CONVERTING HAND-DRAWN SKETCH TO REALISTIC IMAGE USING GANS

Capstone Project Report

Submitted by:

Prakhar Bhateja	**101903098**
Shobhit Gupta	**101903095**
Arpan Rehan	**102083014**
Umar Tariq Wani	**102083065**

BE Third Year- COE

CPG No. 100

Under the Mentorship of

Dr. Yadwinder Singh

CSED Department

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

Computer Science and Engineering Department

Thapar Institute of Engineering and Technology,

MARCH - DECEMBER 2022

ABSTRACT

This project is introducing a technique which will help an appropriate user to convert his/her hand drawn sketch into a realistic image. It can help people present in the forensic art domain to provide a more genuine image of any criminal to the police based on a witness's description.

Our idea is to develop a method that will make the process of converting hand drawn sketches to realistic images very simple, efficient and economical. This system can lead to a fast and reliable process with minimum chances of error. This will help organizations to best utilize their resources.

This project development started after deep research of the current solutions and their short-comings. The application is built using the modern tools and technologies to ensure reliability and ease of maintenance in future.

DECLARATION

We hereby declare that the design principles and the report of the project entitled CONVERTING HAND-DRAWN SKETCH TO REALISTIC IMAGE USING GANS is an authentic record of our own work carried out in the Computer Science and Engineering Department, TIET, Patiala, under the guidance of Dr. Yadwinder Singh during 6th and 7th semesters.

Date: 26 / 08 / 2022

Roll No.	Name	Signature
101903098	PRAKHAR BHATEJA	Prakhar
101903095	SHOBHIT GUPTA	Shobhit
102083014	ARPAN REHAN	Arpan
102083065	UMAR TARIQ WANI	Umar

Counter Signed By

Faculty Mentor:

Yadwinder Singh

Dr. Yadwinder Singh

Computer Science & Engineering Department

TIET, Patiala

ACKNOWLEDGEMENT

We would like to express our thanks to our mentor Dr. Yadwinder Singh. He has been of great help in our venture, and an indispensable resource of technical knowledge. He is truly an amazing mentor to have.

We are also thankful to Dr. Maninder Singh, Head, Computer Science and Engineering Department, entire faculty and staff of Computer Science and Engineering Department, and also our friends who devoted their valuable time and helped us in all possible ways towards successful completion of this project. We thank all those who have contributed either directly or indirectly towards this project.

Lastly, we would also like to thank our families for their unyielding love and encouragement. They always wanted the best for us and we admire their determination and sacrifice.

Date: 26 / 08 / 2022

Roll No.	Name	Signature
101903098	PRAKHAR BHATEJA	Prakhar
101903095	SHOBHIT GUPTA	Shobhit
102083014	ARPAN REHAN	Arpan
102083065	UMAR TARIQ WANI	Umar

TABLE OF CONTENTS

ABSTRACT		**i**
DECLARATION		**ii**
ACKNOWLEDGEMENT		**iii**
LIST OF FIGURES		**iv**
LIST OF TABLES		**v**
LIST OF ABBREVIATIONS		**vi**
1-	**INTRODUCTION**	
1.1	Project Overview	1
1.2	Need Analysis	1
1.3	Research Gaps	2
1.4	Problem Definition and Scope	3
1.5	Assumptions and Constraints	3
1.6	Standards	4
1.7	Approved Objectives	5
1.8	Methodology Used	5
1.9	Project Outcomes and Deliverables	5
1.10	Novelty of Work	6
2-	**REQUIREMENT ANALYSIS**	
2.1	Literature Survey	7
2.1.1	Theory Associated with Problem Area	8
2.1.2	Existing Systems and Solutions	8
2.1.3	Research Findings for Existing Literature	9
2.1.4	Problem Definition	17
2.1.5	Summary	18
2.2	Summary Software Requirements Specification	18

2.2.1 Introduction 18

2.2.1.1 Purpose 18

2.2.1.2 Intended Audience and Reading Suggestions 18

2.2.1.3 Project Scope 19

2.2.2 Overall Description 19

2.2.2.1 Product Perspective 19

2.2.2.2 Product Features 21

2.2.3 External Interface Requirements 21

2.2.3.1 User Interfaces 21

2.2.3.2 Hardware Interfaces 21

2.2.3.3 Software Interfaces 21

2.2.4 Other Non-functional Requirements 21

2.2.4.1 Performance Requirements 21

2.2.4.2 Safety Requirements 22

2.2.4.3 Security Requirements 22

2.3 Cost Analysis 22

2.4 Risk Analysis 22

3- METHODOLOGY ADOPTED

3.1 Investigative Techniques 23

3.2 Proposed Solution 24

3.3 Work Breakdown Structure 26

3.4 Tools and Technologies 26

4- DESIGN SPECIFICATIONS

4.1 System Architecture 28

4.2 Design Level Diagrams 28

4.3 User Interface Diagrams 30

4.4 Snapshots of Working Prototype 30

5- CONCLUSIONS AND FUTURE SCOPE

5.1 Work Accomplished 33

5.2 Conclusions 33

5.3 Environmental, Economic and Societal Benefits 33

5.4 Future Work Plan 33

APPENDIX A: REFERENCES 34

APPENDIX B: PLAGIARISM REPORT 35

LIST OF TABLES

Table No.	Caption	Page No.
Table 1	Research Gaps	2
Table 2	Assumptions	3
Table 3	Constraints	4
Table 4	Standards	4
Table 5	Methodology	5
Table 6	Research Findings	9
Table 7	Cost Analysis	23

LIST OF FIGURES

Figure No.	Caption	Page No.
Figure 1	Block Diagram of Hand Made Sketch to Realistic Image Model	22
Figure 2	Proposed Solution for Converting hand-drawn sketch to realistic image using gans.	26
Figure 3	showing design of generator and discriminator model	27
Figure 4	Work break-down Structure of our project	28
Figure 5	System Architecture	29
Figure 6	Design Level Diagrams	30
Figure 7	User Interface Diagrams	31
Figure 8	Coding implementation of model	32
Figure 9	Coding implementation of model	32
Figure 10	Output generated from our model	33

LIST OF Abbreviations

GAN	Generative Adversarial Networks
CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
VAE	variational autoencoder
SBIR	sketch-based image retrieval

INTRODUCTION

1.1 Project Overview

The power and promise of deep generative models such as GANs[1] lie in their ability to synthesize endless realistic, diverse, and novel content with minimal user effort. The potential utility of these models continues to grow thanks to the increased quality and resolution of large-scale generative models in recent years. Nonetheless, the training of high-quality generative models demands high-performance computing platforms[2], putting the process out of reach for most users. Furthermore, training a high-quality model requires expensive large-scale data collection and careful pre-processing.This leads to the question: how can an ordinary user create their own generative model? A user creating artwork with cats might not want a generic model of cats, but a bespoke model of special cats in a particular desired pose: nearby, reclining, or all looking left. To obtain such a customized model, must the user curate thousands of reclining left-looking cat images and then find an expert to invest months of time in model training and parameter tuning?In this project, we have the task of creating a generative model from just a handful of hand-drawn sketches and we will create a generative model of realistic images from hand-drawn sketches.In our case, the input is a 2D sketch and the output is a network with millions of opaque parameters that control algorithm behavior to make images. We use our method to create several new customized GAN models, and we show that these modified models can be used for several applications such as generating new samples, interpolating between two generated images as well as editing a natural photograph. A user only needs to provide one or a few exemplar sketches for our method to work effectively. Finally, we benchmark our method to fully characterize its performance.

1.2 Need Analysis

Hand drawn sketches exist all over the world. The conversion of these sketches to realistic images is a solution that can be achieved through Deep learning which is solving many problems in many fields. It is an unsupervised machine learning method that uses neural networks to train itself.

A Lot of data is needed for many neural networks:

When it comes to neural networks a lot of data is needed to efficiently train that model. Acquiring this data can become a bottleneck for machine learning models .Here GAN's stand out due to their unique working using generators and discrimintor which can achieve better results even with less training data.

Less work in human portraits generation using hand drawn sketches:

Although a lot of work has been done in the field of hand drawn sketches, by concentrating on human portraits we can try to improve the existing models in this particular field as not a lot of work has been done in this field and holds many practical applications.

Practical use cases of our model :

There are many use cases for our model .From being used to better identify criminals from their sketches to identifying a person using their sketch and even for creative purposes to achieve realistic images using hand drawn sketches.

1.3 Research Gaps

TABLE 1: Research Gaps

S. No.	Research Gaps
1.	Higher accuracy and realistic portrait can be improved by inputting the output generated from Edge network to another Edge network and at the same time reducing the amount of noise present in the edge data that may further enhance the image quality.
2.	Limitation of database size in the human face domain .
3.	There is not a lot of research on combining different models to achieve better accuracy.

4.	Research on the converting of human sketches of human faces to realistic images is lacking can be improved
5.	The database of human faces for our projects can also be improved by using algorithms to convert realistic images to hand-drawn images and using this to better train our model.

1.4 Problem Definition and Scope

Free hand sketch is a universal communication and art modality that transcends barriers to link human societies. It has been used from ancient times to today, comes naturally to children before writing, and transcends language barriers. Different from other related forms of expression such as professional sketch, forensic sketch, cartoons, technical drawing, and oil paintings, it requires no training and no special equipment. Free-hand sketch can be highly illustrative, despite its concise and abstract nature, making it useful in various scenarios such as communication and design. Therefore, free-hand sketch has been widely studied in the field of computer vision and pattern recognition in order to procure some valuable meaning out of it. So, to address this problem we will be developing an application in finding ways to help concerned people in turning their hand drawn sketch into realistic images using GAN's (Generative Adversarial Networks).

1.5 Assumptions and Constraints

TABLE 2: Assumptions

S. No.	Assumptions
1.	Let us assume that the user has access to a fully functioning computing device.
2.	Users should have access to the required input i.e. a digital pencil to draw
3.	Let us assume that the quality of the given sketch is of the required level i.e. it should not be blurry or distorted.
4.	The dataset on which the model is trained should be genuine.

TABLE 3: Constraints

S. No.	Constraints
1.	The user’s device should be capable of running softwares for making sketches
2.	The given input by the user should be meaningful i.e. related to the dataset we are using.

1.6 Standards:

Table 4: Standards

Phase	Number	Standard
Requirement Specifications	IEEE 830 IEEE 1233	Recommended software practice details set A guide for developing the details of a program's requirements. The language of the measurement function.
Design	IEEE 1016 IEEE 1471	Software design descriptions Recommended practice for architectural description of architectural and software-intensive systems.
Implementation, acquisition and tools.	IEEE 1062 IEEE 1462	Software acquisition CASE tool testing and selection guide CASE tool interconnection guide
Testing	IEEE 829 IEEE 1008 IEEE 1012 IEEE 1028 IEEE 1014	Software test documentation Software unit testing Software verification and validation Software reviews and audits Classification for software anomalies
Maintenance	IEEE 1219	Software maintenance

1.7 Approved Objectives:

1. To design a GAN network that can be applied over hand drawn sketches.

2. To achieve desired accuracy using an optimal number of images.

3. To create a generative model that can produce high quality, realistic images with minimal loss of resolution from hand-drawn sketches.

1.8 Methodology

TABLE 5: Methodology

S. No.	Methodology
1.	We study the research papers of all the proposed technologies like GAN's, Rasterization, Vectorization.
2.	This would help in the formulation of the main problem statement
3.	Our goal is to simplify the user creation of a generative model, we must utilize only a very small amount of user-provided sketch data.
4.	Our objective is to design a GAN that can be applied over rasterized images and vector representation of any hand drawn data.
5.	The end result is to obtain a high quality and high-resolution image without losing too much originality of the image

1.9 Project Outcomes and Deliverables

The final outcome will be capable of generating models of realistic images from hand drawn sketches with as few images as possible and models of realistic photographs where the shape and pose are guided by sketches but where the outputs are realistic images, rather than sketches.

1.10 Novelty of work

Our project involves a unique comparative study of multiple and different machine learning models. Among these models, the best and most efficient model is chosen to make sure we get the right results as frequently as possible.We are also using multiple deep learning models to get the best results and accuracy which many platforms tend to leave out.

We used Ensemble models to combine multiple models. Ensemble methods are techniques that aim at improving the accuracy of results in models by combining multiple models instead of using a single model. The combined models increase the accuracy of the results significantly.

This project includes creation of a GAN model using a handful of sketches and it is possible for us to create a generative model of realistic images from hand-drawn sketches.Developing a model that caters to above mentioned needs will facilitate more technical advances in the field.

2.1 Literature Survey

The progress of deep learning has immensely benefited free-hand sketch research and applications.Sketch research and applications in both industry and academia have boomed. Nowadays,acquiring sketch data is much easier than ever; as well as the rapid development of deep learning techniques that are achieving state-of-the-art performance in diverse artificial intelligence tasks.Some classic research topics e.g., sketch recognition, sketch-based image retrieval, sketch-based 3D shape retrieval, have been re-studied in a deep learning context resulting in significant performance improvements. Some brand-new topics have been proposed based on deep learning, e.g., deep learning-based sketch generation/synthesis, sketch-based model generation, reinforcement learning based sketch abstraction, adversarial sketch-based image editing, graph neural network based sketch recognition, graph convolution-based sketch semantic segmentation, and sketch based software prototyping. Beyond global representation-based tasks e.g., sketch recognition, more instance-level and stroke-level tasks have been further studied or proposed, e.g., instance-level sketch-based image retrieval, and deep stroke-level sketch segmentation.Compared with the conventional approach of representing sketches as static images, the trends of touchscreen acquisition and deep learning have underpinned progress on designing deep network architectures to exploit richer representations of sketch. Thanks to works such as SketchRNN, the sequential nature of free-hand sketches is now widely modeled by recurrent neural networks (RNN). More sketch-based applications have appeared, e.g. the online sketch game QuickDraw2 , and sketch-based commodity search engine3.Some large-scale sketch datasets have been collected, e.g., Sketchy and Google QuickDraw4 – a million-scale sketch dataset (50M+).

2.1.1 Theory Associated With Problem Area

Free-hand sketches are highly illustrative, and have been widely used by humans to depict objects or stories from ancient times to the present. The recent prevalence of touchscreen devices has made sketch creation a much easier task than ever and consequently made sketch-oriented applications increasingly popular.With increase in both the quantity and quality of datasets this area is becoming increasingly useful.Our focus on using GANs on sketches of human faces to convert them into realistic images can be used in many fields such as better criminal portraits .In a GAN, the generator and discriminator models receive input information in the form of a vector. This information could be the label of the input image or some other property. The information is one hot encoded and sent to the generator. The generator takes this vector of information and encodes features from an image like the class (male and female if we are trying to generate images of faces) or properties like hair, nose, eyes etc. To make this Generative and Adversarial process simple, both these blocks are made from Deep Neural Network based architecture which can be trained through forward and backward propagation techniques.

2.1.2 Existing Systems and Solutions

A GAN model requires the collection dataset of examples and knowledge in deep learning.Retrieving images that resemble a human sketch has been extensively studied, including classic methods that rely on feature descriptors, as well as more recent deep learning methods[3,4,5]. Sketch-based image retrieval (SBIR) techniques have powered sketch-based 3D modeling systems (e.g Teddy[6]) as as well as image synthesis systems, including Sketch2Photo[7] and Photo-Sketcher[8]. These seminal works have further inspired deep learning solutions based on image-to-image translation such as Scribbler [9], Sketchy GAN[10], Sketchy COCO[11], and sketch-based face and hair editing[12]. Other relevant work includes sketch recognition and sketch generation [13,14]. Collectively, the above methods have enabled A novice user to synthesize a single natural photograph. After years of development, deep generative models are able to produce high-quality, high-resolution images, powering a wide range of computer vision and graphics applications. Recent examples include image projection and editing with GANs, image-to-image translation, simulation-to-real and domain adaptation. In Particular CycleGAN was also studied which was used to convert hand drawn sketches to images specifically for human portraits.In it least squares was used to achieve the translation from sketch to portrait, and finally PCA was used to complete face matching.The CycleGAN model contains two mapping functions Generator S2P(sketch to photo) and Generator P2S (photo to sketch), and associated adversarial discriminators Discriminator P and Discriminator S. D.P encourages G.S2P to translate sketch into outputs indistinguishable from photo, and vice versa for D.P and G.P2S

2.1.3 Research Findings for Existing Literature

Table 6:Research Findings

S.No.	Roll number	NAME	Paper Title	Technology	Findings	Citation
1	101903098	Prakhar Bhateja	Sketch to color portrait generation with generative adversarial networks and edge constraint.	GAN	● Edge information of the image is extracted, its contour is optimized, and the new model constrains the convergence direction of portrait generation. Experiments show that this modification can effectively solve the edge blur problem of facial image generation. ● An improved version of the Pix2Pix model which is a method used for Image-To-Image Translation With Conditional Adversarial Networks. ● We then modify the loss constraint in the Pix2Pix model. The new loss function can evaluate the deviation of generator network output results and add the image contour	Qingyun Liu , Huihuang Zhao, Ying Wang, Feng Zhang,Manimaran Ramasamy,ZhijunQ iao

					information, which is output from edge network. The quality of the contour is improved, the outline is clear, and the details are accurately maintained. ● Evaluation Metric used: SSIM (Structural Similarity) (Average of test done on 6 images) ● Without Edge Constraint: 72.86% ● With Edge Constraint: 77.75%	
2			Sketch guided and progressive growing GAN for realistic and editable ultrasound(US)image synthesis.	GAN	● This paper presented the first work that can synthesize realistic B-mode US (Ultrasound) images with high-resolution and customized texture editing features. ● To enhance structural details of generated images, they have proposed to introduce auxiliary sketch guidance into a conditional GAN. We superpose the edge sketch onto the object mask and use the composite mask as the network input. ● To generate high resolution US images, we adopt a progressive training strategy to gradually generate high-resolution images from low-resolution images. ● In addition, a feature loss is proposed to minimize the difference of high-level features between the generated and real images, which further improves the quality of generated image.	Jiamin Liang, Xin Yang, Yuhao Huang, Haoming Li, Shuangchi He, Xindi Hu, Zejian Chen, Wufeng Xue, Jun Cheng, Dong Ni

					● The proposed US image synthesis method is quite universal and can also be generalized to the US images of other anatomical structures	
3			Sketch to Building: Architecture Image Translation Based on GAN	GAN	● In this paper,they have presented a two-step method consisting of a supervised GAN and an unsupervised GAN for architecture sketch-to-image translation. First,the supervised GAN learns the mapping from hand-drawn sketch to gray sketch domain, filling in the disconnection of boundary lines and adding more texture details. ● Then the multi-modal unsupervised GAN translates the information-rich gray sketch to various color image outputs. ● An interactive system is designed for the architecture of sketch-to-image translation. This system provides two options for users. Users can either directly draw a sketch, or generate a sketch from an input image. If a loaded image initially generates the sketch, the system would use the loaded image as the style as default. ● This is a two-step method to train the mapping from hand-drawn binary sketch to photo-like color image. This framework fills in the disconnected boundary lines in the sketch and alleviates the domain gap between source and	Sidong Jiang, Yuyao Yan, Yiming Lin, Xi Yang, Kaizhu Huang

					target domains, making the unsupervised GAN in the second part more effective.	
4	102083014	Arpan Rehan	Generating Human Body Images from Freehand Sketches	Geometric refinement module, GauGAN	● Geometry refinement module: It aims to refine an input freehand sketch by using human portrait images to train several part-level networks. ● Image generation module(GauGAN) : It takes a Semantic Segmentation Map (left) as input and produces a photo-realistic image as the output. ● Structure refinement module: ● Relative positions and proportions between body parts in a hand drawn sketch might not be accurate. We thus employ the structure refinement module to refine the relative positions and proportions of body parts to get a globally more consistent body image.	Xian Wu, Chen Wang, Hongbo Fu, Ariel Shamir, Song-Hai Zhang, Shi-Min Hu
5			Generating 3D Mesh model from Sketch.	GAN, Deep Neural Network	● The approach is based on a two step process; in the first step we generate multiple 2.5D views from a single sketch and in the second step we generate a 3D model from these 2.5D views. 2.5D is an extended representation of a 2D image or sketch with depth and normal information. Our approach uses deep neural network based models for each step. ● To generate 3D mesh model from multiple 2.5D views, we use GAN based architecture.	Nitish Bhardwaj,Dhornala Bharadwaj, Alpana Dubey, Accenture labs, Banglore

6			A Sketch Recognition System for Recognizing Free-Hand Course of Action Diagrams		● PaleoSketch is used to recognize basic geometric shapes like rectangles, lines, and ellipses, while our handwriting recognizer (HWR) is responsible for recognizing text, decision graphics, and echelon modifiers.	
7	102083065	Umar Tariq Wani	Freehand Sketch Recognition Using Deep Features	CNNs	● Convolutional Neural Networks (CNNs): ● We use two popular CNNs for our experiments – Imagenet CNN and a modified version of LeNet CNN ● Methods for freehand sketch recognition exhibit two broad themes.domain specific and general. ● In the former category, recognition systems have been built for sketches related to mathematical expressions, emergency management and military drawings and chemistry diagrams. ● Convolutional Neural Networks (CNNs) ● Average Accuracy of 54% across the 250 categories when 80% of the sketches are used for training. ● Lenet CNN ● We train a modified version of Lenet CNN using the MNIST handwritten digit dataset for 10000 iterations at which point, accuracies are around 98%	Ravi Kiran Sarvadevabhatla,R. Venkatesh Babu

8			Creating Face Images from Freehand Sketches	GAN	● Image-to-Image Translation: ● Given an input image from one domain, an image-to-image translation model outputs a corresponding image from another domain and preserves the content in the input image. Existing image-to-image translation models are based on generative adversarial networks conditioned on images. ● Sketch-based Image Generation: ● Sketch-based image generation is a hot topic in multimedia and computer graphics. Given a sketch which describes the desired scene layout with text labels for objects, traditional methods, such as Sketch2Photo and PhotoSketcher , search image patches from a large-scale image dataset and fuse the retrieved image patches together according to the sketch. ● Face Image Generation and Editing: ● Face image generation and editing have made tremendous progress. Generative adversarial network (GAN) , which generates images from noise, is widely used in a wide range of applications . Using GAN, realistic face images can be generated from noise vectors. DCGAN introduces a novel network to stabilize training of GAN. PGGAN utilizes a progressively growing architecture to generate high-resolution face images.	Yuhang Li,Xuejin Chen*,Binxin Yang,Zihan Chen,Zhihua Cheng,Zheng-Jun Zha

9	101903095	Shobhit Gupta	Freehand Sketching Portrait Recognition with Least Square CycleGAN	GAN	● CycleGAN improved by least squares to achieve the translation from sketch to portrait, and finally PCA was used to complete face matching. ● CycleGAN model contains two mapping functions Generator S2P: sketch → photo and Generator P2S: photo → sketch, and associated adversarial discriminators Discriminator P and Discriminator S. D.P encourages G.S2P to translate sketch into outputs indistinguishable from photo, and vice versa for D.P and G.P2S ● This does not need paired data, only one set of input images (such as sketch portraits) and a set of output images (such as real portraits). namely a Sketch→Photo one-way GAN plus a Photo→Sketch one-way GAN. ● Least Squares Loss instead of the original GAN Loss, that make network training more stable and improve image quality. ● The average score of SSIM and PSNR is 0.844 and 17.331, and the average success rate of recognition is 88.2% for Rank10.	Xiaosa Gou, Bingguo Liu, Guodong Liu , Binghui Lu, Yu Gan and Fengdong Chen
10			Vectorization and Rasterization: Self-Supervised Learning for Sketch and Handwriting	RNN,CNN	● A Self-supervised pre-text task for sketches and handwriting data ● A pre-text task should encode high-level semantic understanding of the data that can be used to solve other downstream tasks like classification, retrieval	Ayan Kumar Bhunia,Pinaki Nath Chowdhury, Yongxin Yang, Timothy M. Hospedales,

					● Vectorization:For translating an image to its sequential point coordinate equivalent, image encoder EI (·) can beany state-of-the-art convolutional neural network] such as ResNet. To predict the sequential point coordinates, decoder DV (·) could be any sequential network, e.g. RNN ● Rasterization:To translate a sequence of point coordinates V to its equivalent image representation I, any sequential network such as RNN [12, 20] or Transformer ,could be used as the encoder EV (·), and we experiment with both. ● On QuickDraw, Top-1 accuracy of 71.9% and 67.2% is obtained for sketch images and sketch vectors respectively,approaching the supervised counterparts of 76.1% and 73.5%. For TU-Berlin accuracies of 70.6% and 55.6% also approach the supervised figures 78.6% and 62.9%	Tao Xiang Yi-Zhe Song
11			StyleMeUp: Towards Style-Agnostic Sketch-Based Image Retrieval	● VAE	● A novel style-agnostic SBIR model is proposed. Different from existing models, a cross-modal variational autoencoder (VAE) is employed to explicitly disentangle each sketch into a semantic content part shared with the corresponding photo, and a style part unique to the sketcher. Importantly, to make our model dynamically adaptable to any unseen user styles, we	Aneeshan Sain,Ayan Kumar Bhunia ,Yongxin Yang, Tao Xiang,Yi-Zhe Song

					propose to metatrain our cross-modal VAE by adding two style-adaptive components: a set of feature transformation layers to its encoder and a regulariser to the disentangled semantic content latent code ● The core model is a VAE framework that disentangles the model variant and invariant semantics in a sketch in a crossmodal translation setting. While a regulariser network regularizes parameters of the invariant component (Ωinv), feature transformation (FT) layers aid in style-agnostic encoding following a meta learning paradigm. ● Accuracy in FG-SBIR: Chair V2: (top-q score) @1-62.86 @10-91.14 @1-36.47 @10- 81.83 Accuracy IN SBIR Sketchy (ext): mAP-0.905 P@200-0.927 TU Berlin (ext) mAP-0.778 P@200-0.79	

2.1.4 Problem Identified

Dataset: There is a limited amount of hand-drawn sketches of realistic human faces.We will use the data from databases: CUHK database, AR database and XM2VTS database to train our model and improve its accuracy.

Combination of models: There is less research in the field of human portrait sketches therefore existing models have yet to be tried in this field .Using a combination of different models like ConditionalGAN with CycleGAN we might be able to improve the final result.

2.1.5 SUMMARY

Our model is built by taking raw testing/training data from reliable/official resources .Once this data is collected, it is further augmented and preprocessed by our model which removes its abnormalities and extracts the crucial parts to make the data more useful and easier to work with. Once the hand drawn sketches are received, they further undergo image preprocessing, normalization and background segmentation and thus the model is trained using GAN. Once our model is trained and tested using the collected data, it is ready to input new data from the user. Once the hand drawn sketch from the user is taken, it is run through the model and a realistic image of what we get as the output. Moreover our project also explores different deep learning algorithms to come up with the enhanced efficiency of our model.Although work in this field has been done but by focusing on human portraits and testing different combinations of models, we hope to achieve better accuracy and a cleaner image as our output .

2.2 Software Requirement Specifications

2.2.1 Introduction

2.2.1.1 Purpose

This document is intended to provide a detailed overview of our software product, its parameters and its objectives which include conversion of any free hand sketch to its realistic counterpart using GAN's. This document specifies in detail various functional and non-functional requirements and also specifies which software feature satisfies these requirements. It also describes different constraints and standards that apply to this domain's software. It includes in the development of this software description of all software / hardware and third-party dependencies.

2.2.1.2 Intended Audience and Reading Suggestions

Primarily our user base will include the forensic sketch artists whose work is to interview victims or witnesses of a crime scene, and create sketch drawings that are used by police to identify and apprehend criminal suspects. Despite having the sketch, investigators still find it difficult to properly identify the criminal, since the hand drawn sketch misses the essence of realism. Our product aims to bridge the gap between a drawing and realistic image, by transforming the hand drawn sketch into realistic drawing. We believe that, once our product is released in the market and people start to believe in us and in our vision, they won't be disappointed.

2.2.1.3 Project Scope

Free-hand sketches are highly illustrative, and have been widely used by humans to depict objects or stories from ancient times to the present. Sketches can convey many words, or even concepts that are hard to convey at all in words. Free-hand sketch can be illustrative, despite its highly concise and abstract nature. Therefore, free-hand sketch has been widely studied in computer vision and pattern recognition, computer graphics, human computer interaction, robotics, and cognitive science communities.

Traditionally, creating a GAN model has required the collection of a large-scale dataset of examples and specialized knowledge in deep learning. Nonetheless, the training of high-quality generative models demands high-performance computing platforms, putting the process out of reach for most users. Furthermore, training a high-quality model requires expensive large-scale data collection and careful pre-processing. This leads to the question: how can an ordinary user create their own generative model? The major scope of this project includes creation of a GAN model using a handful of sketches and we also wish to understand whether it is possible for us to create a generative model of realistic images from hand-drawn sketches. Further we will try to improve the generated images by tweaking the model parameters, preprocessing the input sketches using well known image preprocessing techniques like Gamma Correction, Histogram equalization, Image Regularization, Edge Smoothing Techniques etc. Developing a model that caters to above mentioned needs will facilitate more technical advances in the field.

2.2.2 Overall Description

2.2.2.1 Product Perspectives

The power and promise of deep generative models such as GANs lie in their ability to synthesize endless realistic, diverse, and novel content with minimal user effort. Automatically converting sketch images into realistic images has significant application value in the fields of digital entertainment, art, law enforcement, and other industries.

1. In the law enforcement industry, there is a big difference between an image drawn by the police or a medical expert according to an oral description from an eyewitness and a portrait image with low recognition. Manually generating a real color portrait requires a skilled labor force with experience in drawing and painting as well as investigation. Automatic generation of realistic facial portraits based on sketch images with generative adversarial networks (GANs) can improve the possibility of identification and enhance police efficiency in solving cases.

2. In the comic domain of the entertainment industry, cartoon sketch coloring is being highly implemented to fill colors into the black-and-white anime sketches and to obtain the colored anime images which are more visually appealing in nature. At present, cartoon sketch coloring mainly relies on the anime/comic painters. It takes a lot of time and effort to manually color the anime sketches and the coloring effect is influenced by the professional ability of the anime painters. In order to reduce the difficulties of manual coloring, it is very important to design an appropriate automatic coloring method that can be used to avoid the complicated work procedures generated by manual coloring. Recently, it has become a new research hotspot in the field of deep learning and many generative adversarial networks (GANs) have been used to design appropriate coloring methods.

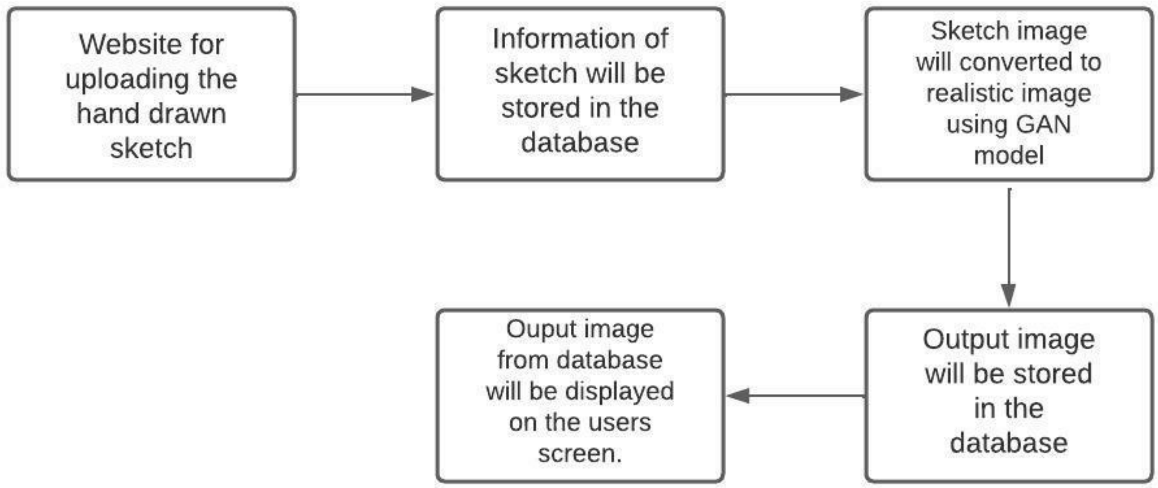

FIGURE 1: Block Diagram of Hand Made Sketch to Realistic Image Model

2.2.2.2 Product Features

Some of the feature of our product are:

- Dedicated web interface for converting hand-drawn sketches to realistic images.
- Easy upload / draw option.
- Highly beneficial for concerned users to generate high quality and detailed images from hand-drawn sketches.

2.2.3 External Interface Requirements

2.2.3.1 User Interfaces

In our project our task is creating a generative model from just a handful of hand-drawn sketches and we will create a generative model of realistic images from hand-drawn sketches. Google colab and google drive can be linked with each other. Whenever we need to predict, images can be imported from the drive and prediction can be done.

2.2.3.2 Software Interfaces

- Any Operating System (Windows/MacOS/Linux).
- Google Colab & Jupyter notebook are used as IDE and the code is written in Python .

2.2.4 Other Non-functional Requirements

2.2.4.1 Performance Requirements

- Optimum performance time of sketch recognition, text recognition and image description algorithms.
- Memory should be available for training.
- Prediction should take less time.
- Accurate and efficient results.
- Low downtime.
- Easy to scale.

2.2.4.2 Safety Requirements

- High scale testing should be done before deploying the machine learning models.

2.3 Cost Analysis

Table 7: Cost Analysis

Machine learning Course	Rs. 2000
Generative Adversarial Networks (GAN) Course	Rs. 1500
Total	Rs. 3500

2.4 Risk Analysis

There are few risk factors included in the building of our project.

- Poor quality of input data such as images can decrease accuracy of result.
- Lack of dataset can lead to degradation of the model's performance
- Overfitting can be misinterpreted as the model having greater accuracy which in reality can be caused due to false positives.

3.1 Investigative Techniques

Investigative Technique Involved: *EXPERIMENTAL*

In our project ‘Converting hand-drawn sketch to realistic image using gan’s’, Experimental investigative techniques have been used with the initial idea of contributing towards society using our engineering skills.

An Experimental Investigative Technique involves coming up with an idea, designing the required procedure for executing the idea and then coming up with the final hypothesis for the desired project. Our idea comprised of converting hand drawn sketches into realistic images using GAN. While building the GAN model, we can apply different activation functions ,loss functions and optimizers. Using different strategies, we can increase accuracy.

Independent variables are those variables which are not dependent on other factors for their value change and are modified for evaluating the effect of dependent variables. The following are the independent variables in our project:

- Input hand-drawn sketch
- Dataset is independent

Dependent variables are those variables which depend on other factors for their evaluation i.e. independent variables are modified to determine the value of dependent variables. These are the variables which act as the motivation for the experiment. The following are the dependent variables in our project:

- GAN model is dependent on training dataset

Hypothesis

We have only one input, i.e , the hand-drawn sketch for our project and therefore we have devised the following hypothesis from our project experimentation.

- If a hand-drawn sketch of a person is given as input , then the image is analyzed through our model and the platform using our trained model will output a realistic image .

Discussion and conclusion

We will implement different Variations of GANs by tweaking different parameters for our project after data cleaning and data augmentation.Then using a web interface we should be able to input a hand-drawn sketch and get a realistic image as an output.We will compare various models with ours to compare accuracy and quality of generated image from existing models such as Pix2Pix,LLE etc.Many databases are also available such as the CUFS database of The Chinese University of Hong Kong as its Face Sketch that is open as a photo that is collected from three sub-databases: CUHK database, AR database and XM2VTS database, which contain a single positive neutral expression photo of 188, 123 and 295 people respectively.Using this we will try to achieve a better accuracy and a more realistic photo.

3.2 Proposed Solution

We are proposing a system using which one would be able to input the hand-drawn sketch from a user and interpret it and output a realistic image from it.It has the following methodology.

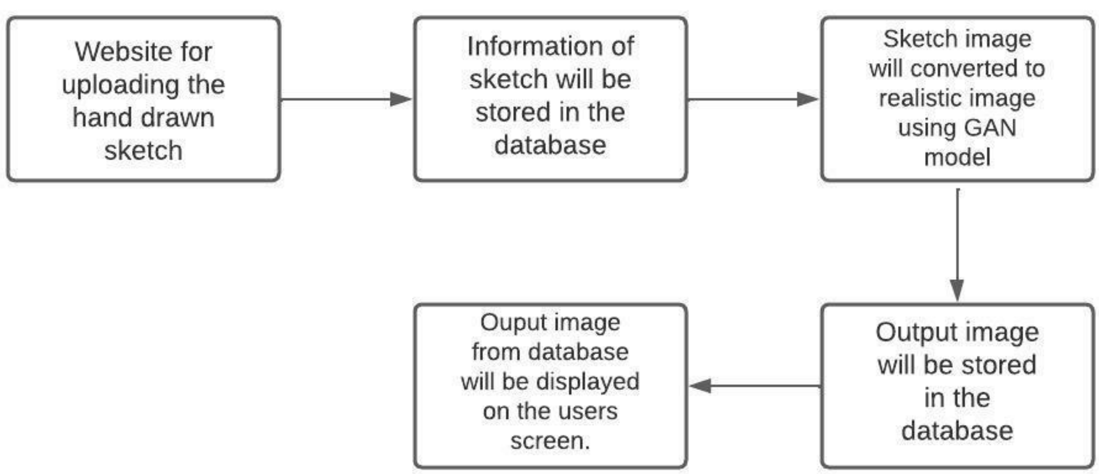

FIGURE 2: Proposed Solution for Converting hand-drawn sketch to realistic image using gans

In a GAN, the generator and discriminator models receive input information in the form of a vector. GENERATOR NETWORK:

The generator part of a GAN learns to create fake data by incorporating feedback from the discriminator. It learns to make the discriminator classify its output as real. Generator training requires tighter integration between the generator and the discriminator. The portion of the GAN that trains the generator includes:

- random input.
- generator network, which transforms the random input into a data instance.
- discriminator network, which classifies the generated data.
- discriminator output.
- generator loss, which penalizes the generator for failing to fool the discriminator.

DISCRIMINATOR NETWORK:

The discriminator in a GAN is simply a classifier. It tries to distinguish real data from the data created by the generator. It uses a simple convolutional neural network with batch normalization. To identify the real and fake images, the output layer stores the probability values for each of the images in the input layer which indicates the probability of the image being real or fake.

The discriminator's training data comes from two sources:

- Real data instances, such as real pictures of people. The discriminator uses these instances as positive examples during training.
- Fake data instances created by the generator. The discriminator uses these instances as negative examples during training.

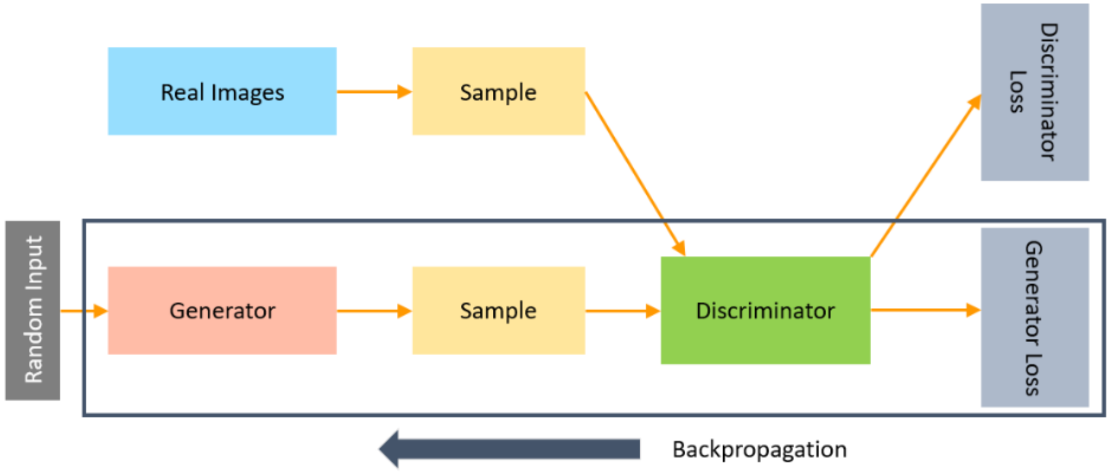

Figure 3: showing design of generator and discriminator model

In our Solution we will Rescale the input image and using methods such as Edge Detection ,Histogram Equalization,Image filtering and segmentation we will process the image and using the above described network we will output a realistic image.

3.3 Work break-down Structure

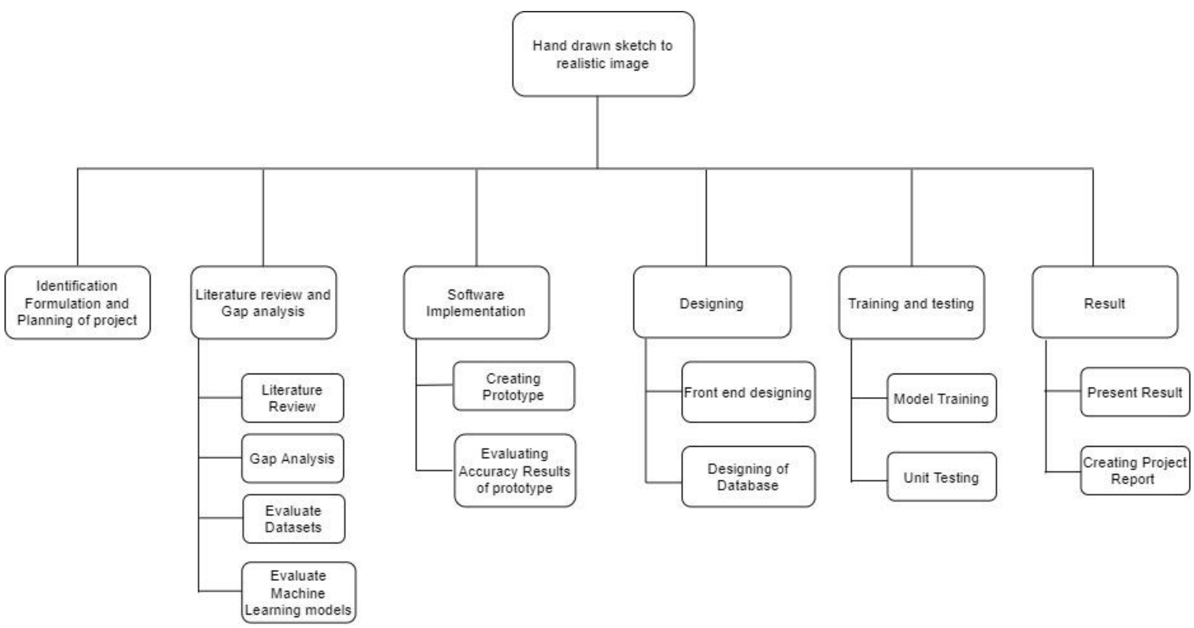

FIGURE 4: Work break-down Structure of our project

3.4 Tools and Technologies Used

- Visual Studio Code: Lightweight code editor. It comes with the host of plugins which makes the development process easier .

- GitHub: GitHub is used to share and manage codebase.
- Jupyter Notebook: easy-to-use interactive data science environment supporting python notebooks.
- Google colab: It is used to test and practice code.
- Google Meet: Video conference tool to collaborate remotely.
- Technologies : Python, Deep learning, OpenCV

This section discussed the design specifications which discusses the system architecture and logical representation of the system.

4.1 System Architecture

A system architecture diagram is used to express the relationship between different components of the system. It is usually drawn when the system involves both software and users.

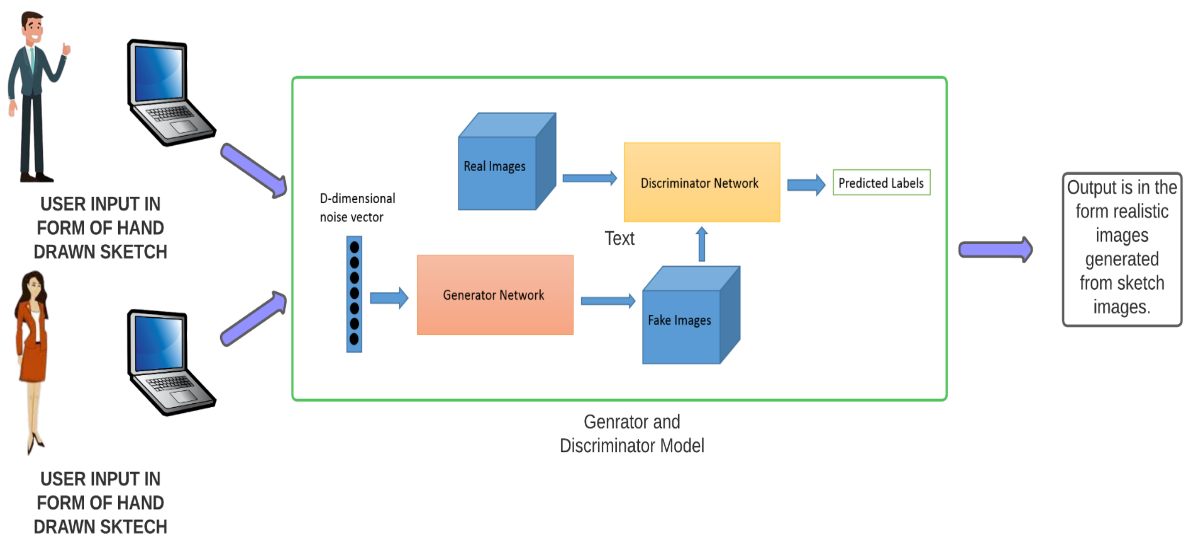

FIGURE 5: System Architecture of Hand Made Sketch to Realistic Image Model

4.2 Design Level Diagrams

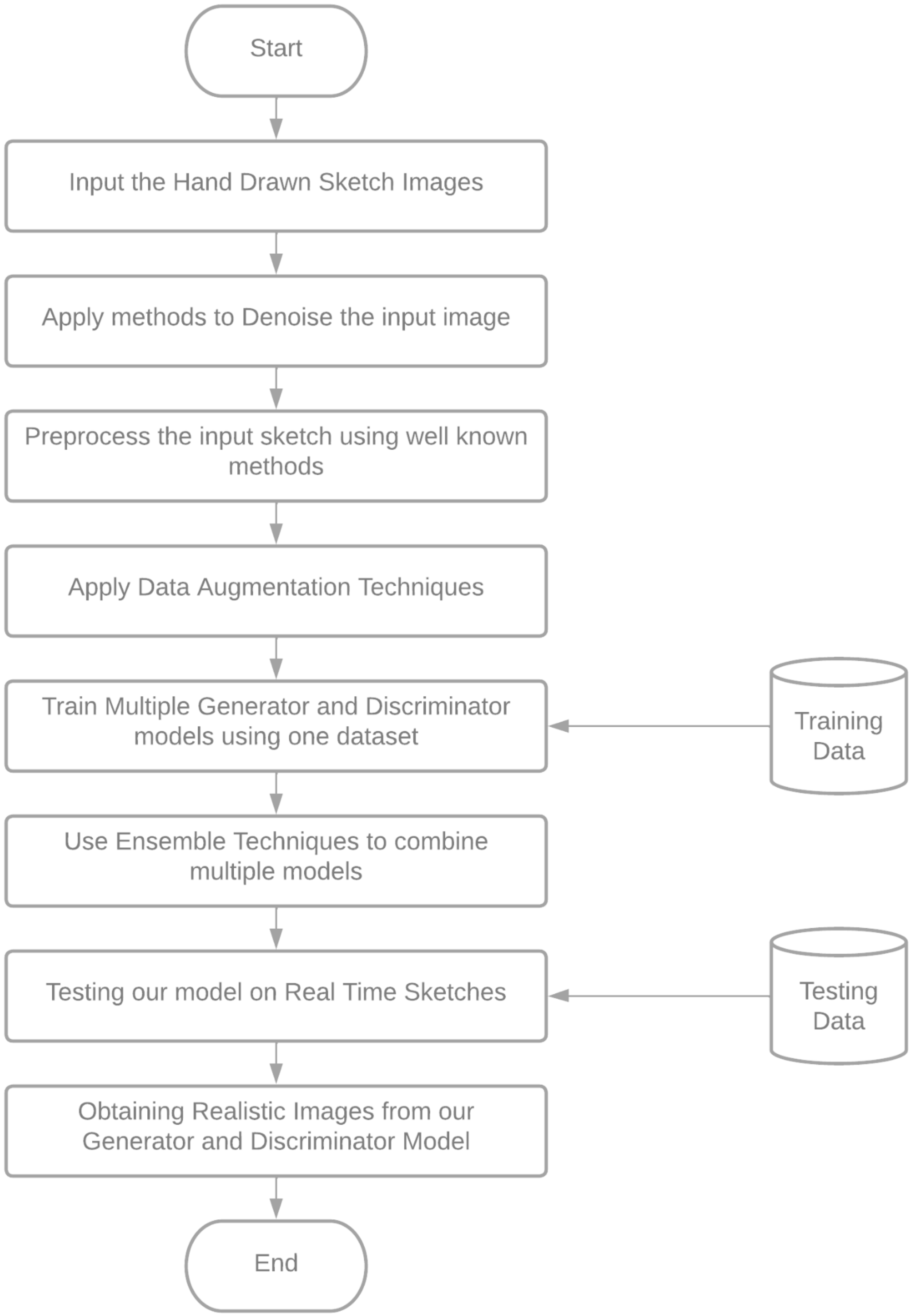

FIGURE 6: Design Level Diagram of Hand Made Sketch to Realistic Image Model

4.3 User Interface Diagrams

User interface are used to model the interaction between the user and the software.

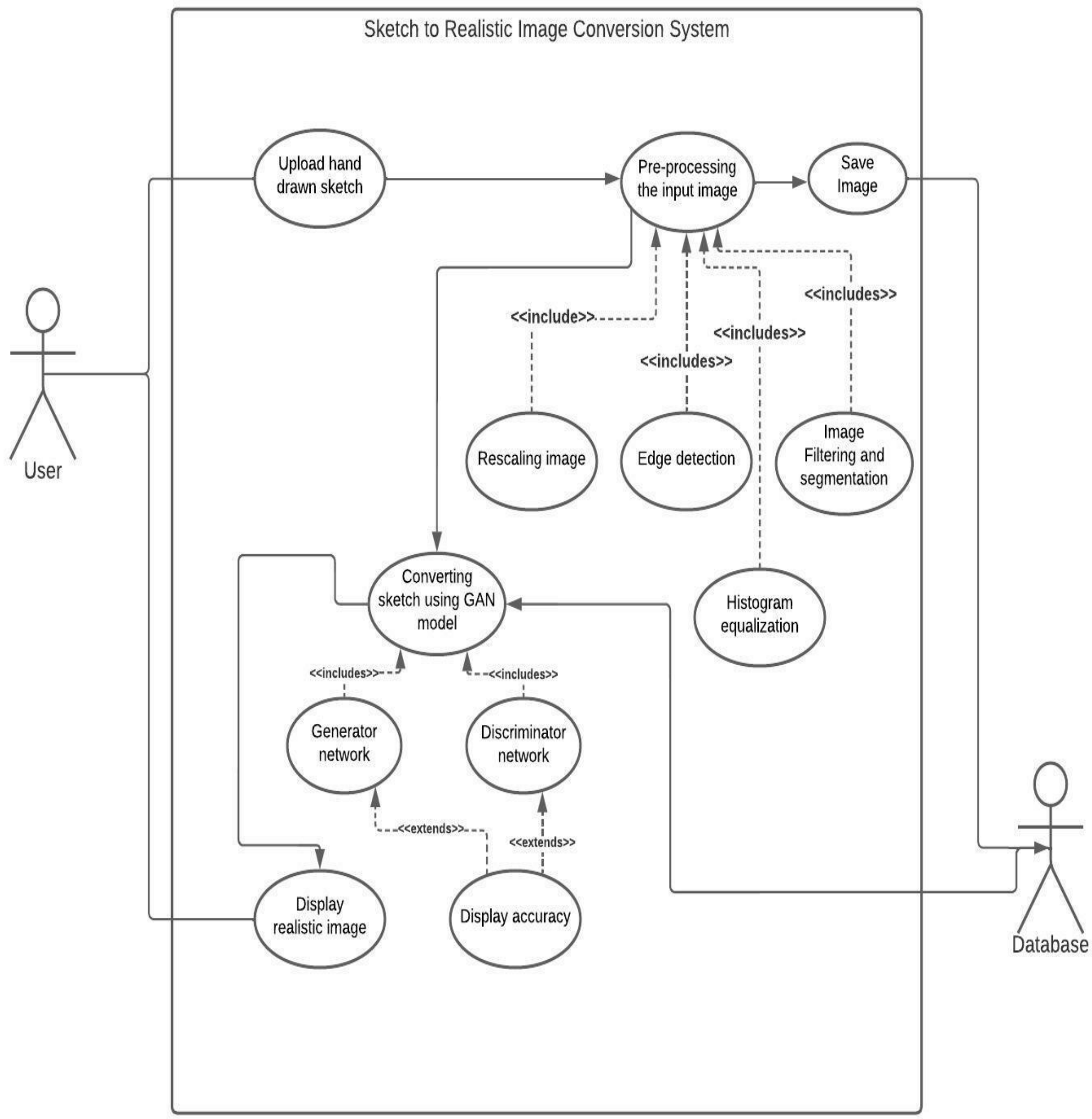

FIGURE 7: User Interface Diagrams of Hand Made Sketch to Realistic Image Model

4.4 Snapshots of Working Prototype

```
with tf.device('/GPU:0'):
  gan = Pix2Pix()
  gan.train(epochs=200, batch_size=5, sample_interval=5)
  gan.generator.save('model_generator.h5')
  gan.discriminator.save('model_discriminator.h5')
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:37: DeprecationWarning: `np.float` is a deprecated alias for the builtin `float`. To silence this warning, use `float` by
Deprecated in NumPy 1.20; for more details and guidance: https://numpy.org/devdocs/release/1.20.0-notes.html#deprecations
[Epoch 1/200] [Batch 1/4] [D loss: 12.814559, acc:  40%] [G loss: 85.778435] time: 0:00:03.431687
[Epoch 1/200] [Batch 2/4] [D loss: 9.329618, acc:  54%] [G loss: 70.719765] time: 0:00:04.456100
[Epoch 1/200] [Batch 3/4] [D loss: 11.362020, acc:  57%] [G loss: 63.014442] time: 0:00:09.302385
[Epoch 2/200] [Batch 1/4] [D loss: 1.964193, acc:  56%] [G loss: 48.847244] time: 0:00:10.063473
[Epoch 2/200] [Batch 2/4] [D loss: 1.441380, acc:  57%] [G loss: 47.068848] time: 0:00:11.039491
[Epoch 2/200] [Batch 3/4] [D loss: 0.707411, acc:  49%] [G loss: 42.866516] time: 0:00:15.554719
[Epoch 3/200] [Batch 1/4] [D loss: 0.714883, acc:  52%] [G loss: 36.763931] time: 0:00:16.296909
[Epoch 3/200] [Batch 2/4] [D loss: 0.706408, acc:  48%] [G loss: 39.000229] time: 0:00:17.285514
[Epoch 3/200] [Batch 3/4] [D loss: 0.593672, acc:  52%] [G loss: 34.948330] time: 0:00:21.930698
[Epoch 4/200] [Batch 1/4] [D loss: 0.551797, acc:  54%] [G loss: 32.529053] time: 0:00:22.704102
[Epoch 4/200] [Batch 2/4] [D loss: 0.553585, acc:  50%] [G loss: 37.609779] time: 0:00:23.751699
[Epoch 4/200] [Batch 3/4] [D loss: 0.514041, acc:  53%] [G loss: 33.908741] time: 0:00:28.438854
[Epoch 5/200] [Batch 1/4] [D loss: 0.511638, acc:  53%] [G loss: 27.987665] time: 0:00:29.478684
saving...
[Epoch 5/200] [Batch 2/4] [D loss: 0.489931, acc:  46%] [G loss: 31.772705] time: 0:00:32.673397
saving...
[Epoch 5/200] [Batch 3/4] [D loss: 0.417579, acc:  51%] [G loss: 29.633139] time: 0:00:38.997877
saving...
```

FIGURE 8: Coding Implementation of model

```
import os
import datetime
import imageio
import skimage
import scipy
import numpy as np
import tensorflow as tf
import matplotlib.pyplot as plt
from glob import glob
from PIL import Image
from skimage.transform import resize
from IPython.display import Image
```

```
class DataLoader():
    def __init__(self, img_res=(128, 128)):
        self.img_res = img_res

    def load_batch(self, batch_size=1, is_testing=False):
        data_type = "concat_image_sketch" if not is_testing else "val"
        path = glob(os.path.join(os.getcwd(), '{}'.format(data_type), '*'))
        self.n_batches = int(len(path) / batch_size)

        for i in range(self.n_batches-1):
            batch = path[i*batch_size:(i+1)*batch_size]
            imgs_A, imgs_B = [],[]
            for img in batch:
                img = self.imread(img)
                h, w, _ = img.shape
                half_w = w//2
                img_A = img[:, :half_w, :]
                img_B = img[:, half_w:, :]

                img_A = resize(img_A, self.img_res)
                img_B = resize(img_B, self.img_res)

                if not is_testing and np.random.random() > 0.5:
                        img_A = np.fliplr(img_A)
                        img_B = np.fliplr(img_B)
```

FIGURE 9: Coding Implementation of model

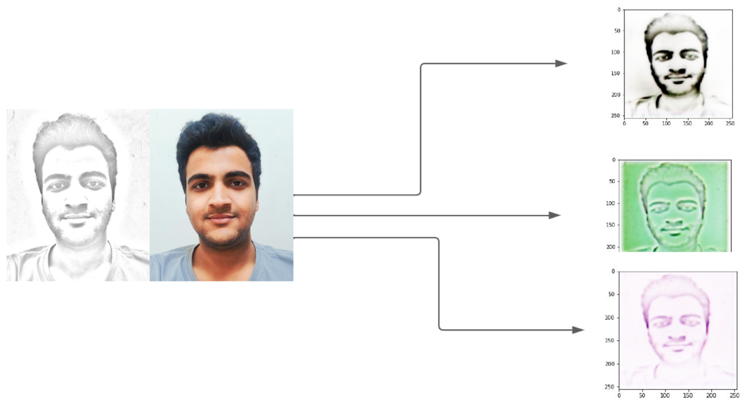

FIGURE 10: Output generated from our model

5.1 Work Accomplished (Discussion w.r.t the approved objectives)

In this work, we present a method that enables the user creation of customized generative models, by leveraging off-the-shelf pre-trained models and cross-domain training. We train multiple generator and discriminator models using one dataset. We've also used Ensemble techniques to combine multiple models that aim at improving the accuracy of results in models. After that we test our model on real time sketches which results in obtaining realistic images from our generator and discriminator model.

5.2 Conclusion:

A generative adversarial network (GAN) is trained to learn the joint distribution and capture the inherent correspondence between a sketch and its corresponding image, thus bypassing the cross-domain learning issues. This approach encodes the "corrupted" joint image into the closest "uncorrupted" joint image in the latent space, which can be used to predict and hence generate the output image part of the joint image. The model demonstrates that sketch completion is useful in various sketch-based applications.

5.3 Environmental Benefits (Economic/ Social)

Our project will help in finding witnesses of a crime scene, and create sketch drawings that are used by police to identify and apprehend criminal suspects. Also our project is low cost as it requires our own machine learning models which are used to make sketches. So our project is economically productive and cost-efficient.

5.4 Future Work Plan

To advance the system in the future, we would like to add a few components. At present our model makes sketch drawings to identify suspects. Further we will add more large datasets where our project will be able to make more accurate sketches.

REFERENCES

[1] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Advances in Neural Information Processing Systems, 2014.

[2] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. In International Conference on Learning Representations (ICLR), 2019.

[3] Yang Cao, Changhu Wang, Liqing Zhang, and Lei Zhang. Edgel index for large-scale sketch-based image search. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011.

[4] Mathias Eitz, Kristian Hildebrand, Tamy Boubekeur, andMarc Alexa. Sketch-based image retrieval: Benchmark and bag-of-features descriptors. IEEE transactions on visualization and computer graphics, 17(11):1624–1636, 2010.

[5] Abhinav Shrivastava, Tomasz Malisiewicz, Abhinav Gupta,and Alexei A Efros. Data-driven visual similarity for cross domain image matching.

[6] Takeo Igarashi, Satoshi Matsuoka, and Hidehiko Tanaka. Teddy: A sketching interface for 3d freeform design. In ACM SIGGRAPH, 1999.

[7] Tao Chen, Ming-Ming Cheng, Ping Tan, Ariel Shamir, and Shi-Min Hu. Sketch2photo: internet image montage. ACM Transactions on Graphics (TOG).

[8] Mathias Eitz, Ronald Richter, Kristian Hildebrand, TamyBoubekeur, and Marc Alexa. Photosketcher: interactive sketch-based image synthesis. IEEE Computer Graphics and Applications.

[9] Patsorn Sangkloy, Jingwan Lu, Chen Fang, Fisher Yu, and James Hays. Scribbler: Controlling deep image synthesis with sketch and color. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[10] Wengling Chen and James Hays. Sketchy GAN: Towards diverse and realistic sketch to image synthesis. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.

[11] Chengying Gao, Qi Liu, Qi Xu, Limin Wang, Jianzhuang Liu, and Changqing Zou. Sketchycoco: Image generation from freehand scene sketches. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[12] Tiziano Portenier, Qiyang Hu, Attila Szab´o, Siavash Arjomand Bigdeli, Paolo Favaro, and Matthias Zwicker. Faceshop: Deep sketch-based face image editing. ACM Transactions on Graphics (TOG).

[13] Mathias Eitz, James Hays, and Marc Alexa. How do humans sketch objects? ACM Transactions on Graphics (TOG).

[14] David Ha and Douglas Eck. A neural representation of sketch drawings. In International Conference on Learning Representations (ICLR), 2018.