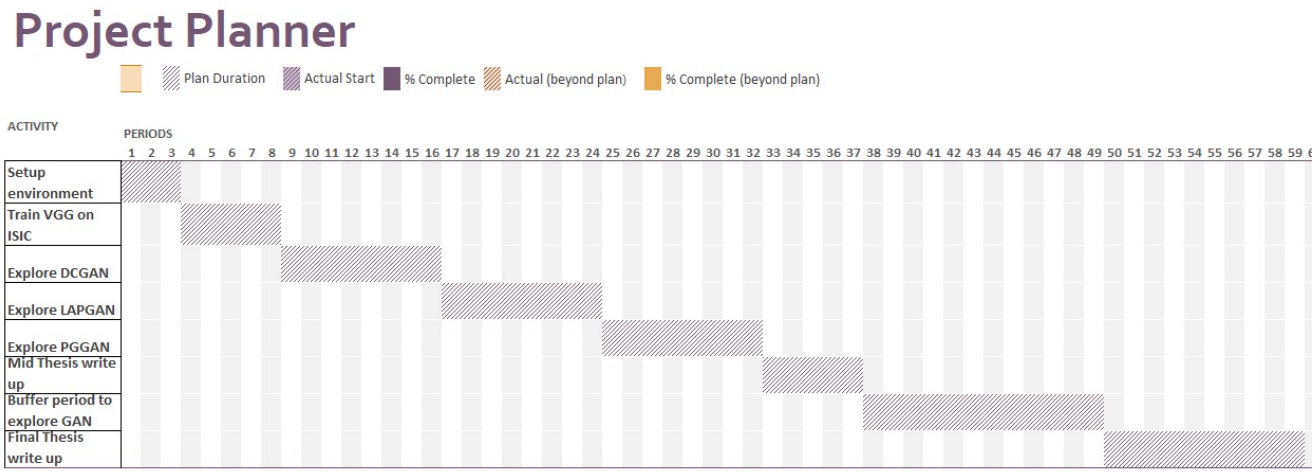

Figure A-12: GANT chart