RE-2022-113726-plag-report

by Research Experts - Turnitin Report

Submission date: 23-Apr-2023 02:25AM (UTC-0400)
Submission ID: 2072537787
File name: RE-2022-113726.docx (4.16M)
Word count: 5491
Character count: 30037

Chapter 1

Introduction

1.1 Motivation and Background

In less than ten years, the GANs technology has developed swiftly and considerably in the field of AI. The continual advancement of GAN technology now makes it possible to alter real photos or produce phoney images that are hardly distinguishable to the human eye. This can be used for a wide range of purposes in a number of different industries, including the development of new product designs and deep fakes.

Frontal view with a realistic photo In the field of face recognition, there are several uses for synthesis from a single face image. A facial image in an arbitrary pose is frontalized using this procedure to create an image with a frontal stance.Generative models can employ face frontalization to solve the issue of model deterioration due to the variable in head attitude in face identification thanks to the great development of generative adversarial networks (GAN).

1.2 Objective

- To use GAN to develop a system that creates the frontal view of a face from random or angular facial poses

- To use the generated facial images for applications such as Criminal Investigation, Plastic surgery preview, Portrait drawing, Increase size of datasets, etc.

1.3 Outline

This dissertation report consists of the following chapters. The contents of the chapters are as follows,

Chapter 2: Literature survey

The domain of Generative Adversarial Networks was surveyed. Several concepts and models related to image generation and manipulation reviewed, their limitations analyzed, a problem statement formulated and the potential scope was reviewed.

Chapter 3: Proposed System

Architecture and working of Two pathway GAN is proposed detailing every block of its work-

Chapter 4: Implementation Plan

The descriptive algorithm for the proposed model is discussed along with hardware system specifications and software details of the stepwise implementation.

Chapter 5: Results and Discussion

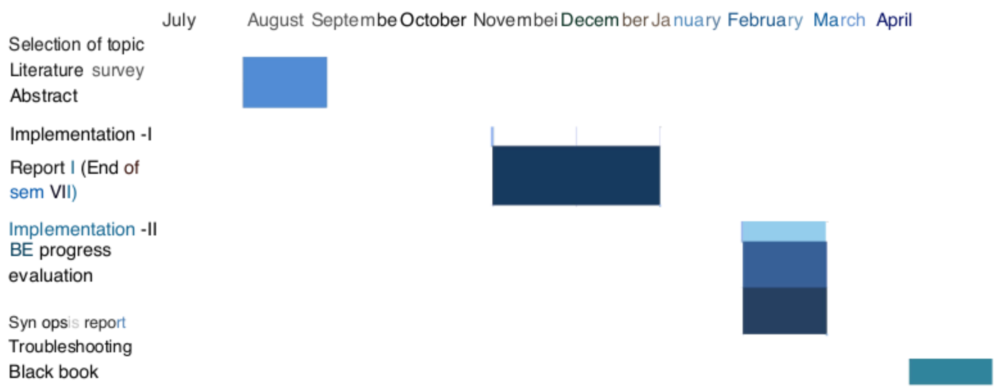
The training and testing details of the model such as number of epochs, batch size etc are described. Preliminary results of generated output from TP GAN along with comparision and analysis

Chapter 6: Conclusion and Future scope

Generated output and the and final results of the projects along with discussion of further implementation ideas

1.4 Time Plan

Table 1.1: Time Plan of the project



Chapter 2

Literature Review

In 2014, computer scientist Ian Goodfellow invented GANs, sometimes referred to as Generative Adversarial Networks. Using competing neural networks dubbed Generator and Discriminator, GAN is a method of generative modelling that creates a new collection of data from training data that is inseparable from it. The discriminator is the detective whose primary responsibility is to distinguish between input data that is authentic and phoney, while the generator is comparable to a forger who creates false data. In GANs, discriminative models perform the function of a classifier. Generative models, on the other hand, create realistic pictures from random samples like random noise.

The technology of GANs has advanced quickly and significantly in less than ten years, having a significant influence on AI. The ability to create false pictures or edit actual images so that they are almost imperceptible to the human eye has become achievable because to the ongoing development of GAN technology. We discuss various cutting-edge models in this work to examine GANs and their use in face synthesis, including GANimation, StyleCLIP, JOJOGAN, InterFaceGAN, and GANs N Roses.

2.1 Survey of Existing systems

2.1.1 ‘2D faciallandmark localization method for multi-view face synthesis image using a two-pathway generative adversarial network approach" , Mahmood H.B. Alhlffee, Yea-Shuan Huang, Yi-An Chen, February 16, 2022 [1]

The robust pose-invariant facial recognition method (LFM) presented in this research attempts to enhance the image resolution quality of the generated frontal faces under a variety of facial positions. We therefore establish reliable feature extractors that pick useful features that simplify operational workflow and enhance the current TP-GAN generative global pathway with a well-constructed 2D face landmark localization to cooperate with the local pathway structure in a landmark sharing manner to incorporate empirical face pose into the learning process to improve the encoder-decoder global pathway structure for better representation of facial image features. On Multi-PIE and FEI datasets, we validate the effectiveness of our proposed method. The results of the quantitative and qualitative tests show that our suggested technique significantly exceeds the output of the TP-GAN and also creates high quality perceptual pictures in extreme positions..

2.1.2 ‘Beyond Face Rotation: Global and Local Perception GAN for Photo-realistic and Identity Preserving Frontal View Synthesis”, Rui Huang; Shu Zhang; Tianyu Li; Ran He, IEEE 2017 [2]

This research proposes a T\vo-Pathway Generative Adversarial Network (TP-GAN) that concurrently observes global structures and local details to synthesise photorealistic frontal perspectives. Four iconic localised patch networks are proposed in addition to the well-known global encoder-decoder network to address local textures. In order to restrict this ill-posed problem, we offer a combination of adversarial loss, symmetry loss, and identity preserving loss, with the exception of the novel architecture. The combined loss function makes use of both the frontal face distribution and previously trained discriminative deep face models to support an identity-preserving inference of frontal views from profiles. Unlike existing deep learning systems, which mostly rely on intermediate estimated attributes for recognition, our method directly employs the identity-preserving picture formed for subsequent tasks like face recognition and attribution. The results of the studies demonstrate that our method not only yields pleasing perceptual results but also outperforms state-of-the-art results for big pose face identification.

2.1.3 'Face Recognition Based On Frontalization Of Multiple Poses Using G-GAN and DWT", IJCRT International Journal of Creative Research Thoughts, May 2021 [3]

Face authentication has various concerns and is utilised in many applications, making it one of the research community's more recent frontiers. The difficulty of face recognition is exacerbated by significant side angles in profile photographs because frontal and profile images contain quite different traits that help identify people. This is the main cause of the decreased performance of several cutting-edge facial recognition systems for human identification. Using global generative adversarial networks (G-GAN), we suggest frontal face synthesis from a profile face image before facial recognition in this research study. Good identity retention and photorealistic frontal face images are produced by employing G-GAN in conjunction with special error loss functions during training.On frontal pictures obtained from profile photos and the frontal image saved in the database, the Discrete Wavelet Transform (DWT) is utilised to extract features. The low-frequency band's compressed low dimensional significant traits are taken into account for face identification. For face recognition, the Euclidean Distance (ED) between the frontal pictures of the frontal faces generated by G-GAN and the frontal images from the ground truth database is computed. The experimental results, which were conducted utilising publicly accessible datasets as Multi PIE, Bosporus, head pose-invariant, and Indian female, are based on qualitative and quantitative analysis and produced promising outcomes. The experimental findings of the proposed system in comparison to other existing methodologies demonstrate the superiority of our approach.

2.1.4 'Towards Large-Pose Face Frontalization in the Wild", IEEEXPLORE, Aug,17 2017 [4]

Despite recent developments in deep learning-based face recognition, a sharp decline in accuracy is seen for significant pose differences in unrestricted situations. One option is position invariant feature learning, however it needs expensive large-scale labelled data and carefully crafted feature learning algorithms. Here, we concentrate on face fronting in various head positions, including extreme profile views, in nature. To produce face images from neutral head positions, we suggest a new deep 3D morphing model (3DMM) processed face front generative adversarial network (FF-GAN). Our approach is distinct from both conventional GAN-based modelling and 3DMM-based modelling. The GAN framework incorporates 3DMM to offer shape and appearance requirements for quick convergence with less training data while providing comprehensive training. The 3DMM-processed GAN uses identity loss to recover high-frequency information and a novel masked symmetry loss to maintain visual quality under occlusions in addition to the separator and generator losses. The benefits of our frontalization method on wild face datasets are repeatedly demonstrated in experiments on face recognition, landmark localization, and 3D reconstruction.

2.1.5 'Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network", IEEE COMPUTATIONAL INTELLIGENCE, May 2017[5]

The generative adversarial network (GAN) for super-resolution of images, or SRGAN, is presented in this paper. For four upscaling factors, it is the first system capable of predicting photorealistic natural images. the first extremely deep ResNet architecture was described, employing the idea of GANs to create a perceptual loss function for photo-realistic SISR. SRGAN reconstructions are, by a significant margin, more photorealistic for large upscaling factors (4), it was determined.

2.1.6 'Loss Functions of Generative Adversarial Networks (GANs)", IEEE ON EMERGING TOPICS IN COMPUTATIONAL INTELLIGENCE, AUGUST 2020 [6]

In this paper we evaluate the loss functions utilised in GANs and weigh their advantages and disadvantages. First, the foundational theory and training process of GANs are introduced. The loss functions employed by GANs are then compiled, comprising both the goal functions and the application-oriented GANs. Thirdly, there is a discussion of the experiments and analysis of representative loss functions. Finally, a number of recommendations are provided on how to select suitable loss functions for a particular assignment. Generative Adversarial Networks (GANs) are currently quickly emerging as a significant and promising study area in artificial intelligence. Loss functions are used to measure the discrepancies between model-generated samples and real samples and force the model to train towards the target, improving the modelling capabilities of GANs. In this study, we evaluate the loss functions utilised in GANs and weigh their advantages and disadvantages.

2.2 Limitations

1. Occlusion: If the input face image is partially occluded or if there are missing facial features due to poor lighting conditions, the TP-GAN may not be able to effectively generate the frontal view.

2. Large pose variations: TP-GAN is most effective when there is a moderate pose variation in the input image. Large pose variations can result in distorted or unnatural-looking frontalized images.

3. Limited training data: TP-GAN requires large amounts of training data to effectively learn the mapping between non-frontal and frontal views. If the training data is limited, biased or of poor quality, the generated frontalized images may not be of high quality.

4. Sensitivity to hyperparameters: As with other GANs, the performance of TP-GAN is highly sensitive to hyperparameters such as learning rate, batch size, and the number of epochs. Finding the optimal hyperparameters can be time-consuming and require significant computational resources.

5. Limited applicability: TP-GAN frontalizes faces in a generic way, without taking into account specific facial characteristics or individual differences. This can limit its applicability in certain scenarios, such as facial recognition systems that require highly accurate and specific representations of individual faces.

   Despite these limitations, TP-GAN remains a powerful technique for face frontalization, and ongoing research aims to address some of these limitations to further improve its performance.

2.3 Problem Statement

To create a system that generates the frontal view of a face from arbitrary or non-functional face poses **using** GAN

2.4 Scope

This project is majorly used for generating frontal faces from side profile images. For future purposes, we can take inputs, as not only side images, but also images taken from different angles which will be useful in the practical world applications.
Some applications of Face Frontalization are:

1. Medical App1ications:- Here the GANs are used for various applications like surgery, face whitening, alignment of face architectures and also for medical face reconstruction of a person.

2. Crime and investigation:- Any investigation must include a thorough examination of the crime scene. CCTV footage not necessarily capture the full face of the criminals which makes them difficult to identify. Therefore, this frontilzation technique can be used to create the whole image of a person that would be beneficial in the identification of a criminal. In surveillance cameras, occasionally the proper image of a person is not caught.

Figure 2.1: Example of reconstruction of a section of face

6

3. Human face models:- A facial recognition system is a piece of technology that can compare a human face in a digital photo or video frame to a database of faces. Such a technology locates and measures face features from an image and is often used to authenticate individuals through ID verification services. Face ageing has been used to forecast a person's appearance in the future.

Chapter 3

Methodology

3.1 Proposed System

3.1.1 GAN (Generative Adversal Network)

The GAN is one of Ian Goodman's (2014) most intriguing research frameworks for deep generative models. The philosophy of the GAN framework can be understood as a non-cooperative, two-player game with the goal of improving the learning model. The discriminator (D) and the generator (GI are the two fundamental components of a GAN model. In order to confuse D, which is trying to distinguish between real generated images and fake ones, G produces a series of images that are as plausible as they possibly can be. The convergence is achieved by training them in various ways. The key difference between GANs and other generative models is that GANs produce complete images as opposed to individual pixels. the hyperparameters that led to the most successful results. The generator in a GAN architecture is made up of two dense layers and a dropout layer. The noise vectors are sampled, and they are fed into the generator networks using a normal distribution. Any supervised learning model can be used as the discriminator. For a variety of tasks, including image synthesis, image super-resolution, image-to-image translation, etc., GANs have been shown to be effective. To handle the most challenging unconstrained face image scenarios, including changes in stance, lighting, and expression, a number of efficient GAN models have been presented.

3.1.2 TP GAN

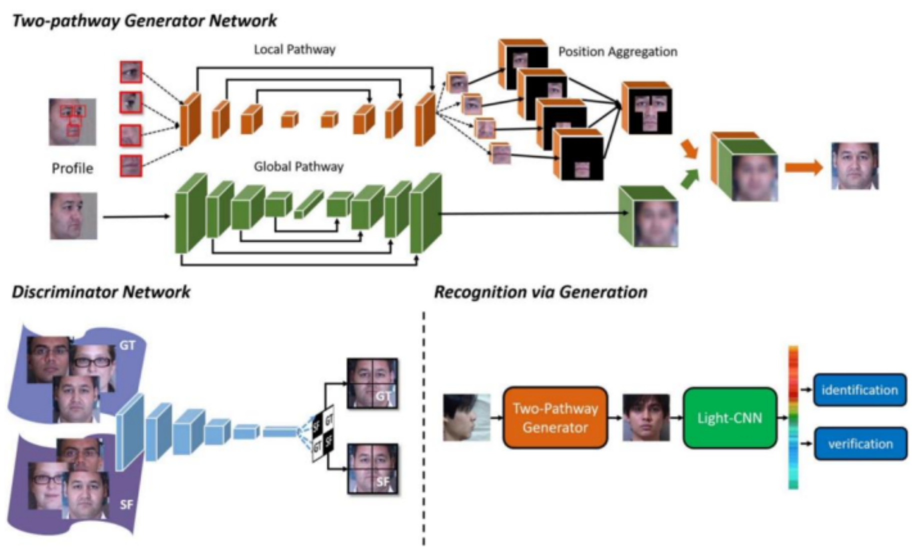


Figure 3.1: Architecture of Two Pathway GAN

The type of GAN we are using is TPGAN. Compared to any other gan TPGAN gives us more photo-realistic images. Since GANs contain a generator and a discriminator but TP-GAN which is a two path-way gans it has 2path-ways in the generator and a discriminator that is a local pathway and a global pathway. The local pathway is used to generate the specific facial parts that are the main features of an image such as 2 eyes nose and mouth. These local pathways are then merged and placed according to their facial location.

Then in the global pathway focus on features apart from eyes nose mouth such as skin color that is to generate the overall view of face images. Then both these pathways local and global are convoluted several times to generate the face image. A discriminator is used to distinguish between real values and the fake samples that are generated so SF is the synthesized front view the image that we got from the generator and the GT is the ground truth is the actual front view of the people so it is used to distinguish between the both.

In comparison to single-pathway GANs, it has been demonstrated that the two-pathway GAN architecture produces images of superior quality. Additionally, it gives the generator the ability to produce images in a variety of resolutions, which is advantageous for uses like image super-resolution.The TP-GAN's general structure.

Two-pathway generator network In a two-pathway generator network (TP-GAN), the generator network is the neural network responsible for generating synthetic data, typically images, that mimic the distribution of a given training dataset. The generator network in TP-GAN is responsible for creating realistic and high-quality images from random noise vectors, and its architecture is designed to balance between the generation of diverse images and the preservation of global image structure.

Two-pathway discriminator network The Two-Pathway Discriminator Network is a type of discriminator architecture used in Generative Adversarial Networks (GANs) to distinguish between real and fake images generated by the generator network. The Two-Pathway Discriminator Network is an effective architecture for improving the quality of the generated images in GANs. By processing the input images at different resolutions, the discriminator network is better able to distinguish between real and fake images, resulting in better convergence and more realistic generated images.

Recognition via generation Recognition via generation is a technique used in Generative Adversarial Networks (GANs) to perform recognition tasks using the generator network by generating images that correspond to specific classes or attributes and using them to train a recognition network. It has the advantage of not requiring a large amount of labeled training data, as the generator network can be trained on unlabeled data and the generated images can be used for the recognition task.

Chapter 4

Implementation

4.1 Dataset

Multipie

We are using the Multipie dataset in this model. The Multi-PIE (Pose, Illumination, and Expression) dataset is a large-scale face dataset consisting of over 750,000 images of 337 subjects, each with 15 different viewpoints, 20 different illumination conditions, and 6 different expressions. The dataset was created by Carnegie Mellon University and the University of Southern California, and it is widely used for research in face recognition, pose estimation, and other related fields. In total, the database contains more than 305 GB of face data.

This dataset was designed to capture the variability of facial appearance due to changes in pose, illumination, and expression, which are common challenges in face recognition. The images were captured using a high-resolution digital camera and a controlled lighting setup, and the subjects were instructed to maintain a neutral expression while their faces were captured from different viewpoints and under different illumination conditions.

The Multi-PIE dataset is annotated with ground-truth information about the subject identity, pose, illumination, and expression, making it a valuable resource for training and evaluating face recognition algorithms. It has been used in a wide range of research projects, including face recognition, pose estimation, expression recognition, and 3D face reconstruction. It has also been used as a benchmark dataset for evaluating the performance of various face recognition algorithms, and many state-of-the-art methods have been trained and evaluated on this dataset.

4.2 Libraries

i. os : This module provides a portable way of using operating system dependent functionality.

2. keras: Keras is a deep learning API written in Python, running on top of the machine learning platform TensorFlow .

3. torch: A library that supports automatic differentiation, a dynamic computation graph, and the construction and **training** of neural networks.

4. torchvision: A package that makes it possible to use PyTorch to access well-liked datasets, model architectures, and picture transformations for computer vision problems.

5. tensorflow: TensorFlow is an open source software library for high performance numerical computation. Its flexible architecture allows easy deployment of computation across a variety of platforms (CPUs, GPUs, TPUs), and from desktops to clusters of servers to mobile and edge devices.

6. numpy: library that provides support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays

7. cv2: OpenCV library for computer vision and image processing tasks, providing tools and functions for image and video processing, object detection, camera calibration, and more.

8. face alignment: A face alignment library is a software library that provides tools and functions for automatically aligning facial landmarks or key points in images or videos.

9. PIL: A library that enhances support for a wide range of image file types in terms of opening, modifying, and saving.

10. numba: Python library for just-in-time (JIT) compiling of numerical computations, optimizing performance by generating optimized machine code for CPUs or GPUs. Ideal for accelerating numerical computations in areas such as data science, machine learning, and scientific computing.

11. random: A module that offers tools for creating pseudo-random integers using different distributions.

12. re: Python module for regular expressions, used for pattern matching and text manipulation tasks such as search, replace, and splitting strings based on patterns.

13. multiprocessing:

14. time: Python module for working with time-related operations, including measuring time, formatting and parsing time, and scheduling tasks. Commonly used for tasks such as timing code execution and working with timestamps.

15. collection: a module that offers specialised container datatypes in addition to the built-in containers like dictionaries, lists, and tuples.

16. threading: The "threading" library is a built-in module in Python that provides tools for working with threads, which are lightweight and concurrent units of a program's execution.

17. skimage: The "skimage" library, short for "scikit-image," is a Python image processing library that provides a wide range of tools and functions for working with digital images.

4.3 Training details

To provide a consistent and repeatable working environment, we used Python 3.6 as our main programming language and built a virtual environment using conda on the GPU. This made it possible for us to install and maintain particular versions of necessary Python packages, such as deep learning modules like PyTorch. In order to train our GAN model effectively, we used the NVIDIA RTX 2060 Super GPU, which offered high-performance processing capabilities. Additionally, this made it possible to simultaneously experiment with other hyperparameters, such learning rates and batch sizes, to enhance model performance. We found the optimum collection of hyperparameters that produced the greatest results after numerous iterations of training with various hyperparameter combinations.

Training Parameters				
Learning Rate (G/D)	Batch Size	Steps per epochs	Epoch trained	Time taken(Hrs)
0.0001	4	100	211	20
0.0001	4	100	100	10
0.0001	2	200	3204	168

Table 4.1: Training Parameters

4.4 Working

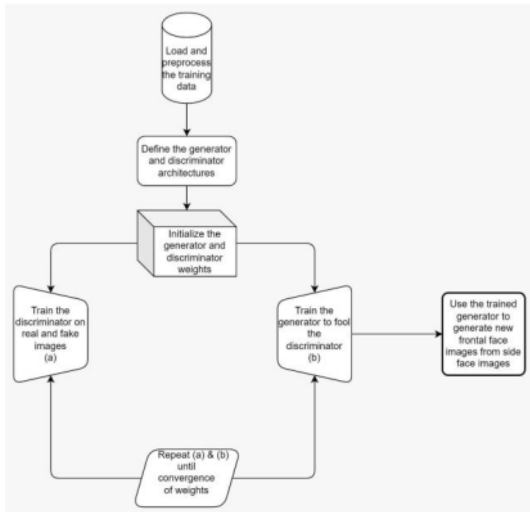


Figure 4.1: Block diagram of working model

Step 1: Dataset; The dataset is at the centre of any machine learning effort. JPEG format should be used for the photos. To create synthesized images that are as realistic as possible, the image dimensions should have a 1:1 ratio.

Step 2: Setting up GPU; First we downloaded the NVIDIA Driver with relevant cuda and CUDNN version according to our hardware compatibility, in our case CUDA version 11.5. After downloading the drivers created environment in Anaconda IDE.

Step 3: Face Identification We run face recognition on MultiPIE with two distinct settings to quantitatively show how well our algorithm preserves identity.The trials are first carried out by usingLight-CNN to extract deep features.

Step4: GANs; A generator (Encoder+Decoder) and a discriminator are the two networks. The Generator's task is to create synthetic images, while the Discriminator's job is to distinguish between authentic photos and those that were created artificially by the Generator.

Step 5: The model is tested and trained. Loss function in training is the binary cross entropy. A neural network is trained when its weights are updated in a way that minimises the loss function.For the discriminator, whose task it is to categorise the images as either real or fraudulent, we require the binary cross entropy loss.Testing: We place the appropriate photographs in a directory named test set that has the same structure as the training set above even though these subjects are absent from our training set.The pre-trained generator network, which we saved during training, was fed the test images as input and produced the results.

4.5 System specifications

- Operating system: 64-bit Microsoft windows 10 • RAM: At least 32 GB of RAM is required for running the software and training models

• Graphics Processing unit: The following NVIDIA GPUs were used for training the GANs: 1. NVIDIA RTX 2060 Super

2. NVIDIA RTX Geforce **3050**

• CUDA Toolkit:The CUDA Toolkit is a software development kit (SDK) developed by Nvidia for building and optimizing applications that run on Nvidia GPUs. It includes a set of libraries, tools, and APIs for developing parallel applications using CUDA, Nvidia's parallel computing platform.

• CPU Architecture:x86 64 architecture with support for Windows Hypervisor. A 2nd generation intel core processor or newer, or an AMD CPU is recommended

4.6 Software Details

[11 Anaconda (version 2019):The Python programming language's Anaconda open-source distribution is well known and optimised for data science and machine learning work1oads.The package manager included in Anaconda, called Anaconda Navigator, makes it easier to install and manage packages for data science and machine learning operations. Popular data science libraries like NumPy, Pandas, Scikit-Learn, and Matplotlib are also included.Python 3.7, which was the most recent stable version of Python at the time of its release, is included in Anaconda's 2019 release. Jupiter Notebook, Spyder, RStudio, and a number of additional data science tools and libraries are also included.

[2] Vscode:-Microsoft created the free and open-source Visual Studio Code (VS Code) source code editor for Windows, Linux, and macOS. Developers can take advantage of a wide range of capabilities, such as syntax highlighting, code completion, debugging, Git integration, and many others.

[31 Cuda:-Nvidia created the parallel computing platform and programming style known as CUDA (Compute Unified Device Architecture) for usage with its graphics processing units (GPUs). Using CUDA, programmers can take advantage of the GPUs' parallel processing capabilities to speed up computationally demanding activities like data processing, scientific simulations, and machine learning.

[4] Cudnn:-Nvidia created a collection of deep neural network-optimized primitives called cuDNN (CUDA Deep Neural Network). It is constructed on top of CUDA and intended to operate without any hiccups with well-known deep learning frameworks like TensorFlow, PyTorch, and Caffe.

[5] Pytorch:-For a variety of applications, including computer vision, natural language processing, and reinforcement learning, PyTorch is a strong and adaptable machine learning framework that is extensively used in business and academia.

Chapter 5

Results & Discussion

We have successfully trained the model for 3200 epochs. In the first attempt we managed to train for 73 epochs, and every attempt epochs kept increasing. An epoch means training the neural network with all the training data for one cycle. In an epoch, we use all of the data exactly once. A forward pass and a backward pass together are counted as one pass: An epoch is made up of one or more batches, where we use a part of the dataset to train the neural network.

With every attempt to train the model, we kept getting output for a higher number of epoch and the last epoch we managed to successfully achieve is 2392. Here at epoch 2390 the generator loss has decreased from 0.402 to lowest of 0.233 and the discriminator loss is 0.797 constant. We can see that the facial features are much visible and sharper in the final output compared to previous epochs.

Figure 5.1: Input image in arbitrary pose

Figure 5.2: Output for 73 epoch

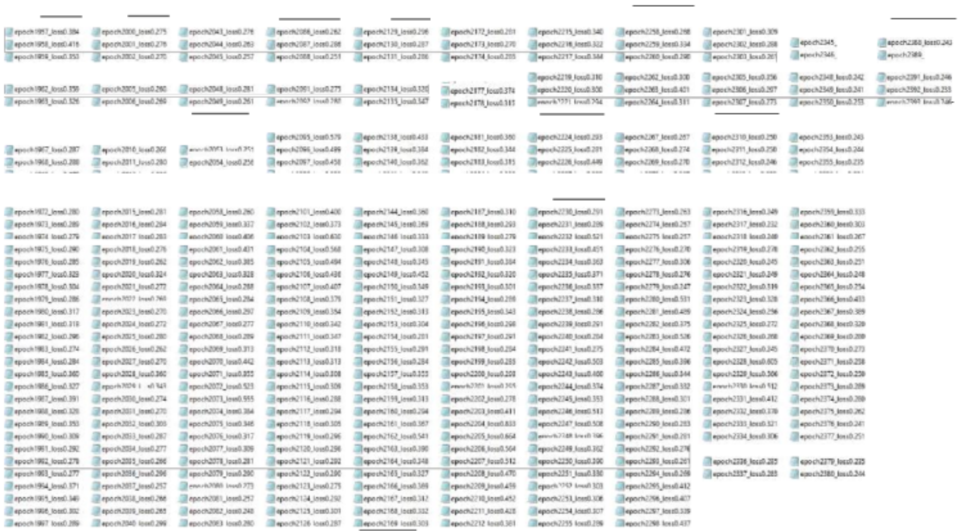


Figure 5.3: Epoch files

Figure 5.4: Output for 2392 epoch

We have plotted the performance graphs for generator loss(g-loss) and discriminator loss(d-loss) where x axis is the epoch and y axis is the loss, these graphs are showing the performance for 20 epoch and thus we can say that as the model is trained for more epochs the loss will decrease and accuracy will increase.

The TPGAN model which is pretrained on the Multipiedataset. The dataset has been used for pretraining the model to learn facial features, representations, or mappings that are specific to the variations present in the Multi-PIE dataset. Face frontalization for Intermediate training epochs for different input faces giving the following outputs.

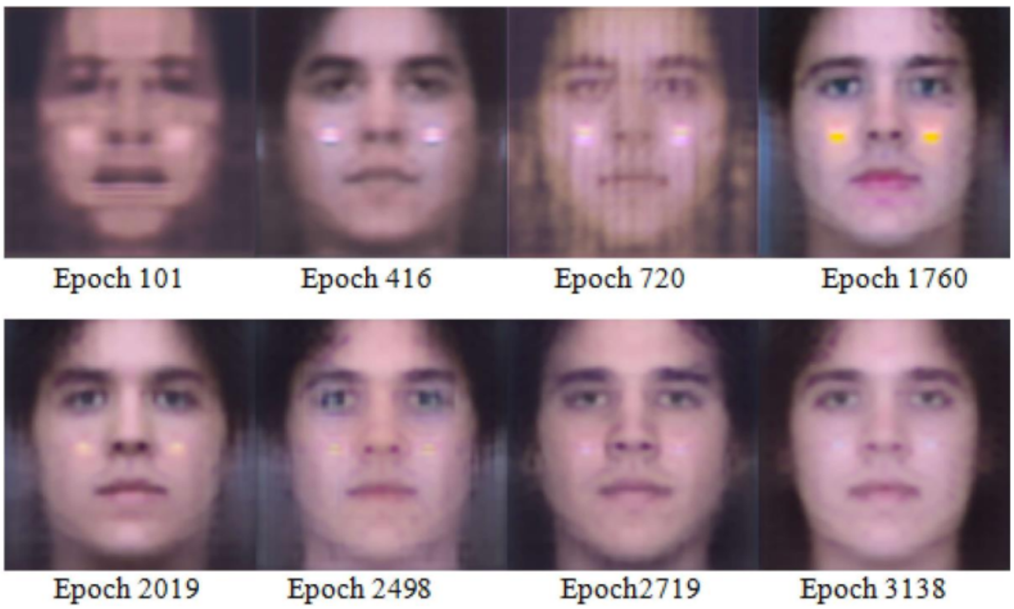


Figure 5.5: Output face 1

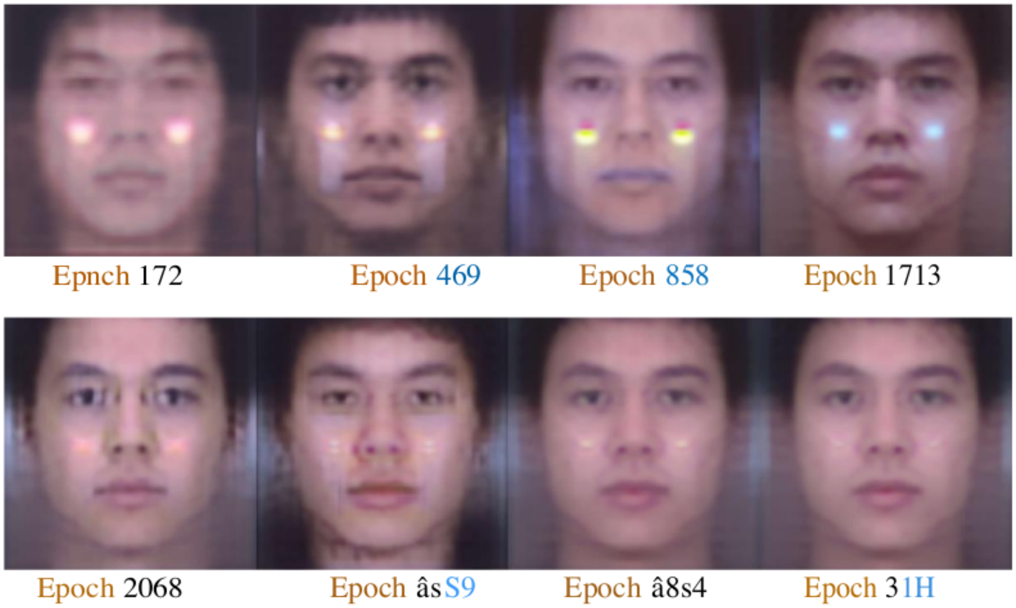


Figure 5.6: Output face 2

Figure 5.7: Output face 3

We have plotted the performance graphs for **generator loss(g-loss)** and **discriminator loss(d-loss)** where x axis is the epoch and y axis is the loss, these graphs are showing the performance for 20 epoch and thus we can say that as the model is trained for more epochs the loss will decrease and accuracy will increase. Batch 4 and Lr = 0.00010 Epoch 109

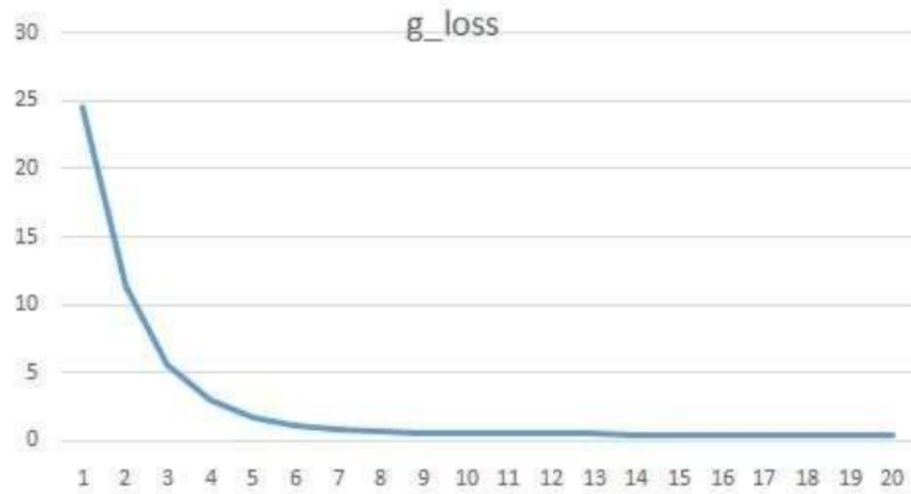
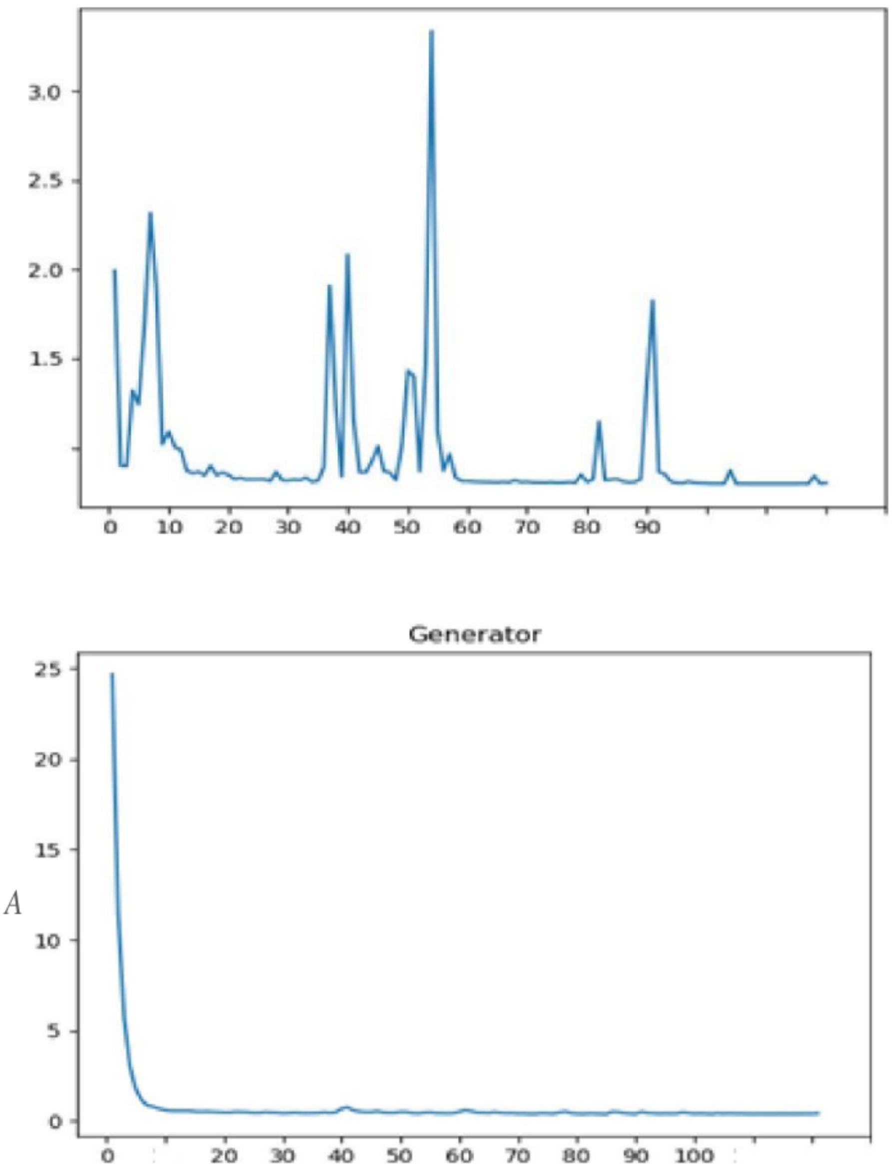


Figure 5.8: Generator loss

Figure 5.9: Discriminator loss

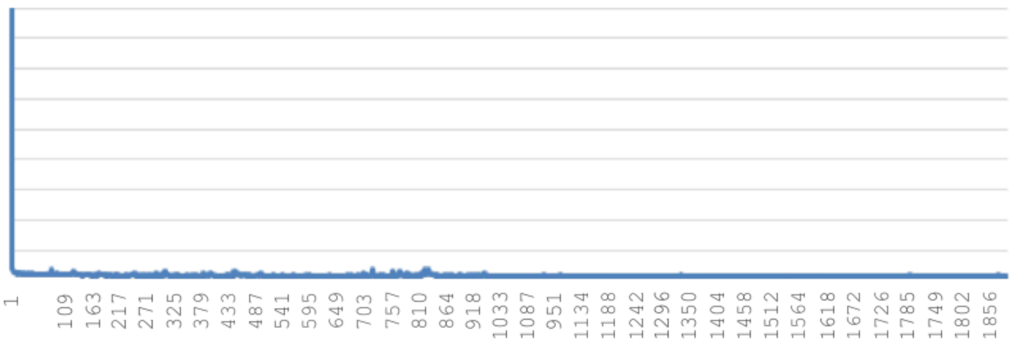
5.0.1 Comparison/Analysis

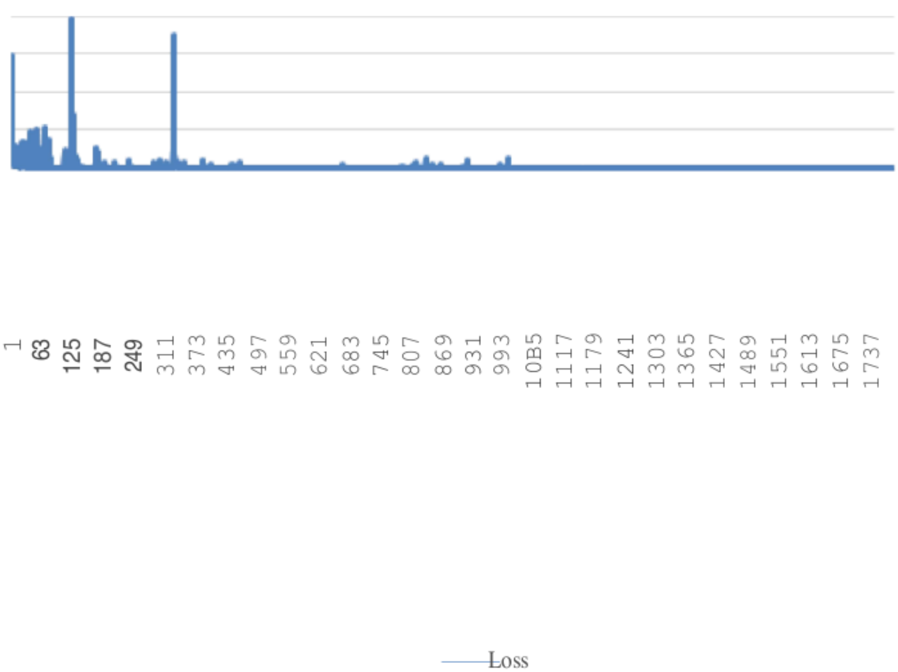
Based on its capacity to produce images while training on various hardware configurations, such as physical GPUs, utilising varying learning rates and batch sizes, TPGAN performance is assessed. A comparative examination of the model indicates that it had a promising training process and relatively successful results. Different hyperparameters, including batch sizes and learning rates, were tested in order to improve the model's performance. Batch-size 2 with Lr= 0.0001 was shown to be the most effective combination. Depending on the learning rate and batch size, an epoch's training time typically lasted between 5 and 6 minutes. Overall, the model showed good fidelity during training and successfully captured the characteristics listed in the captions. However, it is unable to frontalize images outside of the dataset because doing so would require extra epochs and a modification in the hyperparameters, which would cause the model's performance to suffer.



A

Figure 5.10: Graphs for 130 epochs


1
109
163
217
271
325
379
433
487
541
595
649
703
757
810
864
918
1033
1087
951
1134
1188
1242
1296
1350
1404
1458
1512
1564
1618
1672
1726
1785
1749
1802
1856


1
63
125
187
249
311
373
435
497
559
621
683
745
807
869
931
993
10B5
1117
1179
1241
1303
1365
1427
1489
1551
1613
1675
1737
Loss

5.0.2 Subjective analysis

A survey was conducted on 64 people, where they were asked their opinion on the comparison between the original image and the generated frontal image. They were asked to comment or give ratings in the range of 1 to 4, 1 being the lowest and 4 being the highest. Below images were presented to the audience and the results obtained are shown in the graph below.

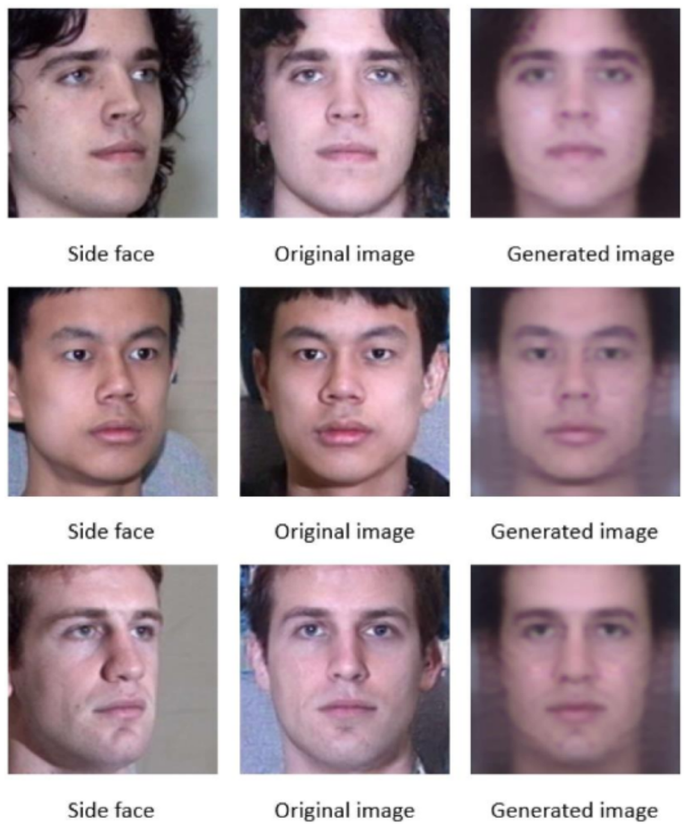


Figure 5.13: Images from the surevy

Figure 5.14: Images from the surevy

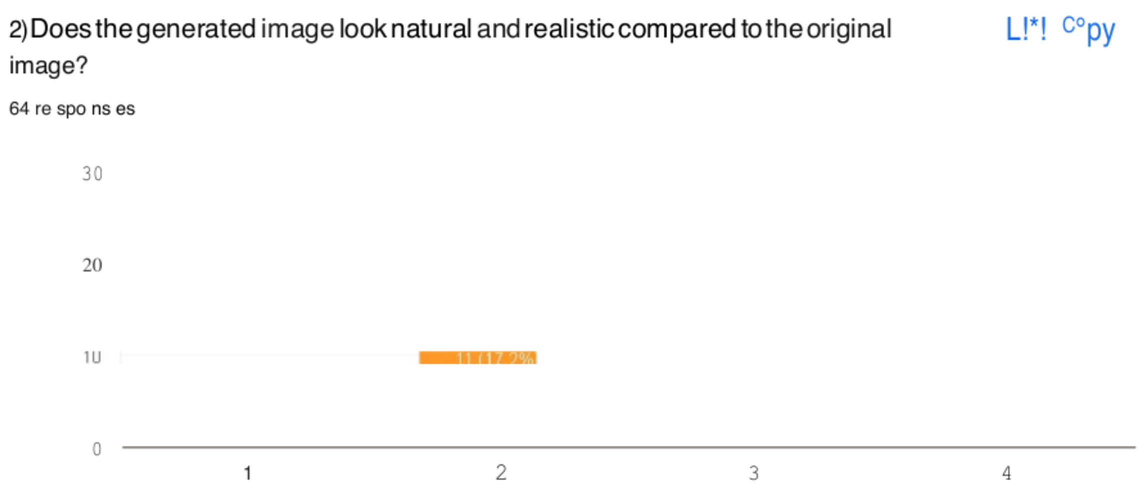


Figure 5.15: Images from the surevy

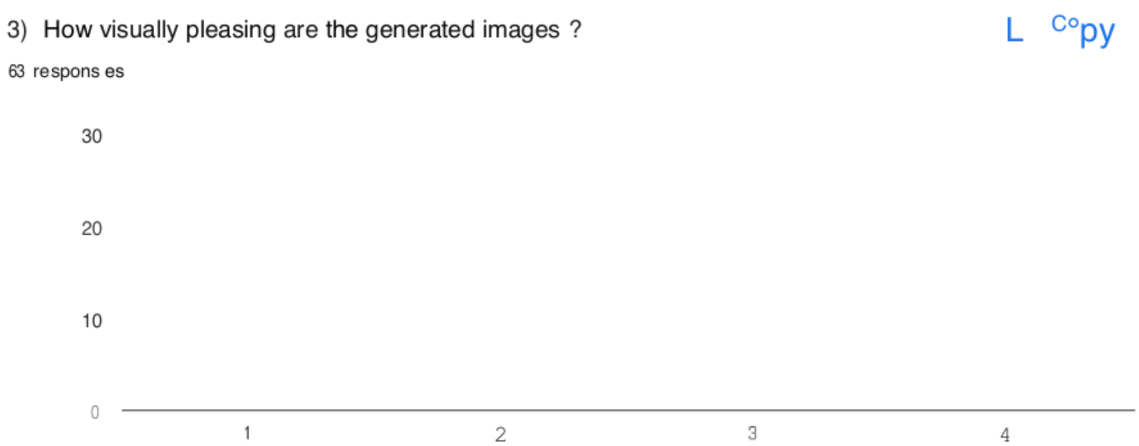


Figure 5.16: Images from the surevy

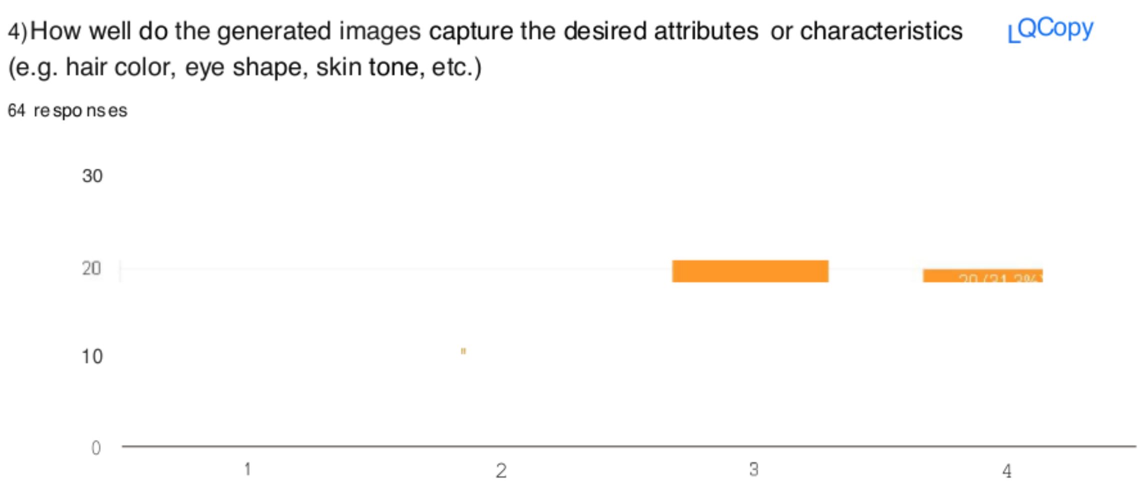


Figure 5.17: Images from the surevy

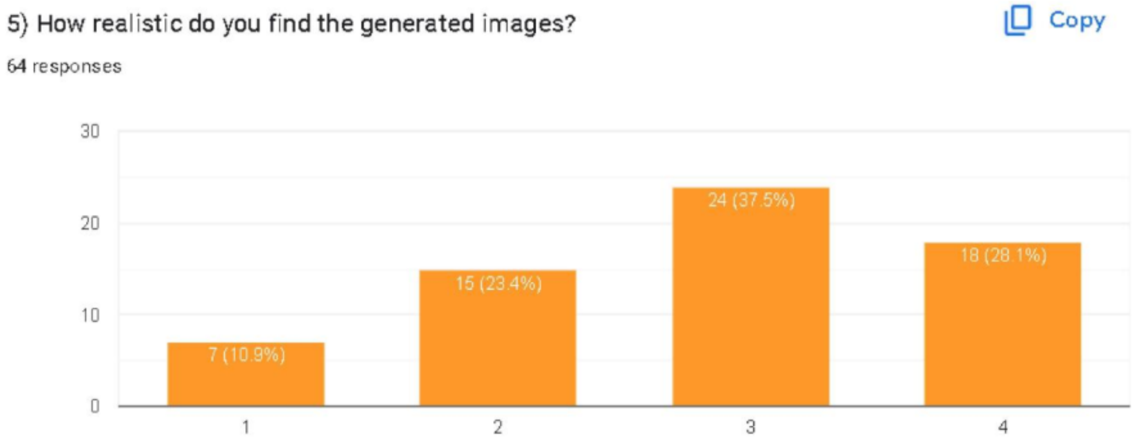


Figure 5.18: Images from the surevy

5.0.3 Objective AzzaJysls

The objective value for face frontalization using TP-GAN can be evaluated using various performance metrics. Some commonly used metrics include:

LPIPS stands for Learned Perceptual Image Patch Similarity, which is a metric used to measure the perceptual similarity between two images.LPIPS leverages deep neural networks to learn representations of image patches, which are small regions of an image, and computes the similarity between these patches using cosine similarity or Euclidean distance. These patch-wise similarities are then aggregated to obtain a final similarity score between the two images. A LPIPS score of 0.262, when normalized to the range of -1 to 1, indicates a moderate level of dissimilarity between the two images being compared. A score of 0 would represent perfect similarity, while a score of 1 would represent maximum dissimilarity. Therefore, a LPIPS score of 0.262 suggests that the two images have some differences in terms of their perceptual content and structure, but they are not completely dissimilar.

Objective analysis		
Epochs	G-Loss	LPIPS(Distance)
197	0.354	0.436
0515	0.319	0.344
1037	0.282	0.294
1918	0.265	0.264
2317	0.232	0.285
3138	0.239	0.262

Table 5.1: Objective analysis

5.0.4 SR GAN

Single image super-resolution (SISR) is a technique with the intent of restoring or recreating a high-resolution (HR) image from its corresponding low-resolution (LR) observation. . A high-resolution (HR) image can be recovered or recreated from its matching low-resolution (LR) observation using the single image super-resolution (SISR) technique. Our approach upscales low-resolution photos by a factor of x4 using a deep convolutional neural network (CNN). It is a development of ESRGAN, a cutting-edge super-resolution technique that creates high-quality images using a GAN. Residual-in-Residual Dense Blocks (RRDBs), which strengthen the network's capacity to learn more complicated characteristics in the images, are one of the extra features that this model adds to improve ESRGAN. In order to improve the super-resolution procedure even more, the model also makes use of feature maps at various scales. This model's ability to produce high-resolution photos with realistic details has been enhanced by being trained on a large-scale dataset of high-quality images, notably human faces. To build HR-LR image pairs and train the model, the model uses a two step degradation strategy that more accurately mimics the degradation that images experience in the real world.

Figure 5.19: TP GAN output vs SR GAN output

Figure 5.20: TP GAN output vs SR GAN output

Figure 5.21: TP GAN output vs SR GAN output

Chapter 6

Conclusion and Future Scope

6.0.1 Conclusion

In conclusion, face frontalization using TP-GAN is a powerful technique for synthesizing frontal face images from non-frontal face images.The approach has shown impressive results on several benchmark datasets and has been demonstrated to be effective for a wide range of applications, including face recognition and facial expression analysis.Overall, face frontalization using TP-GAN is a promising technique with significant potential for various applications in computer vision and facial analysis. We have successfully trained the model to generate the output of 3200 epochs with the lowest generator loss of0.233 and the discriminator loss is 0.797 constant.Overall, face frontalization using GANs is a promising technique that has the potential to revolutionize the way we use facial recognition and detection systems. With further research and development, it could become a valuable tool for various industries, including entertainment, security, and healthcare.

6.0.2 Future Work

With the advances in technology give this project is majorly used for generating frontal faces from side profile images. For future purposes, we can take inputs, as not only side images, but also images taken from different angles which will be useful in real world applications. With the proper resources and a more efficient software version we can also incorporate a lager dataset to give more accurate output images. Super resolution can be applied to enhance the quality if image generated.

RE-2022-113726-plag-report

ORIGINALITY REPORT

14%	8%	9%	7%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.researchgate.net Internet Source	2%
2	medium.com Internet Source	1%
3	web.archive.org Internet Source	1%
4	Submitted to St. Xavier's College Student Paper	1%
5	ir.aiktclibrary.org:8080 Internet Source	1%
6	Submitted to Felician College Student Paper	1%
7	Zhaoqing Pan, Weijie Yu, Bosi Wang, Haoran Xie, Victor S. Sheng, Jianjun Lei, Sam Kwong. "Loss Functions of Generative Adversarial Networks (GANs): Opportunities and Challenges", IEEE Transactions on Emerging Topics in Computational Intelligence, 2020 Publication	1%

8	"Computer Vision – ECCV 2018 Workshops", Springer Science and Business Media LLC, 2019 Publication	1 %
9	arxiv.org Internet Source	<1 %
10	webthesis.biblio.polito.it Internet Source	<1 %
11	Guodong Guo, Na Zhang. "A survey on deep learning based face recognition", Computer Vision and Image Understanding, 2019 Publication	<1 %
12	Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero et al. "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network", 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017 Publication	<1 %
13	repositori.upf.edu Internet Source	<1 %
14	www.ncbi.nlm.nih.gov Internet Source	<1 %
15	Submitted to Australian Institute of Higher Education Student Paper	<1 %

16	Xinyi Duan, Hao Liu, Jiuzhen Liang. "DIQA-FF:dual image quality assessment for face frontalization", Multimedia Tools and Applications, 2023 Publication	<1 %
17	Submitted to Cork Institute of Technology Student Paper	<1 %
18	Submitted to Pace University Student Paper	<1 %
19	Xian Wu, Kun Xu, Peter Hall. "A survey of image synthesis and editing with generative adversarial networks", Tsinghua Science and Technology, 2017 Publication	<1 %
20	Submitted to Carnegie Mellon University Student Paper	<1 %
21	www.semanticscholar.org Internet Source	<1 %
22	"PRICAI 2018: Trends in Artificial Intelligence", Springer Science and Business Media LLC, 2018 Publication	<1 %
23	Submitted to University of Wales Swansea Student Paper	<1 %
24	Mukhiddin Toshpulatov, Wookey Lee, Suan Lee. "Generative adversarial networks and	<1 %

their application to 3D face generation: A survey", Image and Vision Computing, 2021
Publication

25	cse.anits.edu.in Internet Source	<1 %
26	openaccess.thecvf.com Internet Source	<1 %
27	towardsdatascience.com Internet Source	<1 %
28	P. High Court durai, S. Jayasathyan, V. Shanjai Sethupathy, M. Bharathi. "Implementation of Real Time Virtual Clicking using OpenCV", 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS), 2022 Publication	<1 %
29	Yilong Lu, Tanfeng Sun, Xinghao Jiang, Ke Xu, Bo Zhu. "Frontal View Synthesis Based on a Novel GAN with Global and Local Discriminators", 2019 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), 2019 Publication	<1 %
30	microsolresources.com Internet Source	<1 %

www.kdnuggets.com

31 Internet Source <1%

---

32 Khansa Rasheed, Junaid Qadir, Terence J. O'Brien, Levin Kuhlmann, Adeel Razi. "A Generative Model to Synthesize EEG Data for Epileptic Seizure Prediction", IEEE Transactions on Neural Systems and Rehabilitation Engineering, 2021 <1%
Publication

---

33 "Pattern Recognition and Computer Vision", Springer Science and Business Media LLC, 2018 <1%
Publication

---

34 Rui Huang, Shu Zhang, Tianyu Li, Ran He. "Beyond Face Rotation: Global and Local Perception GAN for Photorealistic and Identity Preserving Frontal View Synthesis", 2017 IEEE International Conference on Computer Vision (ICCV), 2017 <1%
Publication

---

35 Yanpeng Cao, Zewei He, Zhangyu Ye, Xin Li, Yanlong Cao, Jiangxin Yang. "Fast and accurate single image super-resolution via an energy-aware improved deep residual network", Signal Processing, 2019 <1%
Publication

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		