Assignment 4
Deep Learning

Submitted By:

Gyanendra Chaubey
M.Tech (AR-VR)
Roll No: M23AIR005

Submitted To:

Dr. Deepak Mishra
Assistant Professor
Dept. of CSE

1. Methodology:

In this work, generation of the dermoscopic lesion images has been done using the conditional Generative Adversarial Networks (GANs). Then we need to generate the images on the dataset and train a classifier on the train dataset. Then check accuracy of the classifier on the generated images and actual test images. We have tried several different methods and architectures for the GANs. Three important ones are given below.

1.1 Architecture 1

In this architecture, we have first tried to implement the random sketches with the labels. In this architecture, first a random image is selected from the given dataset, and it is then concatenated and fed to the Generator to the get the generated image. After that, the real image is concatenated with the labels and generated image get concatenated with the labels and both the images fed to the Discriminator model to classify them as real or fake.

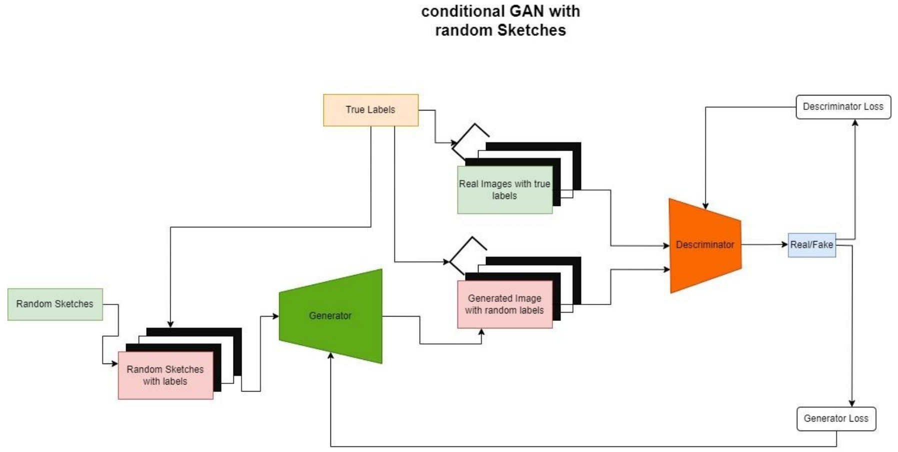


Fig. 1 Conditional GAN with random sketches and labels

1.2 Architecture 2

In the second architecture, we are passing the paired sketches with the labels to the Generator to generate the images. After the images are generated, it is again get concatenated with the labels, also real images are concatenated with the labels and fed into the discriminator classify them as real or fake.

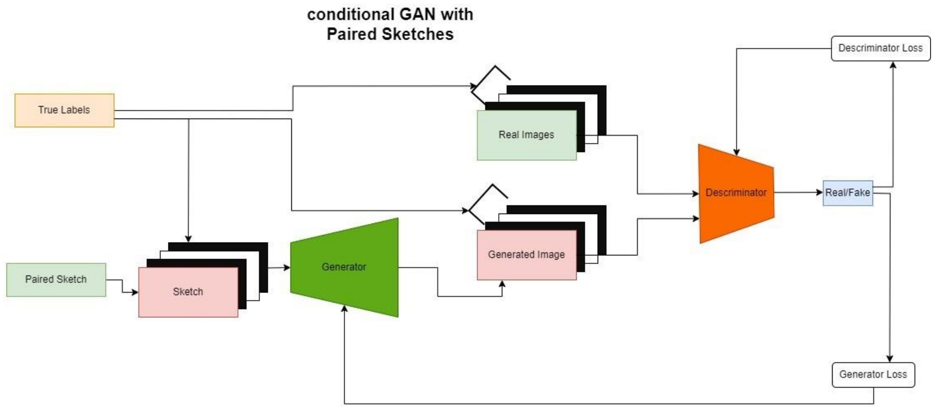


Fig. 2 conditional GAN with Paired Sketches and labels

1.3 Architecture 3

In the third architecture, we have implemented pixel to pixel image translation. In this we have fed the sketches as the conditions to the generator and generated the image. Generated image and Real image afterwards fed to the discriminator to determine the real or fake.

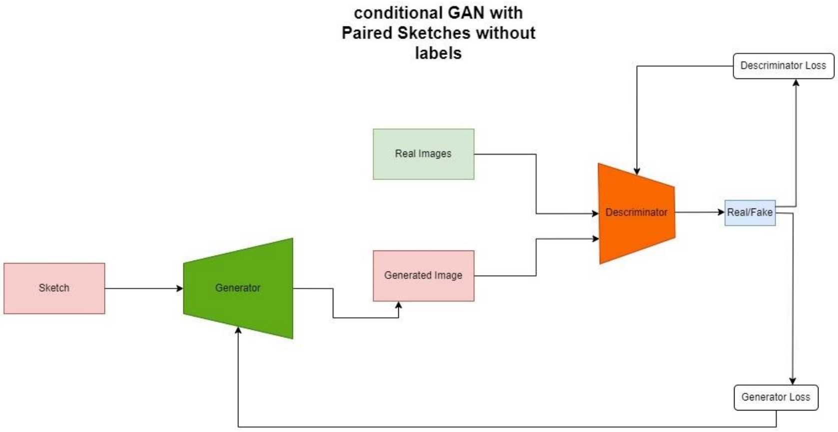


Fig. 3. Conditional GAN with paired sketches without labels

1.4 Loss and Score functions for cGAN

In this work, **Binary Cross Entropy along with L1 loss regularizer** has been used for the training of the model. The output of the generator is the generated image. Since, here only two classes need to be compared one is real image and second one is generated image that's why chosen binary cross entropy. L1 loss is also used as the regularizer for the model. Also, for the evaluation of the model Frechet Inception Distance (FID) and Inspection Score (IS) has been used. The equation for the Binary Cross Entropy, L1 loss, FID and IS is given as Eq.1, 2, 3, and 4 respectively.

$$H(y,\hat{y}) = -\frac{1}{N}\sum_{i=1}^{N}(y_i * \log(\hat{y})) + (1-y_i) * (1-\log(\hat{y})) \tag{1}$$

$$L1 = \frac{\sum_{i=1}^{n}|y_i - x_i|}{n} \tag{2}$$

$$D_F(\mathcal{N}(\mu,\Sigma),\mathcal{N}(\mu',\Sigma')) = ||\mu-\mu'||^2 + tr(\Sigma+\Sigma'-2(\Sigma\Sigma')^2 \tag{3}$$

$$IS(p_{gen},p_{dis}) = \mathbb{E}_{x\sim p_{gen}}\left[D_{KL}(p_{dis}(.\,|x)||\mathbb{E}_{x\sim p_{gen}}[p_{dis}(.\,|x)])\right] \tag{4}$$

1.5 Classifier

A classifier has been built to classify the real images and generated images.

2. Results

The results shown in the below approaches are the best obtained results after several iterations of modifications either in the architecture or in the hyper parameters of the model. For each of the model, G_loss and D_loss is given.

2.1 Architecture 1

The architecture 1 has been trained for the 50 epochs, the D_loss and G_loss of the model is given as below in Fig. 4 and Fig. 5 respectively. We can see that, the D_loss is around 0.5 in each of the epochs iterating over 50 epochs.

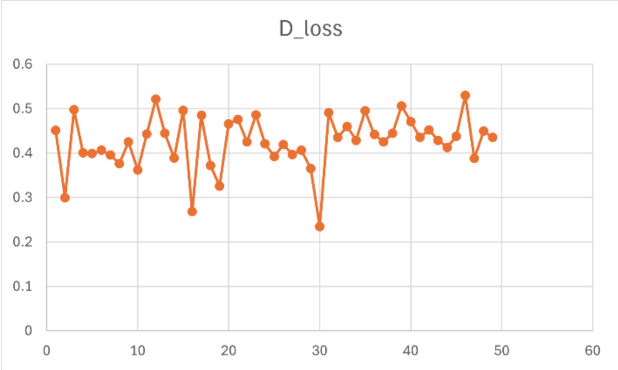


Fig. 4 D_loss of the architecture 1

The G_loss has been observed conserving the Fig. 5 on increasing the training epochs of the model.

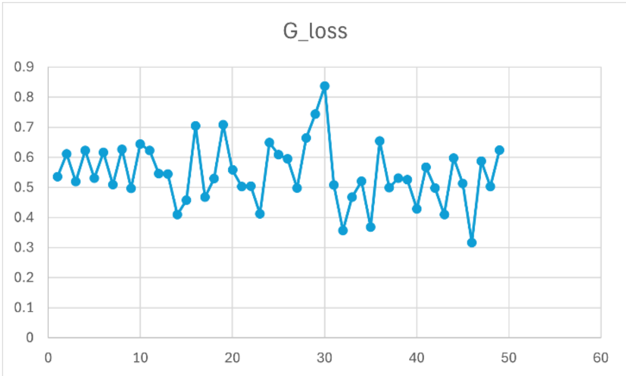
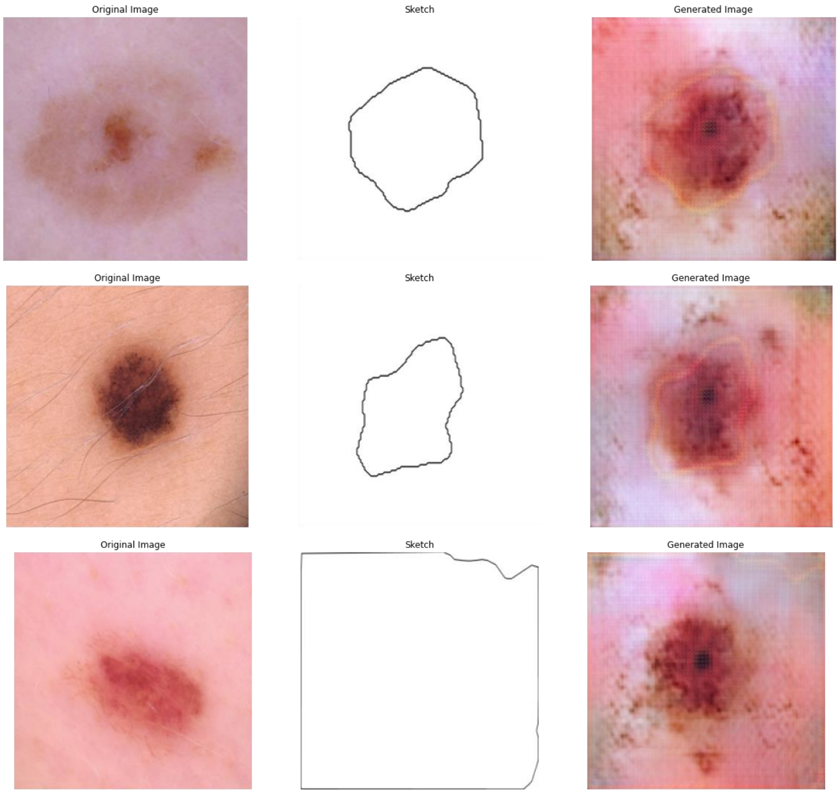


Fig. 5 G_loss of the architecture 1

Some samples of the generated images from the architecture1 are shown in Fig. 6.



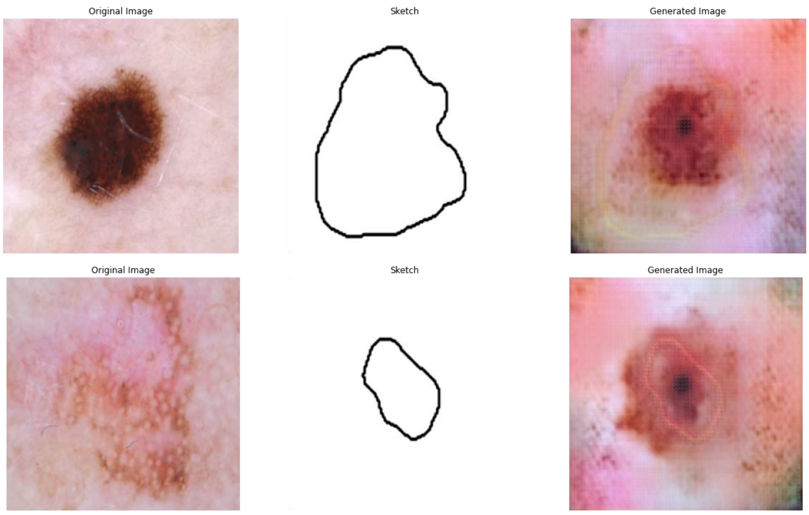


Fig. 6 Some samples of generated images from Architecture 1

2.2 Architecture 2

This architecture has been trained for the 100 epochs and each of the epochs are trained for the 626 iterations based on the number of images. The discriminator loss can be seen as the variating in between the 0 and 1. Discriminator loss was got optimised during its last iterations about 0.5 as can be seen in the Fig. 7.

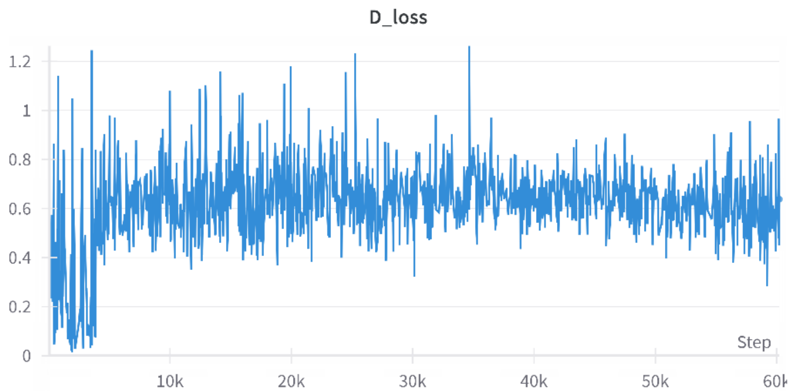


Fig. 7 Discriminator loss for the Architecture 2

In the Fig. 8, generator loss is continuously being reduced and goes about 3 during its last iterations. As it is only run for 100 epochs due to limitation of the time, if it is being trained for iterations it can give the better reduced loss.

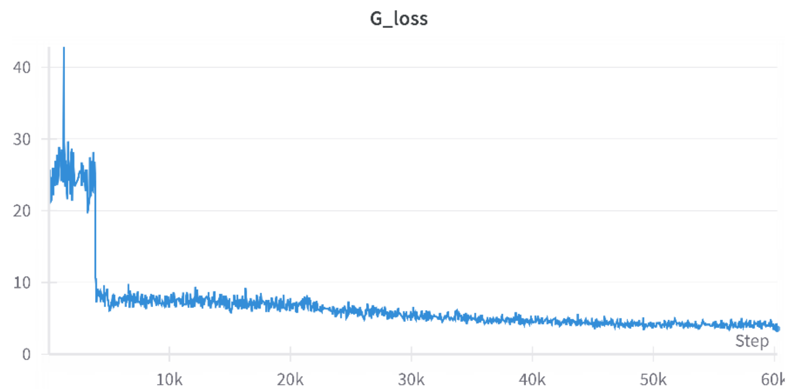


Fig. 8 Generator loss for the Architecture 2

Some of the outputs of the architecture are shown in Fig 9.

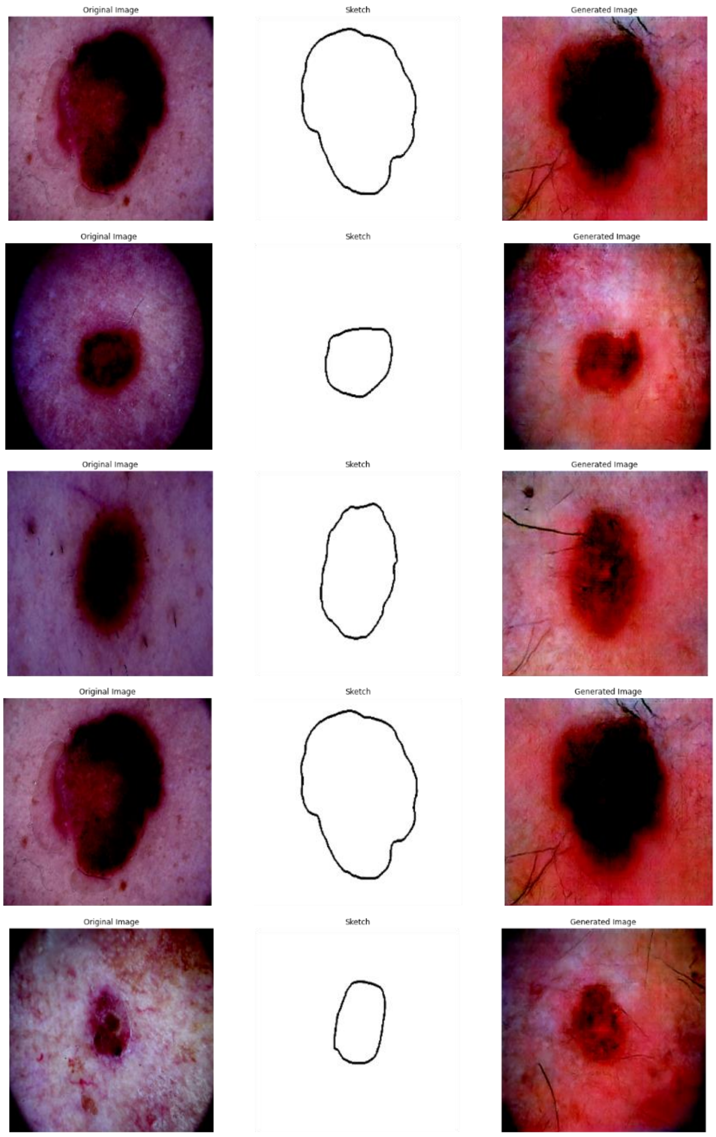


Fig. 9 Samples of the generated images

2.3 Architecture 3

This architecture has been trained for the 1200 epochs and each of the epochs are trained for the 626 iterations based on the number of images. The discriminator loss can be seen as the variating in between the 0 and 1. Discriminator loss was got optimised during its last iterations as can be seen in the Fig. 10.

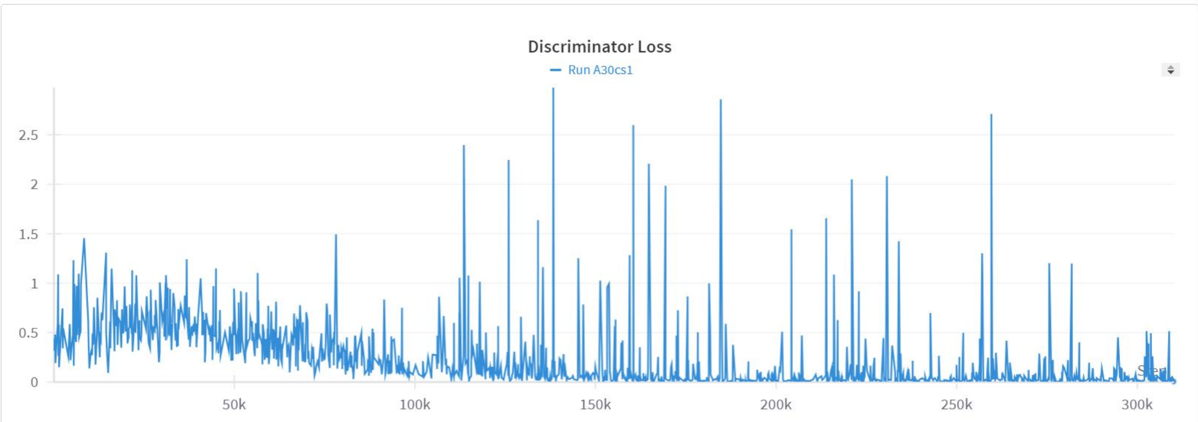


Fig. 10 Discriminator loss for the Architecture 3

In the Fig. 11, generator loss is continuously being reduced and goes about 12 during its last iterations. As it is only run for 1200 epochs optimization and results of the model are obtained better.

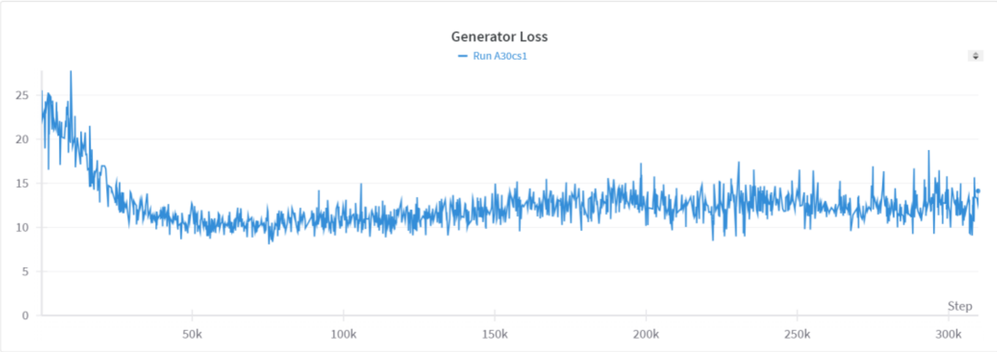
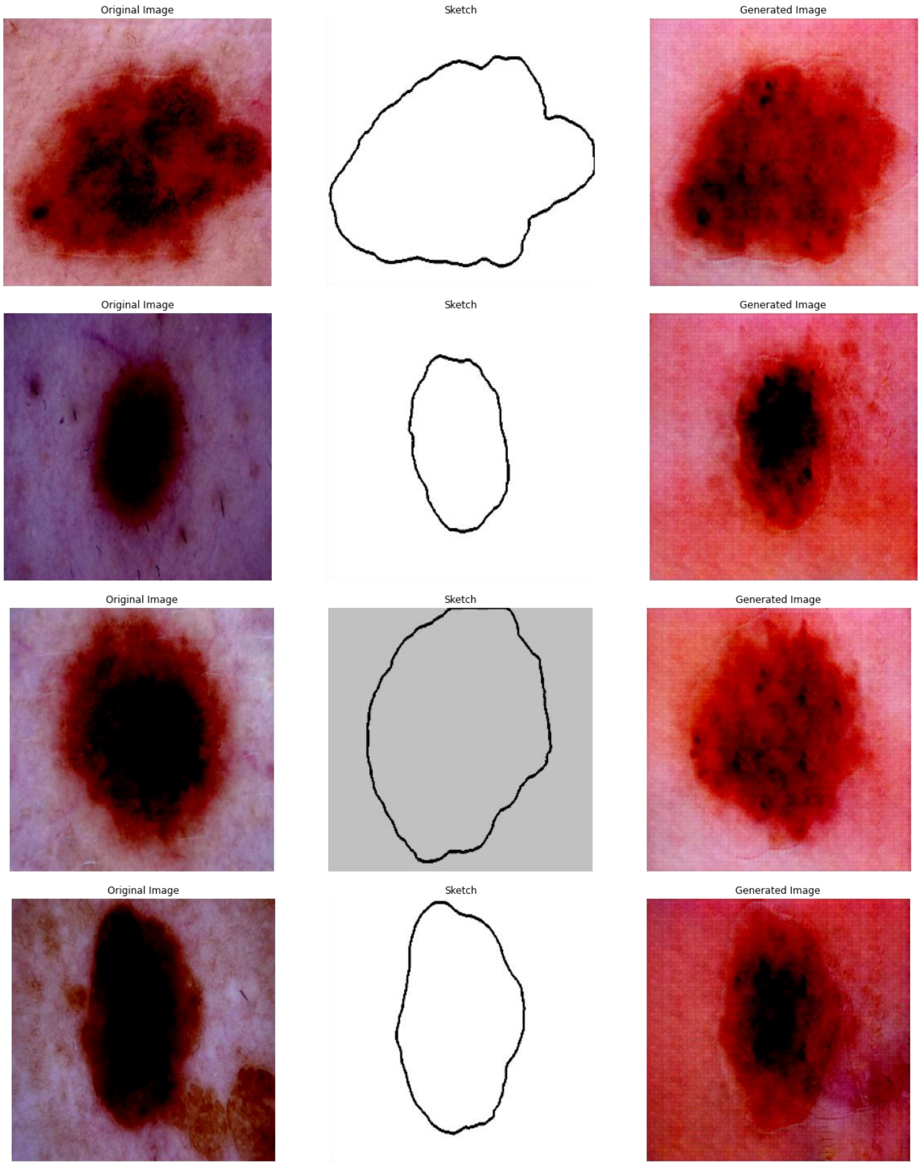


Fig. 11 Generator loss for the Architecture 3

Some samples generated from the Architecture 3 are shown in the Fig 12.



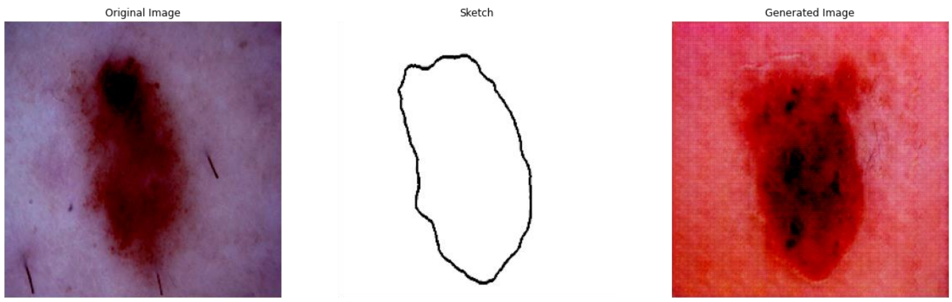


Fig. 12 Samples generated from the architecture 3

The FID and Inceptions score for each of the architectures are given in the Table 1.

	FID	IS
Architecture 1	32.67	5.27
Architecture 2	0.22	16.35
Architecture 3	0.34	11.23

2.4 Classification Results

The classification model has been trained from scratch have the training accuracy 80% on the data on running up to 15 epochs while the testing accuracy observed is 60%. On the generated images the accuracy has been achieved to be 56%.

3. Discussion and Conclusions

From this experiment, we have learned how to train a conditional GAN. I have trained three architectures and came to know that their different possible combinations from which a GAN can be trained, and the results also varies on changing the input method. As it can be observed from the results of the three architectures. Also, the classification model is trained on the given data, and we have obtained a 60% result on the testing the data.