# Mini Project Title

Submitted in partial fulfillment of the requirements of the degree

### BACHELOR OF ENGINEERING IN COMPUTER ENGINEERING

By

**Hriday Keswani 2003088**

**Viren Keswani 2003089**

**Vedant Jumle 2003074**

Supervisor

Ms.Manisha Dumbre



**Computer Engineering Department Thadomal Shahani Engineering College**

**Bandra(w), Mumbai - 400 050**

## University of Mumbai

**(AY 2021-22)**

# CERTIFICATE

This is to certify that the Mini Project entitled **“PC builder”** is a bonafide work of **Hriday Keswani (2003088), Viren Keswani (2003089), Atharva Kadam (2003075)** submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of **“Bachelor of Engineering”** in **“Computer Engineering”.**

#### (Prof.Manisha Dumbre)

Supervisor

#### (Prof. Tanuja Sarode) (Prof. G.T. Thampi)

Head of Department Principal

# Mini Project Approval

This Mini Project entitled “PC builder**”** by **Hriday Keswani (2003088), Viren Keswani (2003089), Atharva Kadam (2003075)** is approved for the degree of **Bachelor of Engineering** in **Computer Engineering.**

**Examiners**

**1………………………………………**

(Internal Examiner Name & Sign)

#### 2…………………………………………

(External Examiner name & Sign)

Date:

Place:

# Content

#### Abstract ii

#### Acknowledgments iii

#### List of Abbreviations iv

#### List of Figures v

#### fig(a) – Website Arhitecture

#### fig(b) – Working of the algorithm

#### fig(c) – Homepage

#### fig(d) – Choosing components

#### fig(e) – Output

#### fig(f) – Accessing documentation

#### fig(g) – Working Configurations of the build

#### List of Tables vi

#### List of Symbols vii

#### Introduction 1

* 1. Introduction
  2. Motivation
  3. Problem Statement & Objectives
  4. Organization of the Report

1. **Literature Survey 4**
   1. Survey of Existing System
   2. Limitation Existing system or research gap
   3. Mini Project Contribution

#### Proposed System (eg New Approach of Data Summarization) 5

* 1. Introduction
  2. Architecture/ Framework
  3. Algorithm and Process Design
  4. Details of Hardware & Software
  5. Experiment and Results
  6. Conclusion and Future work.

**References 32**

# Acknowledgement

We would like to express our gratitude and thanks to **Prof. Manisha Dumbre** for her valuable guidance and help. We are indebted for her guidance and constant supervision as well as for providing necessary information regarding the project. We would like to express our greatest appreciation to our principal **Dr. G.T. Thampi** and head of the department **Dr. Tanuja Sarode** for their encouragement and tremendous support. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of the project.

Hriday Keswani

Viren Keswani

Vedant Jumle

# Chapter 1

### 1.1 Introduction

Generative modeling is an unsupervised learning task in machine learning that involves automatically discovering and learning the regularities or patterns in input data in such a way that the model can be used to generate or output new examples that plausibly could have been drawn from the original dataset.

### 1.2 Motivation

The motivations behind training GANs may not necessarily be the generator or the discriminator per se: the representations embodied by either of the pair of networks can be used in a variety of subsequent tasks.

You can use GANs to:

• Generate simulated training data and simulated training environments

• Fill in missing data

• Train a classifier with semi-supervised learning (where the classifier learns from both labelled and unlabelled data… and with GANs, also learns from completely imaginary data)

• Do supervised learning where the supervision signal says that any one of multiple correct answers are acceptable, instead of just having one specific answer you request for each training example

• Replace expensive simulations with statistical generation

• Sample from the posterior distribution of a generative model

### 1.3 Problem Statement & Objectives

Problem statement:

To make a website that helps people learn how to build PCs and search for cost effective builds.

Objectives:

- To improve technical, know how on the consumer’s end.

- The project aims to provide cost effective and power effective builds to prevent the consumer from overspending

- To encourage independent decision making on the consumer end so that third parties cannot misinform them and overcharge them.

### 1.4 Organization of the Report

This report consists of three chapters. The first chapter deals with introduction of the topic, problem statement, motivation behind the topic and objectives. The second chapter is the Literature Survey. It includes all the research work done related to this topic. All information related to study of existing systems as well as learning of new tools is mentioned in this chapter. The third chapter is about the proposed system which is used in this project. The block diagram, techniques used, hardware and software used screenshots of the project are presented in this chapter. All the documents related to development of this project are mentioned in References

# Chapter 2

## Literature Survey

### 2.1 Survey of Existing System

GAN already exists and are used in various forms:

-DLSS for games

-Deep Fakes

-Filters in various apps

### 2.2 Limitation of existing system

The theory behind our project is only present in the form of literature only. There weren’t a lot of end-to-end system that gave users an UI to interface with the models that are built within this system.

### 2.3 Mini Project Contribution

Our project is an implementation of the original research papers for NST, DCGAN, and SRGAN. NST can be used to style different images with a particular styling image, DCGAN can generate random realistic faces that can unique and never seen before, and SRGAN can be used to upscale images without loss in details. Essentially, this project showcases the new advancements in AI and ML.

# Chapter 3

## Proposed System

### Introduction

The programming language and tools used are:

- Flask

- TensorFlow

- HTML

- CSS

- JavaScript

- Python

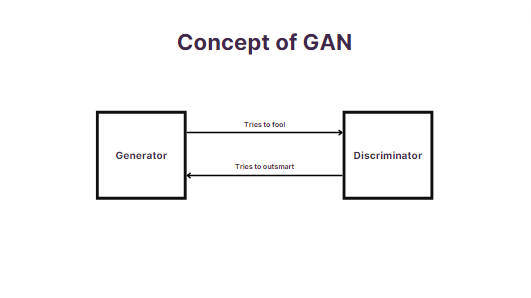
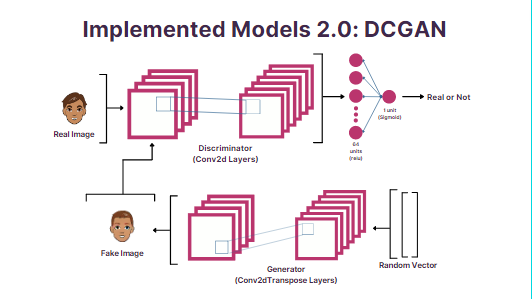
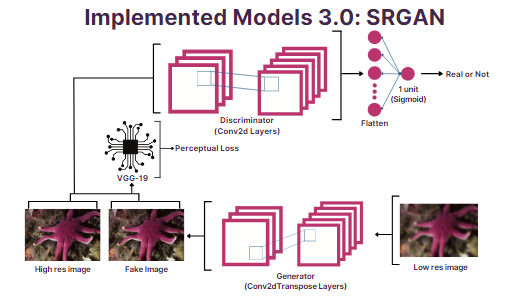
Datasets Used:

- [Google Cartoon set](https://google.github.io/cartoonset/download.html)

- [Div2k dataset](https://data.vision.ee.ethz.ch/cvl/DIV2K/)

### Architecture

### 

### Algorithm and Process Design

1. Formulating the Problem statement:

After much thought and research, we found that GANs are a very interesting topic in ML. These being the new form of image generation, we got curious and finalized to make SRGAN as the most complex sub-topic of our project

1. Understanding the framework and requirements:

We had a relatively good understanding of the concepts that were discussed in the original papers. We only had to figure out how to make custom models in TensorFlow.

1. Identifying tools/technology to be used:

* Python
* TensorFlow
* Flask
* JavaScript
* HTML/CSS

1. Finalizing the features to be included:

We finitized to the project to have three levels of increasingly complex ML models. First being Neural Style Transfer, second being DCGAN, third being SRGAN.

1. Development:

A lot of the time of development was spent in optimizing the machine learning model so that it does not bottle neck the system. Once the model was ready, we worked on integrating it into an API which then can be used in the webserver.

1. Testing:

Testing of ML model was fairly easy. Since it generated images, we can visually check the results and validate if the models were performing as expected, which they did.

### Details of Hardware & Software

The project was developed in the High-end configuration of:

8X2 GB DDR4 3200MHz RAM

AMD RYZEN 5 5600X

RTX 3080 OC 10GB VRAM

Windows 10/11

VS Code text editor

and it can run on any device capable of rendering a website

### Results

### 

### 

### 

### 

### Conclusion and Future Work

In future we can implement multi-topical generator for DCGAN, which can generate random images of different topic. We can also train the DCGAN model to take in required features for the image and make it generate images that align with the given set of features of the image. For SRGAN, we can improve upon it by implementing the Extended SRGAN model (ESRGAN).

**References**

L. Gatys, A. Ecker, and M. Bethge, “A neural algorithm of artistic style,” Journal of Vision, vol. 16, no. 12, p. 326, 2016.

A. Radford, L. Metz, and S. Chintala, Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks, Aug. 2015.

C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, “Photo-realistic single image super-resolution using a generative adversarial network,” 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.