**Realistic Face Image Generation from Sketch**



**Arkabrato Chakraborty**

**19CS4109**

Under the guidance of **Associate Prof. Dakshina Ranjan Kisku**

National Institute of Technology, Durgapur

*A Thesis submitted in Partial fulfilment for the Degree of*

**Master of Technology**

**in**

**Computer Science and Engineering**

May 2021

**Abstract**

Generating a face image from a sketch image given is a topic on which a lot of recent work is going on but still there has been lot of accuracy and perceptual issues with the generated image. It has a lot of applications in various fields especially criminal investigation and identification along with digital entertainment purposes. We have aimed at generating a face image which has higher accuracy perceptually. We have tried approach of using Conditional GAN’s and making it an image translational task and have achieved considerable accuracy.

**Acknowledgement**

I would like to express my sincere gratitude to my thesis advisor Prof Dakshina Ranjan Kisku for being a wonderful mentor. I am very thankful for his constant encouragement, guidance and unlimited patience in explaining things to me. Without his help and support, it would have been impossible for me to complete the thesis. I would like to thank Chinese University of Hong Kong for publishing a database needed for my project. I would like to thank all the faculty and staff of the Computer Science & Engineering department for providing an environment which promotes learning and all my batch mates for creating a good environment. I would like to thank my family specially my mother for their moral support.

**List of Figures**

1.1 Sketch image and its corresponding ground truth real face image…......2

1.2 A police sketch of a criminal suspect…………………………………….....3

3.1 Structure of a typical CNN with all the layers………………………………9

3.2 An example of a convolution operation in CNN in convolutional layer….10

3.3 A model of simple vanilla GAN………………………………………………11

3.4 Block Diagram of Conditional GAN where Y is the condition…………….12

3.5 Block diagram showing working of generator of cGAN……………………13

3.6 Block diagram showing working of discriminator of cGAN………………..14

* 1. Generator Architecture- consisting of encoder decode network………….16
  2. Discriminator Architecture…………………………………………………….17

5.1 Input Sketch, Generated Face image and Ground truth image…………..18

**List of Tables**

5.1 FID scores obtained on various sized data result……………………………….20

5.2 Graphical representation of the experiment with x-axis representing the

%of the dataset and y-axis representing the FID score achieved………………...21

5.3 FSIM scores obtained on various sized data of result………………………….22

5.4 Graphical representation of the experiment with x-axis representing the

%of the dataset and y-axis representing the FSIM score achieved………………22

5.5. Comparison values of experiments with other methods……………………….23

**Table of contents**

**Abstract**

**List of Figures**

**List of Tables**

**List of Abbreviations**

Chapter

1. **INTRODUCTION 1**
   1. Problem Definition and Objective
   2. Motivation
   3. Solution Approach
   4. Technologies used
2. **Literature Review 5**
3. **Preliminaries………………………………………………………………………………….9**
   1. Convolutional Neural Networks
   2. Generative Adversarial Networks…..
      1. Conditional Generative Adversarial Networks([c-GAN)](#_TOC_250006)

1. **Project Implementation……………………………………………………………….15**

3.1. Dataset

3.2. Experimental Configurations

3.3. Network Architectures

1. **Results 18**
   1. [Value loss or Accuracy](#_TOC_250002)
   2. [Experiments](#_TOC_250001)
2. **Conclusion & Future Work 25**
3. **Bibliography………………………………………………………………………………………………………………26**

**Chapter 1**

**Introduction**

Computer vision is and [interdisciplinary scientific field](https://en.wikipedia.org/wiki/Interdisciplinarity) that deals with how [computers](https://en.wikipedia.org/wiki/Computer) can gain high-level understanding from [digital images](https://en.wikipedia.org/wiki/Digital_image) or [videos](https://en.wikipedia.org/wiki/Video). From the perspective of [engineering](https://en.wikipedia.org/wiki/Engineering), it seeks to understand and automate tasks that the [human visual system](https://en.wikipedia.org/wiki/Human_visual_system) can do.

Computer vision tasks include methods for [acquiring](https://en.wikipedia.org/wiki/Image_sensor), [processing](https://en.wikipedia.org/wiki/Image_processing), [analysing](https://en.wikipedia.org/wiki/Image_analysis) and understanding digital images, and extraction of [high-dimensional](https://en.wikipedia.org/wiki/High-dimensional) data from the real world in order to produce numerical or symbolic information, e.g. in the forms of decisions Understanding in this context means the transformation of visual images (the input of the retina) into descriptions of the world that make sense to thought processes and can elicit appropriate action. This image understanding can be seen as the disentangling of symbolic information from image data using models constructed with the aid of geometry, physics, statistics, and learning theory.

The [scientific discipline](https://en.wikipedia.org/wiki/Scientific_discipline) of computer vision is concerned with the theory behind artificial systems that extract information from images. The image data can take many forms, such as video sequences, views from multiple cameras, multi-dimensional data from a 3D scanner, or medical scanning device. The technological discipline of computer vision seeks to apply its theories and models to the construction of computer vision systems.

These project deal with a particular branch of computer vision that entails generating a human face image from a given sketch image of a human face. This falls under the category of image-to-image translation where we are trying to find some kind of mapping between two images.

This field of work has application in various fields including criminal identification, digital entertainment purposes. There has been various models and methods to generate image from a sketch with varied degree of success. We have approached deep learning models to target our problem with a aim to better perceptual quality of the generated image.

* 1. **Problem Definition and Objective**

Generation of a realistic face image of a human being from a given sketch drawn of a human face.

Our aim is to generate corresponding human face image with significant accuracy and which is perceptually close to ground truth face image from the sketch image.



Fig 1.1 Sketch image and its corresponding ground truth real face image

As shown in above figure are the sketch image and the real ground truth face image of the human being. Our aim is to generate an image which is as close as possible to the given ground truth image and is perceptually accurate .

* 1. **Motivation**

In various fields generating a face image from a sketch image can be of immense help. One such is in field of criminal identification. Many situations arise where it becomes difficult to capture photo of a criminal suspect due to insufficient CCTV cameras in a location and other time related constraints. In



Fig1.2 A police sketch of a criminal suspect

such situations we have to depend on memory of witnesses and his description. From such a description a sketch is drawn by an expert and on its basis, investigation is performed. If we can generate a realistic perceptually accurate face image of the person from the sketch it can help in criminal identification and tracking and make the overall investigation more accurate and efficient. Also, it has application where robots draw human faces and for educational and digital entertainment purposes.

* 1. **Solution Approach**

The problem falls in the category of image-to-image translation where recently various Deep Learning Models are being applied to varied success. We decided to apply Generative Adversarial Model (GAN) as it working is similar to our problem. GAN generates a data distribution similar to an existing data distribution from a random distribution. We have applied GAN in the conditional setting which helps in controlling the input and in our case, condition is a particular ground truth face image. Our aim is to generate a mapping between the input sketch image and the output face image and training it to reduce losses and increase accuracy.

* 1. **Tools and Technologies used**

The following tools and technologies were used for implementation purposes.

-NumPy

-Pandas

-OpenCV

-Pytorch

-Google Colab

-Python

**Chapter 2**

**Literature Review**

There has been a lot of work done and also going on in this problem. Earlier works include traditional non-Deep Learning based models as well as recent works based on Deep learning. We have discussed here some of the relevant methods and work done.

Existing methods can be classified into three main categories on the basis of: subspace learning, sparse representation, and Bayesian inference. Subspace learning mainly refers to linear subspace-based methods (e.g., principal component analysis, or PCA), and nonlinear subspace methods such as manifold learning-based methods (e.g., local linear embedding). Tang and Wang [1], proposed a linear face sketch synthesis method based on PCA called eigen sketch transformation. They assumed that a source input photo and the target output sketch shared the same projection coefficients obtained from the PCA procedures. The coefficients are first obtained by projecting the input photo onto the training photos. The target sketch is then synthesized from a linear combination of training sketches weighted by the obtained projection coefficients. Tang and Wang then proposed an improved method by separating the shape from the texture. The eigen sketch transformation method is applied to shape and texture to compute the target shape and texture, respectively. The synthesized shape and texture are fused to obtain the final target sketch.

Li et al. [2] proposed a hybrid subspace method for face photo synthesis by concatenating the training sketches and photos. Since the mapping between the whole sketches and corresponding photos is not linear, the above methods hardly synthesize a realistic sketch, especially when the hair region was included.

Liu et al improved these linear models by exploiting LLE. Images are divided into overlapping patches. For each test photo patch, K photo patches are selected from the training set according to the Euclidean distance metric. Simultaneously, the corresponding K sketch patches are taken as the

candidates. The weights for combining the K candidates are calculated by minimizing the least square of the reconstruction residual between the test photo patch and its K nearest neighbours. The target sketch patch is generated from a linear combination of those K candidates. Finally, the target sketch is fused from these patches with an averaged overlapping area, which incurs a degree of blurring.

The same LLE procedure was applied to generate an initial estimate in Liu et al.’s work . They explored the idea of face hallucination to further compensate the residual, which might be lost in the initial estimation procedure. Sparse representation has wide applications in compressed sensing, image processing , computer vision, and pattern recognition. It can decompose a signal into a linear combination of atoms (or bases) weighted by a sparse vector. Chang et al. introduced sparse coding to face sketch synthesis. They first learned two coupled dictionaries (sketch patch dictionary and photo patch dictionary) via sparse coding. The coupled dictionaries denotes that they are learned from joint training by concatenating these two sub dictionaries into one dictionary, so that the input training photo patch and the corresponding sketch patch have the same sparse representation coefficients. A test photo patch is then decomposed on the photo patch dictionary, weighted by the sparse representation coefficients. Hence from the linear combination of the atoms of the sketch patch dictionary, the target sketch patch is produced. Considering that the weighted combination of the candidates may result in the loss of high-frequency information, Wang et al. proposed a two-step framework [16], [29] to further enhance the definition of the target output. In the first stage, a sparse feature selection algorithm is explored to synthesize an initial image which found closely related neighbors adaptively through sparse representation. In the second stage, the sparse coding strategy is explored to learn the mapping between the high frequency information of sketch patches and photo patches

Bayesian inference-based methods include embedded hidden Markov model (E-HMM)-based and Markov random fields (MRF)-based methods. Gao et al. [3] modelled the mapping between sketches and their photo counterparts by E-HMM. A face is decomposed into five super-states (corresponding to forehead, eye, nose, mouth, and chin) and substates (states in each super-state). Assuming the sketch and the photo have the same transition probability matrix, a joint training strategy is adopted. They then improved the proposed model in a local patch-based form which synthesized more detail. All the above methods (subspace learning-based, sparse representation-based and E-HMM based) independently synthesize a target image patch and ignore the neighboring relation between overlapping patches. Taking the neighboring relation between overlapping patches as a regularization term, a number of MRF-based methods were proposed.

All the aforementioned methods conduct face sketch synthesis or photo synthesis from the inductive learning perspective. Although they have obtained promising synthesis performance, these methods may result in high losses for a particular set of test samples. This is because inductive learning minimizes the empirical loss for training samples.

In contrast, transductive learning algorithms minimize the expected loss for test samples by incorporating the given test samples into the learning process. Therefore, a transductive learning oriented method may significantly reduce the high expected loss and improve the synthesis performance for the given test samples. This paper presents a novel transductive face sketch-photo synthesis method. We mainly focus on face images in a frontal pose with normal lighting and neutral expression. All sketches and photos in this paper are divided into even patches with some overlap between neighboring patches. We design a probabilistic graphic model to model the relationship between sketch-photo patch pairs. This model takes both the reconstruction fidelity of the input photo (sketch) and the synthesis fidelity of the target output sketch (photo) into account. Furthermore, the relation between neighboring sketch patches is considered as a prior regularization on the hidden parameters. An alternative optimizing scheme is adopted to solve the proposed probabilistic model, which converges in a small number of iterations. Finally, a min cut algorithm is adopted to find the minimum error boundary to stitch the overlapping areas. The proposed method has the capability to handle both sketch synthesis and photo synthesis, because these two procedures are symmetric.

Jian Zhao et al in their paper, treat the sketch to face problem as a face hallucination reconstruction problem. In order to solve this problem, they propose an image translation network by exploiting attributes with generated adversarial network (GAN). And it can significantly contribute to the authenticity of the generated face by supplementing sketch image with the additional facial attribute feature. The generator network is composed of a feature extracting network and a downsampling-upsampling network, both networks use skip-connection to reduce the number of layers without affecting network performance. The discriminator network is designed to examine whether the generated faces contain the desired attributes or not. In the underlying feature extraction phase, our network is different from most attribute-embedded networks, we fuse the sketch images and attributes perceptually. They set the network sub Branch A and B, which receive sketch image and attribute vector in order to extract low-level profile information and highlevel semantic features

**Chapter 3**

**Prerequisites**

**3.1 Convolutional Neural Network**.

A **Convolutional Neural Network (CNN)** [5]is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The pre-processing required in a CNN is much lower as compared to other classification algorithms. While in primitive methods filters are hand-engineered, with enough training, CNN have the ability to learn these filters/characteristics.

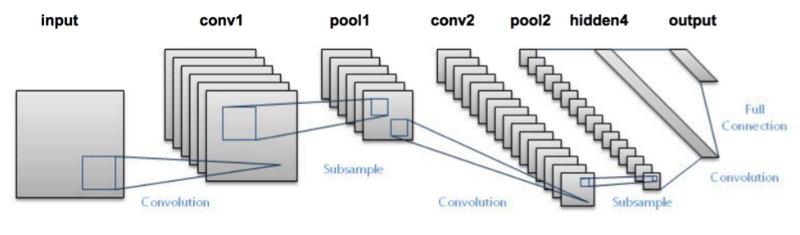


Fig3.1 Structure of a typical CNN with all the layers.

The architecture of a CNN is analogous to that of the connectivity pattern of Neurons in the Human Brain and was inspired by the organization of the Visual Cortex. Individual neurons respond to stimuli only in a restricted region of the visual field known as the Receptive Field. A collection of such fields overlaps to cover the entire visual area.

A CNN is able to **successfully capture the Spatial and Temporal dependencies** in an image through the application of relevant filters. The architecture performs a better fitting to the image dataset due to the reduction in the number of parameters involved and reusability of weights. In other words, the network can be trained to understand the sophistication of the image better.

The role of the CNN is to reduce the images into a form which is easier to process, without losing features which are critical for getting a good prediction. This is important when we are to design an architecture which is not only good at learning features but also is scalable to massive datasets.

A CNN typically consists of an input and an output layer, as well as multiple hidden layers. The hidden layers of a CNN typically consist of convolutional layers, non-linear layer i.e., activation function (e.g: ReLU), pooling layers, fully connected layers and normalization layers.

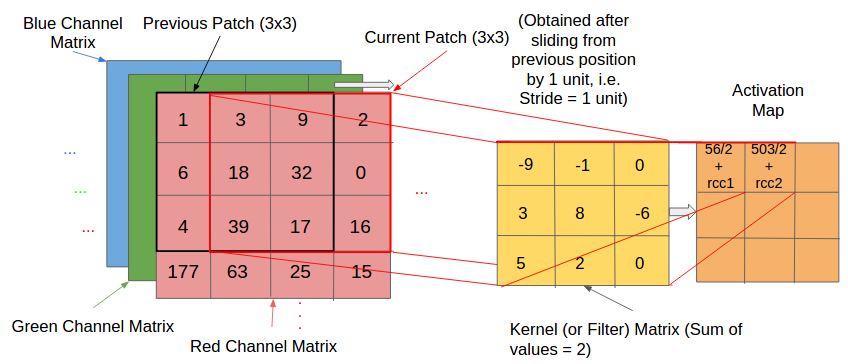


Fig. 3.2 An example of a convolution operation in CNN in convolutional layer

Convolution operation occurs with a filter slides over the input image pixel and does the convolve operation and stores the most important features of the input information. It is further pruned using a max pooling layer. Information then passes through an activation layer to add non linearity. This same process can be repeated again or not depending on the architecture. Then a fully connected layer and at end the output layer. Training is done on whole model to reduce loss by backpropagation and its many variants like Adam optimiser.

**Generative Adversarial Network (GAN)**

GAN’s[6][7] are deep neural networks which consists of two sub networks competing against each other in zero sum fashion to produce a data distribution. consists of two neural networks fighting with each other to generate a probability distribution. The neural networks can be DNN, CNN. The two networks are called generator and discriminator. By mimicking an input probability distribution from a given random distribution, the generator generates a distribution. Discriminator now checks how close are the two distributions by checking how close the generated distribution is close to the actual distribution. Training is done then to both generator and discriminator in zero-sum manner to reduce the error between actual distribution and generated distribution

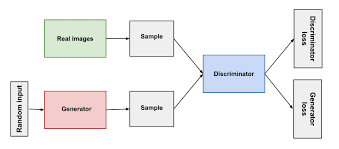


Fig 3.3. A model of simple vanilla GAN

GANs are an exciting and rapidly changing field, delivering on the promise of generative models in their ability to generate realistic examples across a range of problem domains, most notably in image-to-image translation tasks such as translating photos of summer to winter or day to night, and in generating photorealistic photos of objects, scenes, and people that even humans cannot tell are fake.

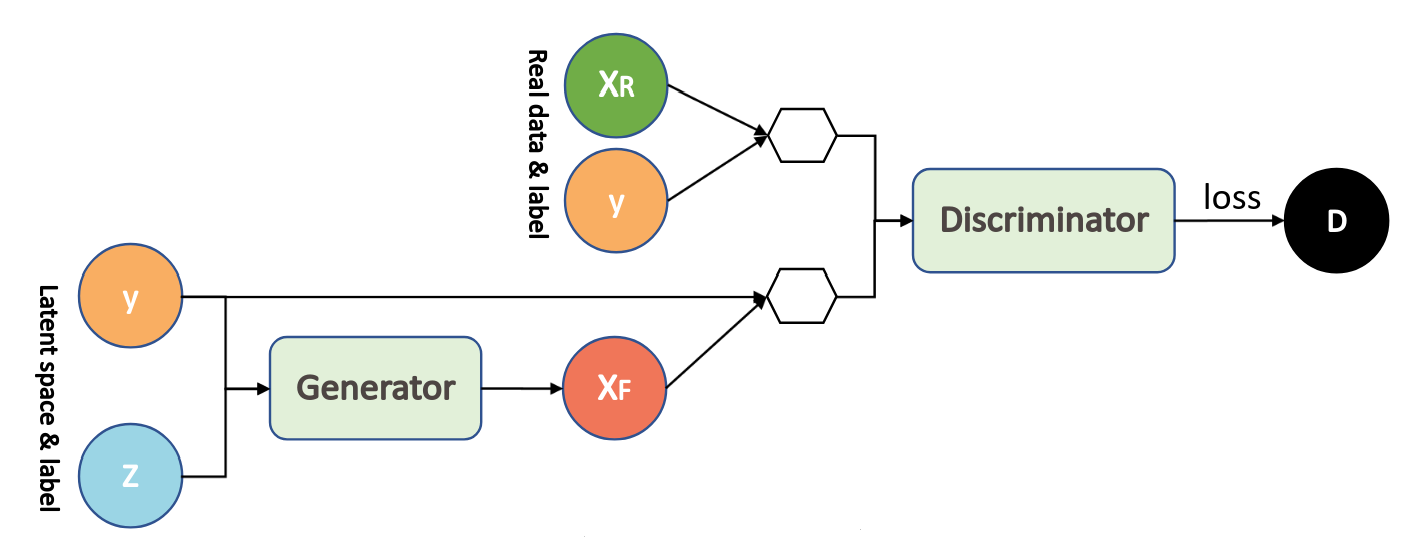
**Conditional Generative Adversarial Networks**

GAN’s have various number of variations and among them Conditional Generative Adversarial Network(cGAN) [8]is one.

GANs are capable of producing examples ranging from simple handwritten digits to photorealistic images of human faces. However, although we could control the domain of examples our GAN learned to emulate by our selection of the training dataset, we could not specify any of the characteristics of the data samples the GAN would generate. For instance, the DCGAN could synthesize realistic-looking handwritten digits, but we could not control whether it would produce, say, the number 7 rather than the number 9 at any given time.

On simple datasets like the MNIST, in which examples belong to only one of 10 classes, this concern may seem trivial. If, for instance, our goal is to produce the number 9, we can just keep generating examples until we get the number we want. On more complex data-generation tasks, however, the domain of possible answers gets too large for such a brute-force solution to be practical.

The ability to decide what kind of data will be generated opens the door to a vast array of applications.

 Fig. 3.4Block Diagram of Conditional GAN where Y is the condition

Architecture of cGAN:

8.2.1. CGAN Generator

The Generator uses the noise vector z and the label y to synthesize a fake example G (z, y) = x\*|y (read as “x\* given that, or conditioned on, y”). The goal of this fake example is to look (in the eyes of the Discriminator) as close as possible to a real example for the given label.

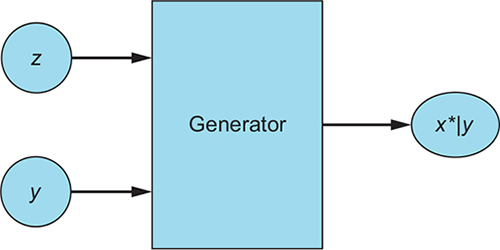


Fig 3.5 Block diagram showing working of generator of cGAN

8.2.2. CGAN Discriminator

The Discriminator receives real examples with labels (x, y), and fake examples with the label used to generate them, (x\*|y, y). On the real example-label pairs, the Discriminator learns how to recognize real data and how to recognize matching pairs. On the Generator-produced examples, it learns to recognize fake image-label pairs, thereby learning to tell them apart from the real ones.

The Discriminator outputs a single probability indicating its conviction that the input is a real, matching pair. The Discriminator’s goal is to learn to reject all fake examples and all examples that fail to match their label, while accepting all real example-label pairs.

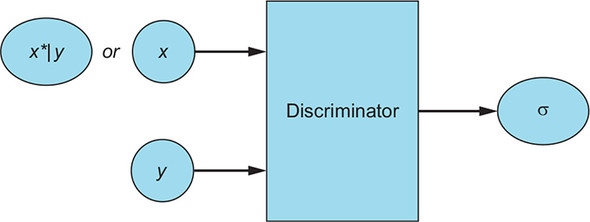


Fig3.6 Block diagram showing working of discriminator of cGAN

GAN’s are trained similar to any normal deep neural network but here the generator and discriminator are separately trained which in combination trains the GAN. Optimisers like Adam are used which are based on backpropagation techniques.

**Chapter 4**

**Project Implementation**

**4.1 Dataset**

The dataset is sourced from face -sketch image pairs from CUHK database. CUHK Face Sketch database (CUFS) is for research on face sketch synthesis and face sketch recognition. It includes 188 faces from the Chinese University of Hong Kong (CUHK) student database, 123 faces from the AR database , and 295 faces from the XM2VTS database . There are 606 faces in total. For each face, there is a sketch drawn by an artist based on a photo taken in a frontal pose, under normal lighting condition, and with a neutral expression. Accordingly, the dataset is divided into training, test and validation sets in the ratio 70:25:5.

**4.2 Experimental Configuration**

Data Pre-processing:

-All images are resized to 256x256

-Input sketch image and ground truth Face image are combined by concatenating horizontally to form a single image. Above process is followed as a requisite for Conditional GAN. The combined image is then fed to the model.

**4.3 Network Architecture**

1)Using Conditional GAN to perform sketch to image translation.

The model consists of a generator to generate images and a discriminator to verify whether they are coming from fake or real source.

Here for a data, combined image of input image(sketch) and ground truth image is fed to the generator. The ground truth image acts as the conditional component.

2)Generator is using a U-net architecture[9]. It consists of an encoder-decoder network Here the 256x256x3 input image is fed which goes through layers of down sampling followed by up sampling to generate the face image. Here i*th* layer of encoder concatenated with n-i *th* layer of decoder where n is total no of layers in encoder part.

All ReLUs in the encoder is leaky, with slope 0.2, while ReLUs in the decoder is not leaky

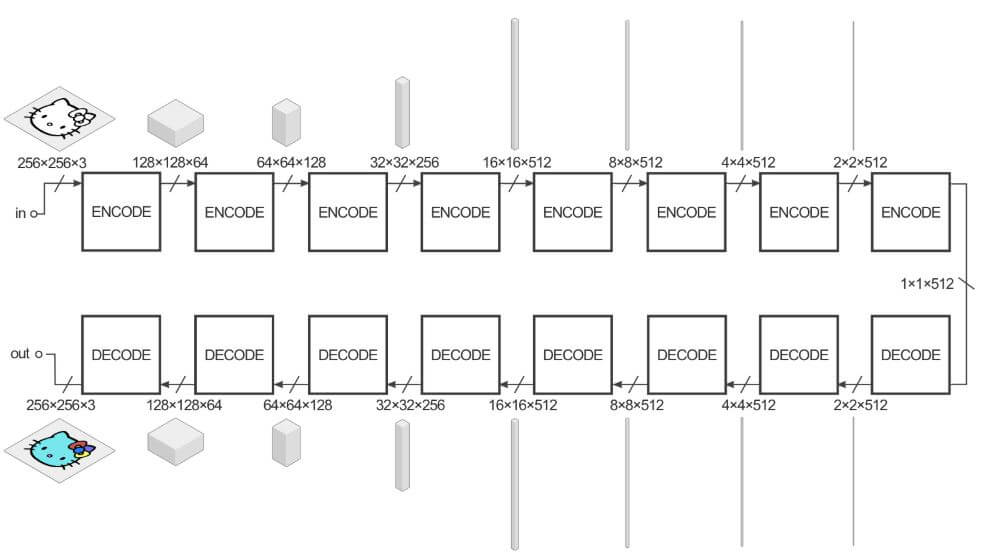


Fig4.1. Generator Architecture- consisting of encoder decode network

3)The generated image, input image pair then compared with discriminator to check how close the generated image is to the target image.

Discriminator penalizes structure at the scale of patches. This discriminator tries to classify if each NXN patch in an image is real or fake. This discriminator is run convolutionally across the image, averaging all responses to provide the ultimate output of D. A convolution is applied after the last layer to map to a 1-dimensional output, followed by a Sigmoid function. BatchNorm is not applied to the first C64 layer. All ReLUs are leaky, with slope 0.2

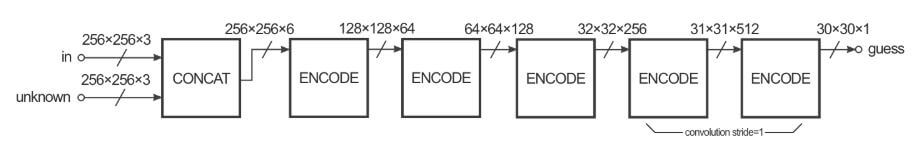


Fig4.2 Discriminator Architecture

3)**Loss functions**:

Normal adversarial losses along with L1 norm loss[13] is done on the images generated by generator to check how much it differs from target image.

L1 norm loss calculates the loss pixel wise in the two images and sums up the loss.

4)**Training:**

For training mini batch SGD is used alongside Adam Solver[14].

The model is run for 200 epochs using a pre-trained model on Google Colab environment using their GPU.

**Chapter 5**

**Results**

**5.1 Accuracy**

The face images generated are fairly accurate and can be easily identified from a perceptual viewpoint. Here given below is an example of the input sketch, generated image and the ground truth face image

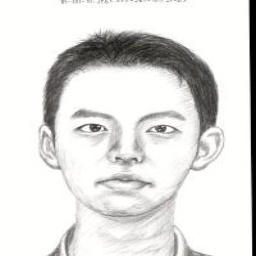


Fig5.1 Input Sketch, Generated Face image and Ground truth image

**5.2 Experiments**

Two standard image similarity metrics were used to check the accuracy of our result. The first test is Fréchet Inception Distance (FID) between the ground truth image and the generated images. The second one is another such similarity measurement metric Feature Similarity Index (FSIM). The experiment was performed gradually over the dataset taking on random 10% of the dataset and gradually increasing the dataset.

**5.2.1 Database Description** Experiments are performed on the test results generated. We conducted experiments publicly available dataset: the CUHK Face Sketch

. The composition of the datasets is briefly introduced below.

- The CUFS dataset consists of 327 face photos from two datasets:

* the CUHK student dataset [54] (188 persons),
* the AR dataset [55] (139 persons),

Experiment is performed on test results consisting 82 images

For each person, there are one face photo and one face sketch drawn by the artist.

Female photos: 29

Male Photos:53

**Dataset partition**. There are great divergences in the experimental settings among existing works. In this paper, we follow the most used settings and split the dataset in the following ways. Dataset is partitioned into training, testing and validation sets in the ratio 75:20:5. Experiments are performed on the test set.

**Pre-processing**. For each face, there is a sketch drawn by an artist based on a photo taken in a frontal pose, under normal lighting condition, and with a neutral expression. Following existing methods, all these face images (photos and sketches) are geometrically aligned based on three points: two eye centers and the mouth center. The aligned images are cropped to the size of 256X 256.

Of the test dataset randomly 10% is chosen and the experiment is performed. Then same process performed on 20% and repeated until experiment is performed on the whole dataset.

**FID**: Fréchet Inception distance (FID) used to evaluate the realism and variation of synthesized photos and sketches. FID measures the Earth-Mover Distance (EMD) between the distribution of generated samples and that of the ground-truth samples, in the feature space. The 2048-dimensional feature of the Inception-v3 network pre-trained on ImageNet is used for the experiment. Lower FID values means closer distances between the generated face image and the ground truth images.

**Experimental Results**

FID

|  |  |
| --- | --- |
| **Percentage of Dataset** | **FID Score obtained** |
| 10 | 129.8941362722024 |
| 20 | 96.96407251277694 |
| 30 | 101.5418828753562 |
| 40 | 102.19270259222989 |
| 50 | 106.29098120348479 |
| 60 | 103.25030773186066 |
| 70 | 103.39977730588271 |
| 80 | 98.59174647477053 |
| 90 | 94.77772414677688 |
| 100 | 93.38565774468366 |

Fig5.1. FID scores obtained on various sized data of result.

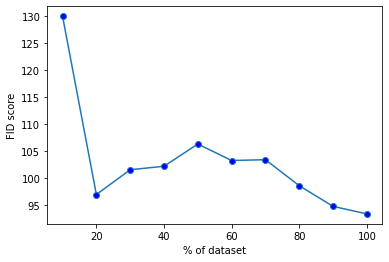


Figure 5.2 graphical representation of the experiment with x-axis representing the %of the dataset and y-axis representing the score achieved.

FSIM

Feature Similarity Index Metric between a synthesized image and the corresponding ground-truth image to objectively assess the quality of the synthesized image. In FSIM, the phase congruency (PC) and the image gradient magnitude (GM) is employed as features, and the feature similarity between a test image and its corresponding reference image is employed as the quality index. Notably, although FSIM works well for evaluating quality of natural images and has become a prevalent metric in the face photo-sketch synthesis community, it is of low consistency with human perception for synthesized face photos and sketches.

**Experimental Results**

|  |  |
| --- | --- |
| **Percentage of Dataset** | **FSIM score** |
| 10 | 0.513098781 |
| 20 | 0.470293226 |
| 30 | 0.501717344 |
| 40 | 0.48682433 |
| 50 | 0.48682433 |
| 60 | 0.486779394 |
| 70 | 0.5027147 |
| 80 | 0.516639213 |
| 90 | 0.486637213 |
| 100 | 0.532996056 |

Figure 5.3 FSIM scores obtained on various sized data of result

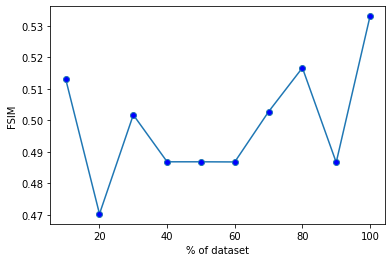


Figure 5.4 represents the graphical representation of the experimental results.

x-axis is the %of dataset and y-axis represents the FSIM score achieved.

**Analysis and Discussions**

As we can see from the figures of FID there is a steady decrease in FID score with increase in size of dataset and there are some variations which is due to the randomness of the data chosen. Lowest score 93 is achieved which is quite good. If larger dataset is available then we think even better score can be achieved.

In the other experiment of FSIM the graphical data shows a lot of variations and its it tough to draw any conclusions. As dataset is chosen randomly that also impacts the experiment. But we can see that for 80% of dataset and when complete dataset is used a higher score of 0.535 and 0.555 is achieved. Considerable amount of similarity has been achieved from this experiment. FSIM is designed for evaluating the quality degradation of photos caused by blurring, noise, or compression, and is not suitable for evaluating visual quality of sketches and so higher FSIM value will lead to lesser perceptual quality.

This demonstrates that our methods dramatically improve the realism of the synthesized sketches

**Comparisons**

Our method is compared with previous existing methods on the above two parameters of FID and FSIM tests with similar experimental settings as ours.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Criterion** | **Dataset** | **Traditional Methods** | **Deep Methods** | **Our Method** |
| **FID** | CUFS | 106.9 | 99.7 | 93 |
| **FSIM** | CUFS | 69.6 | 69.3 | 53.5 |

Fig5.5 Comparison values of experiments with other methods

We can see that our method performs better in FID test compared to older traditional methods and other Deep Learning based methods. In case of the other test FSIM our methods performs considerably but not as good as other methods.

**Chapter 6**

**Conclusion**

**6.1 Conclusion**

The model has performed quite well in generating realistic face images which are quite accurate even in loose comparison between ground truth images and fake generated images.

The model could work better with if dataset with more labelled images are found.

Certain deformities were found in eyes which we feel can be rectified with addition of component-based checking.

**6.2 Future Scope**

There is a lot of scope of future work and adding other features. One primary addition is adding feature of age detection in the generated images. For example, for a missing person whose sketch is available, his face can have certain changes in features with age. Specially for a child his/her face can have considerable changes with passage of time. So, our model should accordingly generate the face image taking the age or time elapsed in cognisance.

**Chapter 7**

**Bibliography**

[1] Xiaoou Tang and Xiaogang Wang, Department of Information Engineering the Chinese University of Hong Kong Shatin, Hong Kong Face Sketch Synthesis and Recognition

[2] Jian Zhao1, Xie Xie1, Lin Wang1, Meng Cao1, Miao Zhang1 1School of Information Science and Technology, Northwest University, Xi’an 710127, China Generating Photographic Faces from the Sketch Guided by Attribute Using GAN,

[3] Y. Li, M. Savvides, and V. Bhagavatula, “Illumination tolerance face recognition using a novel face from sketch synthesis approach and advanced correlation filters,” *in Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, 2006, pp. 357– 360.*

[4] X. Gao, J. Zhong, J. Li, and C. Tian, “Face sketch synthesis using e-hmm and selective ensemble,” *IEEE Transactions on Circuits and Systems for Video Technology, vol. 18, no. 4, pp. 487–496, 2008*

[5]CNN. <URL:https://towardsdatascience.com/a-comprehensive-guide-to-> convolutional-neural-networks-the-eli5-way-3bd2b1164a53

[6] GAN. URL: <https://machinelearningmastery.com/what-are-generative-adversarial-networks-gans/>

[7] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, Yoshua Bengio Departement d’informatique et de recherche op ´ erationnelle ´ Universite de Montr ´ eal ´ Montreal, QC H3C 3J7, *In NIPS, 2014*.

[8] J. Gauthier. Conditional generative adversarial nets for convolutional face generation. Class Project for Stanford CS231N: Convolutional Neural Networks for Visual Recognition,Winter semester, (5):2, 2014

[9] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional

networks for biomedical image segmentation. In MICCAI, 2015. 2, 3

[10] Phillip Isola, Jun-Yan Zhu Tinghui, Zhou Alexei, A. Efros. Image-to-Image Translation with Conditional Adversarial Networks, *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR),*2017

[11] X. Wang and X. Tang, “Face Photo-Sketch Synthesis and Recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), Vol. 31, 2009.*

[12] C. Li and M. Wand. Precomputed real-time texture synthesis with markovian generative adversarial networks. ECCV,2016. 2, 4

[13] L1 norm. URL: <https://machinelearningmastery.com/vector-norms-machine-learning/>

[14] [Diederik P.Kingma](https://arxiv.org/search/cs?searchtype=author&query=Kingma%2C+D+P), [Jimmy Ba](https://arxiv.org/search/cs?searchtype=author&query=Ba%2C+J). Adam: A Method for Stochastic Optimization, *3rd International Conference for Learning Representations, San Diego, 2015*