2024 3rd International Conference for Innovation in Technology (INOCON)
Karnataka, India. Mar 1-3, 2024

Image Inpainting for Facial Recognition using Generative Networks

Pooja Srinivasan
Department of CSE
B. M. S. College of Engineering
Bangalore, India
poojasrinivasan18@gmail.com

Anusree Manoj K
Department of CSE
B. M. S. College of Engineering
Bangalore, India
anusreekmanoj2000@gmail.com

Manu Saraogi
Department of ECE
B. M. S. College of Engineering
Bangalore, India
saraogimanu@gmail.com

Jeevan Nataraju
Department of ECE
B. M. S. College of Engineering
Bangalore, India
jeevannataraju.30@gmail.com

Dr. Geetishree Mishra
Department of ECE
B. M. S. College of Engineering
Bangalore, India
geetishreemishra.ece@bmsce.ac.in

Dr. Selva Kumar S
Department of CSE
B. M. S. College of Engineering
Bangalore, India
selva.cse@bmsce.ac.in

Saritha A N
Department of CSE
B. M. S. College of Engineering
Bangalore, India
saritha.cse@bmsce.ac.in

2024 3rd International Conference for Innovation in Technology (INOCON) | 979-8-3503-8193-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/INOCON60754.2024.10511863

Abstract—**Face recognition algorithms are complicated by the presence of a face mask that obscures the majority of facial features. People wear face masks for a variety of reasons; some wear them to shield themselves from pollution, while others wear them to disguise their feelings. Masks are worn by criminals to conceal themselves from surveillance cameras and other devices. Owing to the COVID-19 pandemic, masks have become a necessity. Many Machine Learning approaches have been implemented for image inpainting and object removal. Through this study, few of the available Generative Adversarial Networks are reviewed. Further these architectures are compared and the ones suitable for removing masks from the face and regenerating the masked portion of the face have been listed. GANs are essentially computational structures that involve two neural networks against one another to generate new, synthetic examples of data that can be mistaken for real data. In this paper an analysis of four different GAN architectures have been presented. These architectures have been broadly used in image inpainting. These architectures can be used to unmask the masked faces by inpainting the portion of the face with a mask.**

Index Terms—**Generative Adversarial Networks (GANs), Face Recognition, Image inpainting**

I. Introduction

Machine learning is an area of computer science that helps computers learn without being explicitly programmed. It can improve on its own as a result of experience and data. A wide variety of applications like email filtering, computer vision, speech recognition, and also medicine use machine learning algorithms where otherwise developing conventional algorithms to perform the above tasks is very hard or impossible. Facial recognition, a technology where a person can be identified only by just looking at them, uses machine learning techniques. Here the facial traits are detected, collected, stored, and analyzed so that they can be compared with the photographs of people in a database.

With the advent of technology, Face Recognition systems have gained a lot of popularity in numerous applications like access control, security, video surveillance, attendance recording systems, etc. The existing face recognition systems use different facial features to identify an individual. Face recognition algorithms are challenged by the existence of a face mask that covers most of the facial features. People tend to wear a face mask for several reasons; few wear masks as protection from pollution while few wear them to hide their emotions. Criminals wear masks to hide themselves from surveillance cameras and so on. Due to the COVID-19 outbreak, masks have become a necessity.

GANs are a sort of generative modeling that uses convolutional neural networks and other deep learning techniques. A generator and a discriminator are the two models that make them up. The generator creates synthetic or fictitious images that resemble training images. The discriminator determines whether an image is real or fake by examining an image and its output. During the training phase, the Generator model and the Discriminator model compete with each other and both of them are defined as Neural Networks. During this phase, the generator makes better false images, deceiving the discriminator into thinking that a real image is created, while the discriminator attempts to improve at detecting and categorizing the image as real or fake. There are three parts to GANs:

Generative: A generative model is a learning model, which is a probabilistic model that defines how data is generated.
Adversarial: An adversarial environment is used to train a model.
Networks: The artificial intelligence (AI) algorithms employ deep neural networks for training purposes.
There are various types of GAN. Some of the types of GAN architecture are Vanilla GAN, MocoGAN, Conditional GAN (CGAN), DiscoGAN, Deep Convolutional GAN (DCGAN), Laplacian Pyramid GAN (LAPGAN), AlphaGAN, Super Resolution GAN (SRGAN), CycleGAN, DualGAN, SRGAN,StyleGAN, Face Completion GAN, SinGAN [1]. In this study we focus on DCGAN and CycleGAN.

An analysis of four GAN architectures CycleGAN, Pix2Pix, DCGAN and NestedGAN are presented in this work. In picture inpainting, these architectures have been widely used. By inpainting the portion of the face with a mask, these architectures can be used to unmask the masked faces. The regenerated face image thus produced can be used in facial recognition for a variety of applications.

Organization:
The paper is presented as follows: An overview of some related research that has been done previously is summarized in section 2. The implementation of the proposed methodology of the work conducted is elaborated in section 3 of this study. The methodology discusses how the dataset was prepared for this work and then proceeds to the different GAN architectures discussed that are CycleGAN, Pix2Pix, DCGAN and NestedGAN. Section 4 is reserved for results and analysis followed by section 5 which provides the conclusion to this study.

II. Related Work

Object detection and object removal have been a topic of research in recent years, with numerous deep learning approaches presented that generate outstanding results in image editing applications. Satoshi Iizuka et. al. [2] have proposed a novel image completion methodology to produce locally and globally consistent images. This deep convolutional neural network based method is trained for completing the arbitrary missing areas in an image. A single completion network is used for image completion, and local and global context discriminators to ensure consistency in image completion. Unlike region-filling and patch-based approaches [3] for image removal and completion, this work can produce fragments that don't occur anywhere else in the real image. A deep generative model for implementing an efficient technique for face completion is proposed by Li et. al [4]. Face completion is more challenging than the well-studied backdrop completion task because it frequently demands the production of semantically new pixels for missing critical components (e.g., mouth and eyes) with significant appearance changes. This work uses neural networks for generating content for missing areas unlike other nonparametric algorithms that synthesize by looking for patches. Two adversarial losses, a reconstruction loss, and a semantic parsing loss are used to train the model, which assures consistency and pixel faithfulness between local and global contents. This work shows qualitatively and quantitatively that the model can deal with a huge area of missing pixels in any shape and create realistic face completion results using comprehensive testing findings.

Image inpainting or completion which is filling in the missing pixels of an image, has numerous applications in image editing, computational photography, etc. Several Deep Learning algorithms and architectures have been developed for this application [5]. Many of the existing image inpainting approaches use patches from the same image to synthesize the patches for completing the missing patch which would appear similar to the known regions of the image. These methods work in cases of background completions such as grass, mountains, sky, buildings, etc. But in cases where complex images like faces are involved, this strategy cannot be applied. But image regeneration and image completion applications for complex images like faces have been made possible with the introduction of Generative Adversarial Networks (GAN) [6].

With the help of deep learning-based algorithms, good results are obtained for the tough problem of inpainting huge missing sections in an image. A variety of visual structures and textures can be generated by these methods, but typically deformed structures or blurry textures are produced that are not in sync with the surrounding environment. Convolutional neural networks' inability to clearly borrow or replicate information from distant geographical locations is the primary reason for this. Contrarily, conventional methods of texture and patch generation are quite helpful for capturing textures from the environment. These difficulties can be overcome using GAN-based approaches. Jiahui Yu et al. [5] developed a new deep generative model-based technique capable of not only synthesizing novel image structures but also directly using surrounding image properties as references during the training of the network to increase prediction accuracy. During the testing period, the model is a fully convolutional neural network and a feedforward capable of interpreting sizes and images with many holes in random places. Muhammad Kamran Javed Khan et al. [7]presented MRGAN, an interactive technique that allows the user to approximate the microphone area. A generative hostile network-based image-to-image conversion approach was employed to fill the gap. N. Ud Din et al. [8] proposed a novel method that uses GAN architecture to remove Face masks from the masked images and regenerate the masked portion of the face. This work unmasks the masked face in two stages, the object detection phase, and the image completion phase. The first stage detects the mask object in the masked facial image. The missing region of the face is synthesized with the actual facial structure in the second phase. To keep the face's genuine structure and consistency, this work employs two discriminators. The global face structure is learned with the help of one discriminator, while the deep missing region is learned with the help of the other discriminator. Various GAN architectures have been introduced for object removal and image completion.

The generator and discriminator in DCGAN use convolutional and convolutional-transpose layers, respectively. Radford et al. proposed the idea. The discriminator is composed of batch normalization layers, strided convolution layers, and LeakyRelu as the activation function. A 3x64x64 picture can be entered. The generator consists of batch normalization layers, ReLU activations, and convolutional-transpose layers. There will be a 3x64x64 RGB picture as a result [9].

The CycleGAN model was created to address the difficulty of image-to-image translation. A training set of matched picture pairings is used by the image-to-image translation problem and it aims to learn the mapping between a source and output image. Without needing matched input-output images, CycleGAN learns this mapping using cycle-consistent adversarial networks. [10].

This work analyzes various GAN architectures with respect to its application in face mask removal and face image completion. The reconstructed images can be used in various facial-recognition applications, for example, criminal identification.

III. Proposed Work

With the current global outbreak of COVID 19, it has become essential to wear face masks and they have become an integral part of our daily lives. People are encouraged to cover their faces in public places to avoid the spread of the infection. Face masks have lowered the accuracy of facial recognition systems, which can be utilized in a variety of situations when facial identification is required, such as criminal identification. Facial recognition is also used in many offices, schools and other businesses for authentication and verification. Masking the face, on the other hand, has no effect because it is difficult to detect and recognize.

Thus we make an attempt to address the above problem stated in two parts: unmasking the masked face using the GAN approach and unmasked face identification. GANs are a type of neural network that can be used for unsupervised learning. The generator and discriminator refer to the two neural networks that make up a GAN.
Generator: Generates the facial features in the masked region of the face
Discriminator: Classifies the generated image (impainted with facial features in the masked region) as real or fake

A. Dataset Preparation

This work requires a dataset consisting of a set of masked images and their corresponding unmasked images of individuals. First images of people were collected from the CelebA [11] dataset. This consisted of more than 200k unmasked images of celebrities. A python script was used to synthetically generate masked images by applying face masks . The coordinates to apply the face mask was determined using the facial landmarks. The facial landmarks for a given face can be obtained using the Dlib library. The 68 (x, y)-coordinates associated with facial structures are estimated using the pre-trained facial landmark detector from the dlib library. The final dataset used in our work included both unmasked and masked faces. The dataset consisted of around 5000 images each with a resolution of 32x32x3. These images were loaded in the form of a numpy array.

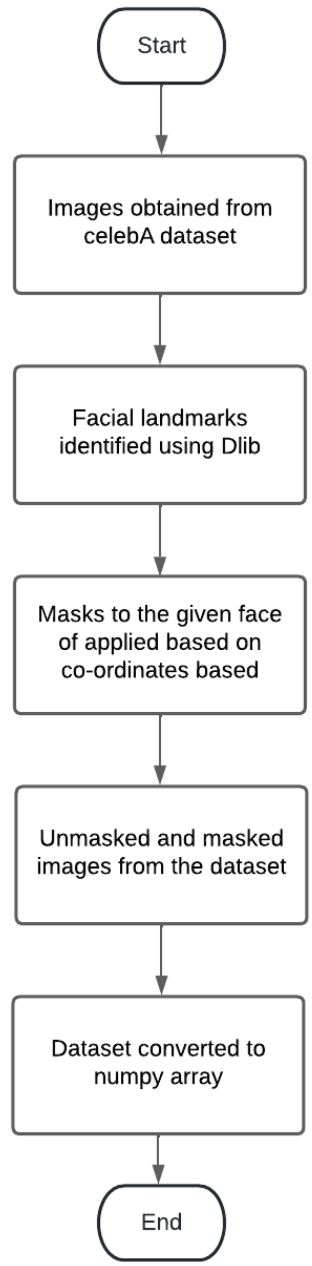

Fig. 1. Dataset Preparation

B. Unmasking the masked face

This work analyzes 4 GAN architectures for unmasking the masked face.

1) DCGAN: GANs are used to train deep learning models to produce new data from the same distribution of training data. DCGAN is one of the most well-known and successful GAN designs. The Convolution layers with no maximum pooling or entirely connected layers are the primary design that DCGAN follows.For downsampling and upsampling, it employs transposed convolution and convolutional stride.

DCGAN replaces pooling all maximum with convolutional stride, uses upsampling for transposed convolution and eliminates fully connected layers. It uses Batch normalisation with the exception of the generator's output layer and the discriminator's input layer. It uses ReLU in the generator except for the output which uses tanh and the discriminator is LeakyReLU .

To downsample the input picture in the discriminator model, a conventional convolutional layer is three convolutional layers followed by a stride of 22. The model simply contains one sigmoid activation function node within the output layer, and there are no pooling layers to forecast whether the input sample is true or false. For binary classification, the model minimizes the binary cross entropy loss function. Since it lacks zero-slope sections, LeakyReLU is employed instead of ReLU to solve the "dying ReLU" problem. Dropout is used to prevent all neurons in a layer from maximizing their weights at the same time. The Adam version of stochastic gradient descent can be utilized. Instead of employing pooling layers, the large stride can be used to achieve a comparable downsampling effect. The model will get batch updates using a collection of real examples featuring a variety of manufactured samples.The generator model converts a vector from 100-dimensional latent space to a 2-dimensional image array.

DCGAN has been used in different image inpainting applications involving facial image datasets like CelebA [12]. It can be applied for applications involving object removal and image inpainting such as face mask removal and face restoration, by improving the existing architecture.

2) Cycle GAN: The Cycle Generative Adversarial Network (CycleGAN), is one of the GAN approaches where a deep convolutional neural network is trained for image-to-image translation tasks. Unlike other existing GAN models, a dataset of paired images is not required for image translation. The image to image translation models are trained automatically without paired image examples. The unsupervised training method is employed that uses an unrelated set of images from both the source and target domains. This implementation is employed in different application domains with impressive results. Few of the notable implementations are in translation of photographs of horse to zebra, and also zebra to horses as a reverse implementation. As shown in Fig. 2 two generator models are employed in the CycleGAN architecture:

- First generator - Images for domain one (Domain-A) are generated (Generator-A)
- Second generator - Images for the domain two (Domain-B) are generated (Generator-B).

The image generation by the generator models are conditional in nature, that is they consider an image from the supplementary domain. An input image from the second domain (Domain-B) is used for Generator A and an input image from the first domain (Domain-A) is used for Generator-B. There are two discriminator models defined corresponding to each generator model:

- Discriminator model one (Discriminator-A) - The prediction as real or fake is made by considering images generated from Generator-A and real images from Domain-A.
- Discriminator model two (Discriminator-B) - The prediction as real or fake is made by considering images generated from Generator-B and real images from Domain-B.

The discriminator model and the generator models are trained in an adversarial zero-sum process. The discriminators are trained to detect the fake images effectively and the generators are trained to effectively fool the discriminators. Together the models during this training, find an equilibrium. The discriminator predicts how likely the output image is real or fake, for an input image. For the model:

- Convolutional-BatchNorm-LeakyReLU layer patterns are used.
- InstanceNormalization is used where values on each of the output feature maps are standardized.

The discriminator model implements the 70×70 PatchGAN discriminator model [10]. An input image of size 256×256 is considered by the model and a patch of predictions is given as output. The model implements mean squared error for the least square loss which is used for optimization. A weighting scheme is used to have updates with half (0.5) the usual effect for the model. During training, the changes to the discriminator are slowed down due to the weighting of model updates, when compared to the generator model [13]. The output of the model either takes up a single value or a square activation map of values where each value maps to the probability of how likely a patch in the input image is real. This is computed based on the input image size. A classification score or an overall likelihood score can be computed by taking the average of these output values. An encoder decoder model architecture is followed by the generator. The model is intended to generate the target image i.e an un-masked image taking a masked image as input. Initially, the downsampling or encoding of the input image to a bottleneck layer is done. A number of ResNet layers are used to interpret this encoding and skip connections are used by these layers. Further it is followed by a series of layers for this representation to be upsampled or decoded to the size of the target image. From the source domain, more versions of reconstructed input images can be translated by regularizing the generator models. A recursive technique can

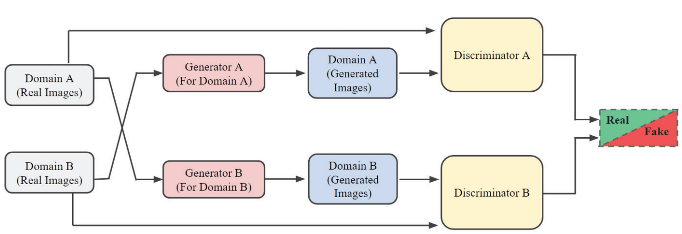

Fig. 2. Cycle GAN Architecture

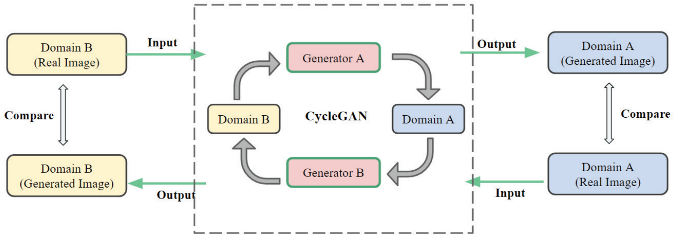

Fig. 3. Cycle GAN Representation

be employed to achieve this, where generated images are used as input by the corresponding generator model and the corresponding output images are compared with the original images as shown in Fig. 3. This technique where input image is passed through both the generators, is termed as a cycle. Each generator model pair is together trained to effectively regenrate the original input image. This is termed as cycle consistency. From the target domain, input images can be considered by the generator and the same image generation without change can be expected, to result in an enhanced color profile matching of the input image. This is termed as identity mapping.

Loss Function

CycleGAN loss primarily comprises GAN loss with generators G and F and discriminators DX and DY [13]. For a generator and discriminator pair evaluated on X and Y, the GAN loss is given by (1),

$$\begin{aligned}\mathrm{LGAN}(G, D_Y, X, Y) = \mathbb{E}_{y\sim p_{\text{data}}(y)}[\log D_Y(y)] \\ + \mathbb{E}_{x\sim p_{\text{data}}(x)}[\log(1 - D_Y(G(x)))]\end{aligned} \tag{1}$$

The image translation cycle for an input image, should be able to retrieve the image back to the original image. This cycle consistency is expected for both the forward and the backward cycle and they are expressed as (2),

$$\begin{aligned}&\text{Cycle consistency for forward cycle:} \quad x \to G(x) \to \\ &\qquad F(G(x)) \approx x \\ &\text{Cycle consistency for backward cycle:} \quad y \to F(y) \to \\ &\qquad G(F(y)) \approx y\end{aligned} \tag{2}$$

A cycle loss term which penalizes the unnecessary image changes is given by (3)

$$\begin{aligned}L_{\text{cycle}}(G, F) = \mathbb{E}_{x\sim p_{\text{data}}(x)}[\|F(G(x)) - x\|_1] \\ + \mathbb{E}_{y\sim p_{\text{data}}(y)}[\|G(F(y)) - y\|_1]\end{aligned} \tag{3}$$

where Lcyc(G, F) evaluates the distance between the original image and the result of attempted mapping of that image into the complementary dataset and then back into the original dataset. The total loss is given by (4),

$$\begin{aligned}L(G, F, D_X, D_Y) = L_{\text{GAN}}(G, D_Y, X, Y) + L_{\text{GAN}}(F, D_X, Y, X) \\ + \lambda L_{\text{cycle}}(G, F)\end{aligned} \tag{4}$$

where λ is the cycling parameter and the relative importance of the two objectives is controlled by this.

The discriminator models are trained with the help of real images and the generated images. The generator model training is through their related discriminator models. They are updated:

- Based on adversarial loss associated with mean squared error, intended to minimize the loss of generated images marked as "real", predicted by the discriminator. This way they are intended to generate images that effectively fit into the output domain.
- Based on cycle loss (both backward and forward) associated with mean absolute error, which gives efficiency of input image regeneration when employed with the other generator model. Based on identity loss associated with mean absolute error, where an output image without translation is expected to be generated when an example is provided from the target domain.
- More weight (10-times) is given to the cycle loss than the adversarial loss [13]. Always, half of the cycle loss (5-times) is used as weighting for the identity loss.

3) Pix2Pix GAN: Pix2PixGAN is a popular method for image-to-image conversion. It is built on a conditional GAN, which generates a target image from a certain input image. The loss function of this scenario is adjusted by Pix2Pix GAN so that the generated image is reliable in both the content of the target domain and the plausible input image translation. The Pix2Pix GAN networks learn not only the input-to-output image mapping but also the loss function used to train the mapping. This helps in the application of the same generic method to problems that normally require very different loss equations. The generator model, unlike a typical GAN model, accepts an image as input instead of a point from the latent space. Unpredictability is caused by the use of dropout layers, which are used both during training and prediction.

The Generator in this architecture uses a U-Net model instead of the usual encoder-decoder model. It is similar to the encoder-decoder model in terms of downsampling the input image to a bottleneck and then upsampling again to an output image but encoder layers are connected to corresponding decoder layers having feature maps of the same size by using skip-connections. Fig. 4 shows the skip connections in the U-Net architecture.

Isola et. al [13] proposes the linear combination of L1- loss between target image, generated image, and GAN loss for defining the generator loss. Instead of the usual GAN model, which employs a deep convolutional neural network in the discriminator to determine if the images are real or fake, the Pix2Pix model employs a PatchGAN. PatchGAN is a deep convolutional neural network model that identifies images as real or fake by classifying patches of images rather than the complete image. It compares an image created in the target domain to an image generated in the source domain. The design of PatchGAN is based on the receptive field size, which is sometimes referred to as the effective receptive field [14].

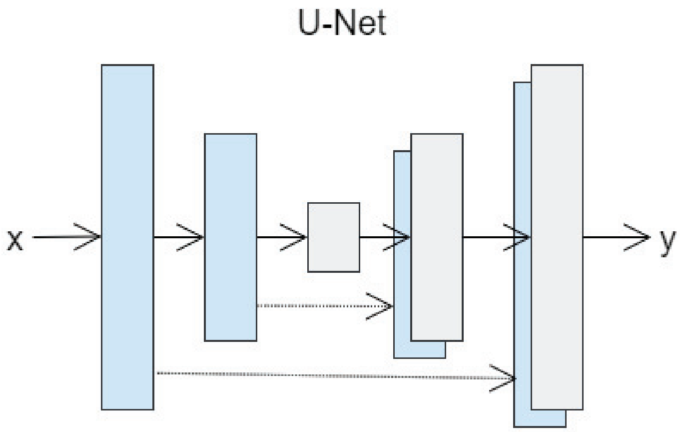

Fig. 4. U-Net Architecture

This work proposes Pix2Pix GAN as one of the GAN architectures for unmasking the masked face. Pix2Pix GAN [15] is one of the best-suited architectures for image to image translation and image inpainting. The proposed process will be an image to image translation and inpainting where the masked portion of the face is painted with plausible facial features. The output image can be used for facial recognition.

4) Nested GAN: Face inpainting (face completion) is the process of creating plausible facial structures to fill in missing pixels in a face image. Face inpainting's primary goal is to generate a more legible and visually realistic face image from a masked region or missing content in an image. Nested GAN is one such GAN-based architecture that can be implemented because it has direct parallelism with the application in question. Facial inpainting can be used to unmask the masked face and bring it as close to the original image as possible using this nested GAN architecture.

The following is the model architecture as shown in Fig. 5. It is composed of a nested generative convolutional network, which means it has two generators and three discriminators that collaborate to form a nested structure of one GAN inside another to improve performance. To improve training, the network also employs Dilated Convolution and Novel Residual Connections. The study [16] describes a training procedure for the problem of hole filling in facial images by means of a generative nested CNN network. It consists of two distinct generation networks known as described in Fig. 5.

The **Sub-Confrontation Generation Network** is made up of the code generator and code discriminator. The corrupted image is provided as input and the encoded code information is given as output which is used for decoding later. This network assists in the extraction of the image's strong features as well as the location of the image's missing or corrupted area. It generates a blurry image with the missing area partially fixed and the rest of the image nearly intact. Because they have an adversarial structure, the code generator generates images and the code discriminator classifies them,

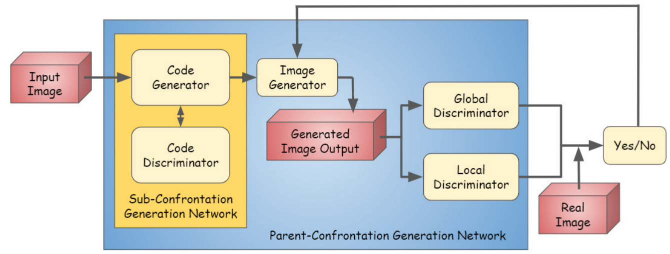

Fig. 5. Nested GAN model architecture

and both networks are trained alternately, they form a GAN. Li et al. [16]proposed the structure for this generation network.

The **Parent-Confrontation Generation Network** is divided into two parts, generation and discrimination. It attempts primarily to reconstruct the repaired image from the corrupted image. The Sub-Confrontation Generation Network is included in the generation section, and it outputs the partially fixed image to another image generator, which uses it to construct a fixed image. In this way, two different generators are used to generate the same fixed image twice, increasing the model's robustness. The discriminator section contains two discriminators: a global discriminator and a local discriminator. The global discriminator examines the entire image to determine the image's overall authenticity. The local discriminator, on the other hand, only looks at the corrupted area of the image. To determine whether an image is real or fake, both discriminators collaborate with a fully connected layer.

Loss Functions:

NGAN employs a total of five networks: two generators and three discriminators. As a result, the model's five networks use five loss functions: the code generator, the code discriminator, the image generator, the global discriminator, and the local discriminator[16].

- The entropy deviation of information between the code generator's input and output is referred to as **code generator loss**. It is propagated backwards through the code generator.
- The distance between the synthesized image and the ground truth is defined as **code discriminator loss**. It is back-propagated through the code discriminator.
- The GAN's generator loss as well as the loss in the coding reconstruction process is defined as the **image generator loss**. It is back-propagated through the image generator.
- The accuracy of differentiating synthesized image and ground truth is computed by **global and local discriminator loss**.

IV. Results and Analysis

On performing the analysis using a subset of celeba dataset, the DCGAN model provides the following output. It was observed that after 12 epochs the real accuracy was found to be

24% and fake accuracy was found to be 88%. Similarly after 20 epochs the real accuracy was found to be 80% and fake accuracy was found to be 98%. It is seen that accuracy varies between epochs, and accuracy for real examples may correlate more strongly with subjective image quality than accuracy for fake ones.

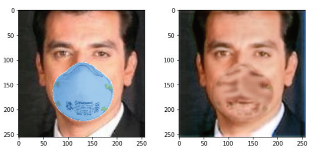

Fig. 6. DCGAN output

The approach used in the above implementation generates a random face at every epoch and the quality of the regenerated image increases with the number of epochs. Further training it for more epochs, we see a face with the masked area regenerated to some extent but the results obtained are not clear and satisfactory. After choosing a final generator model, it can be used to re-generate a face on its own. Fig.6 shows the input and output image after 150 epochs with DCGAN architecture.

Whereas, the results produced by the CycleGAN model improves upon training and it is shown in the below figure. Fig.7 shows the input and output images after 150 epochs obtained with CycleGAN architecture. It was observed that with fewer epochs the masked region of the image had changed its color and tried to acquire the color of the skin of that particular picture. Thus the model was further trained for 150 epochs and the results were observed. It was found that the masked portion was converted to the corresponding facial features as the epochs were increased. Therefore, CycleGAN is a comparatively better approach for object removal and reconstruction when compared to DCGAN when complex images like faces with a lot of different characteristics and structures are considered.

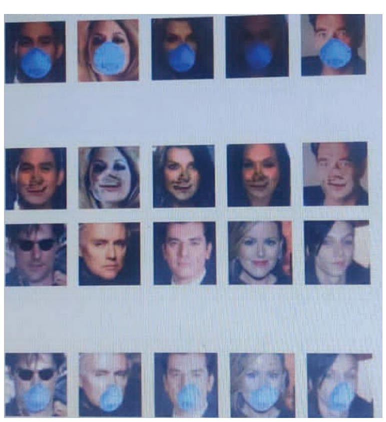

Fig. 7. CycleGAN output

Pix2Pix GAN, which is based on CycleGAN and specializes in pixel to pixel transformation of input image to output image can be used to get better results compared to the CycleGAN model. The Generator in this architecture uses a U-Net model which takes an image as input rather than a random point from latent space. The Pix2Pix model utilizes a PatchGAN, which in turn uses a deep CNN in the discriminator to identify whether the images are real or fake. PatchGAN neural network model classifies patches of images as real or fake instead of the entire image and it compares the generated image in the target domain with an image from the source domain. Hence this architecture of Pix2Pix GAN is one of the best suited architectures for facial image inpainting and image to image translation. The model can be used to remove the face-mask from the image and paint the masked region of the image with plausible facial features. The output image obtained can further be used for facial recognition.

Nested GAN is another GAN architecture which can provide great results with facial inpainting related applications. It uses a nested architecture with one generative network inside another to improve performance. The corrupted image is fed into the Sub-Confrontation Generation Network, which outputs a blurry image. It includes a code generator as well as a code discriminator. The blurry image is fed into the Parent-Confrontation Generation Network, which then feeds it to the image generator in order to generate a fixed image, which is further fed into the global and local discriminators. To improve training, the model also employs dilated convolution and novel residual connections. It further employs five loss functions since its architecture utilizes five networks. Hence it is well suited and very helpful for Facial Inpainting and Face completion applications.

V. Conclusion

Face completion is a task wherein the missing portions of the face are rightly predicted and filled to ensure that the facial features are complete. This can further make the process of identifying the person a lot easier when compared to a picture of that person with missing portions in it. It has a variety of applications and with the recent outburst of covid and usage of masks, this can become a handy solution in recognizing people with masks. Furthermore it can also prove beneficial in identifying criminals, burglars, terrorists and individuals who try to hide their identity by covering their faces inorder to perform some unethical tasks. Generative Adversarial Networks have proved as a good solution to the problems of facial inpainting. There are various types of GANs available and to choose a particular GAN architecture for this purpose can sometimes be time consuming and exhausting. This work analyzes a few important GAN architectures that could be used for facial image reconstruction after extracting facial images from masked faces. Results obtained from DCGAN and the Cycle GAN approach present significant feature reconstruction and the unmasking capabilities. The analysis results and the observations on the training procedure, and architecture details like the loss functions used, shows that Pix2Pix GAN and NestedGAN can be further explored in terms of facial mask removal and facial reconstruction. Thus a descriptive analysis of the above models and their performance is summarized in this work. Necessary modifications to the above-mentioned models can further optimize the overall performance with respect to facial feature inpainting and thus solving the problem of face reconstruction with improved accuracy.

REFERENCES

[1] Z. Wang, Q. She, and T. E. Ward, "Generative adversarial networks in computer vision: A survey and taxonomy," *ACM Computing Surveys (CSUR)*, vol. 54, no. 2, pp. 1–38, 2021.

[2] S. Iizuka, E. Simo-Serra, and H. Ishikawa, "Globally and locally consistent image completion," *ACM Transactions on Graphics (ToG)*, vol. 36, no. 4, pp. 1–14, 2017.

[3] A. Criminisi, P. Perez, and K. Toyama, "Object removal by exemplar-based inpainting," in *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, vol. 2, pp. II–II, IEEE, 2003.

[4] Y. Li, S. Liu, J. Yang, and M.-H. Yang, "Generative face completion," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3911–3919, 2017.

[5] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative image inpainting with contextual attention," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5505–5514, 2018.

[6] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[7] M. K. J. Khan, N. Ud Din, S. Bae, and J. Yi, "Interactive removal of microphone object in facial images," *Electronics*, vol. 8, no. 10, p. 1115, 2019.

[8] N. U. Din, K. Javed, S. Bae, and J. Yi, "A novel gan-based network for unmasking of masked face," *IEEE Access*, vol. 8, pp. 44276–44287, 2020.

[9] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

[10] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, pp. 2223–2232, 2017.

[11] Z. Liu, P. Luo, X. Wang, and X. Tang, "Large-scale celebfaces attributes (celeba) dataset," *Retrieved August*, vol. 15, no. 2018, p. 11, 2018.

[12] F. Zhang, X. Wang, T. Sun, and X. Xu, "Se-dcgan: a new method of semantic image restoration," *Cognitive Computation*, vol. 13, no. 4, pp. 981–991, 2021.

[13] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1125–1134, 2017.

[14] W. Luo, Y. Li, R. Urtasun, and R. Zemel, "Understanding the effective receptive field in deep convolutional neural networks," *Advances in neural information processing systems*, vol. 29, 2016.

[15] Z. Li, H. Zhu, L. Cao, L. Jiao, Y. Zhong, and A. Ma, "Face inpainting via nested generative adversarial networks," *IEEE Access*, vol. 7, pp. 155462–155471, 2019.

[16] H. Yin, Y. Wei, H. Liu, S. Liu, C. Liu, and Y. Gao, "Deep convolutional generative adversarial network and convolutional neural network for smoke detection," *Complexity*, vol. 2020, 2020.