Low-quality facial image restoration using generative models

Christian Dave Cobalida
School of Computing (Dublin City University)
MSc. in Computing (Artificial Intelligence)
Dublin, Ireland
christian.cobalida2@mail.dcu.ie

Yamini Pravin Karpe
School of Computing (Dublin City University)
MSc. in Computing (Data Analytics)
Dublin, Ireland
yamini.karpe2@mail.dcu.ie

***Abstract*—This paper presents a study on the task of restoring facial images using image inpainting techniques. The approach involves repairing damaged or obscured areas in low-quality images by filling them with believable content. The study focuses on the use of patch-based and exemplar-based inpainting methods, which leverage image priors to achieve smoother and more coherent results. The proposed methodology utilises a dataset of facial images and employs a two-step process involving edge prediction and image completion. The results demonstrate the model's ability to understand image context and accurately restore images with correct colors and content preservation. The limitations of the model are discussed, along with potential areas for improvement.**

***Index Terms*—GAN, in-painting, masked, CNN, MSE, BCE, generator, discriminator, PatchGAN, Skip connections, adversarial**

I. Introduction

Restoring facial images is a difficult task that attempts to improve the quality of images that have been damaged or obscured. This is particularly useful in areas such as biometrics, security, entertainment, and social media. Unfortunately, low-quality images often have missing or damaged areas that can't be fixed through traditional methods. As a result, image inpainting techniques are commonly used to fill in these areas with believable content, making them a popular choice for restoring images.

Inpainting encompasses various approaches, ranging from manual methods to diffusion-based approaches [1]. Among these, patch-based and exemplar-based inpainting are commonly used for simpler tasks. Patch-based synthesis involves extracting similar patches from an exemplar image to fill in the missing regions [2]. Exemplar-based inpainting takes a step further by considering a larger search area across the entire image, allowing for a more comprehensive search for suitable patches. This approach excels in handling complex inpainting scenarios by identifying matching structures or patterns [3]. Leveraging image priors, exemplar-based methods achieve smoother and more coherent results, enhancing the overall visual fidelity of restored images.

Machine learning-based approaches like Context Encoders (CE) employ convolutional neural networks and adversarial loss for image restoration tasks. While CE effectively fills missing regions with similar reconstructions, it may struggle with fine-grained details and become noticeable for larger missing areas. Nonetheless, CE serves as a promising starting point for addressing the challenges in image restoration [4].

Another significant advancement in this field is the Pix2Pix approach [5], which combines conditional GAN with a reconstruction loss. The generator and discriminator architectures are derived from DCGAN, with specific modifications such as concatenated skip connections and the use of a PatchGAN discriminator to penalise structure at the patch scale. While Pix2Pix exhibits the ability to fill in regions with similar reconstructions, it may still encounter difficulties with fine-grained details and noticeable artefacts for larger missing regions. Nevertheless, Pix2Pix provides valuable insights for further improving image restoration techniques.

A number of design improvements are made to this existing work in our approach. To evaluate generated images at macro and micro levels, we use a global discriminator and a PatchGAN [6] discriminator. The publically available masks from NVIDIA [7] can be used in various ways to damage images. The experiments use the Flickr-Faces-HQ Dataset (FFHQ) [8] with 256x256-sized facial images. Considering computational limitations, we only trained models for one category of images. To keep important details and improve the quality of the generated regions, input images are downsampled to 64x64 and then reconstructed using upsampling with skip connections.

In order to improve and upgrade the generator model in our GAN setup, we use a weighted combination of Binary Cross Entropy (BCE) loss and Mean Squared Error (MSE) loss. Incorporating a global discriminator in our model lowers the overall loss and improves the image quality by capturing the overall context of the image, leading to better consistency in the generated outputs. This global discriminator works together with the PatchGAN discriminator.

In this paper, we discuss the following topics :

- Review of previous works on the image inpainting and restoration in the related work section.
- Review of the dataset used, Image preprocessing, the network design of the generator and discriminator, and the different key design aspects and the loss function in the proposed method section.
- Review of the architecture design steps, visual evaluation of the model in addition to analytical evaluation and

comparison with other architectures in the evaluation section.
- Review of the system limitation, future work, and finally the conclusion.

II. Related Work

Over the years, image inpainting, a crucial aspect of image restoration, has undergone various approaches, among them being patch synthesis. Inpainting primarily aims to repair damaged images and fill in missing regions seamlessly. While patch-based methods have shown promise in tackling image inpainting tasks, they do have certain limitations, particularly when dealing with novel objects in images [9]. These techniques heavily rely on neighbouring pixels and scour available images to identify suitable substitutes for the affected regions [10]. As a consequence, they might face challenges when it comes to handling large missing areas or replacing novel objects, often resulting in out-of-place inpainted regions. Moreover, the computational speed of patch-based methods can raise concerns, making them less practical for real-world applications where efficiency is crucial.

Many recent approaches are centred around CNN-based methods [4] [9] [11] [12] [13] [14] [15] [16] [17] [18] [19] [20] [21] [22] [23] which allows for learning features and can lead to better and more robust methods for image inpainting, especially when it comes to novel objects in images. While CNN-based approaches are more robust and faster than patch-based methods, this alone is not enough as it can lead to results lacking some finer details [4]. This is why many CNN-based approaches incorporate additional functionality and metrics to preserve the quality of the generated regions.

A method that has been used to enhance the quality of the filled-in areas is by utilising Generative Adversarial Networks (GANs). GANs function by putting several networks in opposition to each other. This includes a generative model that produces a new output and discriminators that classify the generator's outputs as authentic or fake [26]. Many different variations of this approach have been employed with varying degrees of success. Several methods have been developed based on the initial proposal in [4], which involves a singular global discriminator. The aim of the model is to decrease adversarial and reconstruction loss. Numerous iterations have modified the original discriminator configuration, including the utilisation of a PatchGAN discriminator [6] and the addition of extra discriminators [13] [14] [16] [18] [20] [23] [24].

Various techniques have been developed to assess generated images on a global and local level [12] [14] [18] [23]. These methods involve analysing both the image as a whole and individual regions separately. This is achieved by combining various types of losses. To further enhance the results, some methods use a local discriminator that improves the quality of the generated regions and overall image. However, one limitation of such methods is that they require identification of the damaged region and a fixed size during training. This information is passed on to the local discriminator during the training process.

In [14] [18] and [23] we can see that they use Wasserstein GAN (WGAN) loss whereas in [13] GAN loss [26] is implemented. WGAN uses the Earth-Mover distance metric [27] for comparing the distribution of generated 'fake' images to the real data distributions that are observed, this is particularly useful when a generator gets stuck at creating the same type of images over and over again. While all three approaches incorporate dual discriminators, they each employ a very different network architecture ranging from multiple encoders [18] [23] in one network to two networks end-to-end [14].

Many approaches make use of irregular masks during training [4] [11] [14] [14] resulting in a robust implementation that can handle holes of any shape and size. One approach which combines the concept of irregular masks and evaluation at a local scale is the use of a PatchGAN discriminator alongside a global discriminator [16]. PatchGAN discriminators function by converting an input image to an N x N format, with each pixel representing a section of the image. Each pixel is then categorised as either real or fake. This approach enables the use of non-uniform masks and combines the benefits of multiple discriminators at both local and global levels.

Many models include dilated convolution, which involves inserting zeros between filter weights. This expands the receptive field of a kernel without adding more parameters, leading to shorter training time and less complexity. Another advantage of this method over downsampling is that it doesn't decrease the spatial resolution of the resulting image. [28].

Different papers have used various techniques to improve the inpainting performance. In [16], Interpolated Convolution was employed to enhance the performance. In addition to the dilated convolution, [18] used Multi Column CNNs with different kernel sizes (3x3, 5x5, 7x7) to achieve different receptive fields. They also used a spatial variant reconstruction loss (weighted L1 loss) where each missing pixel is assigned a weight based on its distance from a known pixel. Pixels further away from known pixels are given a lower weight while those closer are given a higher weight. [4] used a channel-wise fully connected layer to capture features from distant spatial locations in feature maps by connecting each channel independently, thereby reducing the number of parameters. [13] did not use a fully connected layer to allow for flexibility in image size and reduce the number of parameters, hence decreasing training time. In [11], Partial Convolution was introduced to separate missing pixels from valid pixels during convolution. They also used perceptual loss and style loss, which are L1 distances between feature values and gram metrics of pre-trained VGG-16 features maps, respectively. [12] combined data-driven CNNs with conventional copy-and-paste methods. They used a shift-connection layer and skip connections to inject the "copy and paste" method by concatenating the encoded features into the decoded features to borrow the nearest neighbour information from valid pixels. In [9], enhancements were made to the context encoder in [4]. They employed a texture network (Style transfer network) to ensure that the pixel textures in the in-painted region are the same as the valid pixels [29]. They used a pre-trained VGG16 to extract feature maps and a gram matrix to calculate the correlation between valid and in-painted pixels. The concept of EdgeConnect was introduced in [20]. The first generator predicted the skeleton of the missing parts (Edge prediction) and used it as a condition for the second generator to fill colours (Image completion) and avoid the blurry effect caused by the L1 Loss and ensure local fine texture details.

III. Methodology

In this section, we will break down the approach taken into step-by-step details of each element of the design, beginning with the selected dataset.

A. Dataset

Due to the fact that the chosen task is focused on repairing facial images, the Flickr-Faces-HQ Dataset dataset was chosen. In terms of age, ethnicity, and image background, the dataset contains 70,000 high-quality PNG images at 1024x1024 resolution [8]. The dataset consists of 70 folders, each containing 1000 images. However, due to limited computing resources, we had to work with a subset of the dataset consisting of only one category of images, with a total of 11336 images. We divided this subset into three sets - train, validation, and test - with an 80/10/10 split. This resulted in 9062 images for training, 1151 for validation, and 1123 for testing.

For the purpose of training our model, the images from the dataset were pre-processed to create damaged images with irregularly shaped holes. This was done using 12,000 publicly available NVIDIA masks [29] with different hole to images ratios for each 1000 mask. Images were damaged by combining the images with a randomly selected mask using OpenCV python library and masks were selected in a way to ensure an unbiased distribution of masks between the training, validation and testing images. Figure 1 shows an example of how damaged images can be generated.

The code has been developed using Python with some of the available libraries (Tensorflow, Keras, NumPy, OpenCV and Matplotlib). The code was developed using Visual Studio Code on Linux Ubuntu. Using a local AMD GPU with a capacity of 12GB, the system had moderately fast performance. Despite this, it was only given 40 hours to complete the task.

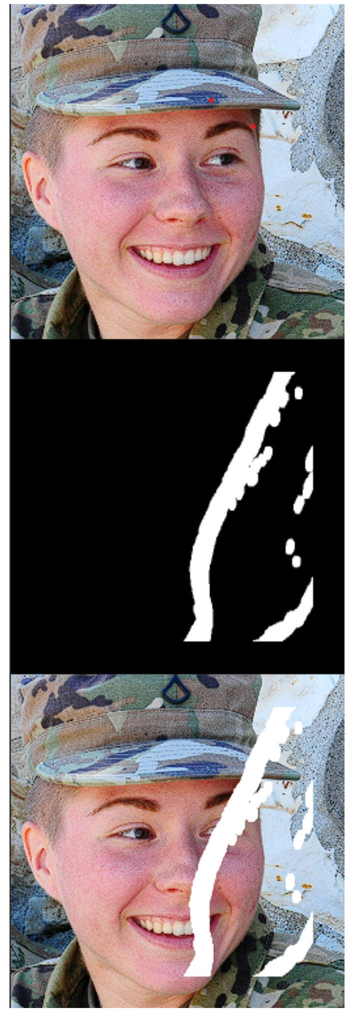

Fig. 1. Example of damaging an image using an NVIDIA mask

B. Network Design

Our chosen architecture builds upon the work covered in [9] which uses a GAN architecture to complete the task. The performance of the generative model is driven by the discriminator networks' loss values as they compete against each other in a minimax game competing to fool each other.

- ***Generator network***: The network responsible for generating the output is designed using the U-Net architecture [5]. Unlike traditional GAN models, this network does not require noise as input. Instead, it takes in an image that is damaged and processes it to create a latent representation. This latent representation is then used to reconstruct the image to its original size, while also repairing the damaged regions. The input images of 256 x 256 are compressed to 64 x 64 representations with stride convolutional layers. Later, they are reconstructed back to their original size of 256 x 256 with transposed convolutional layers. This approach is preferred over pooling layers as it results in minimal information loss. The generator model is comprised of 17 convolutional layers. The downsampling layer applies a LeakyReLU activation function, while the upsampling layer uses a ReLU activation function. It has been discovered that using LeakyReLU for downsampling and ReLU for upsampling is an effective combination. A successful technique for generating images, including image-to-image translation and image super-resolution, involves using LeakyReLU for downsampling and ReLU for upsampling. This approach enables the model to capture the essential features during downsampling and accurately reconstruct the image during upsampling, as supported by research [9] [32]. The final layer employs a hyperbolic tangent activation function. The first 16 layers of the generator network feature batch normalisation,

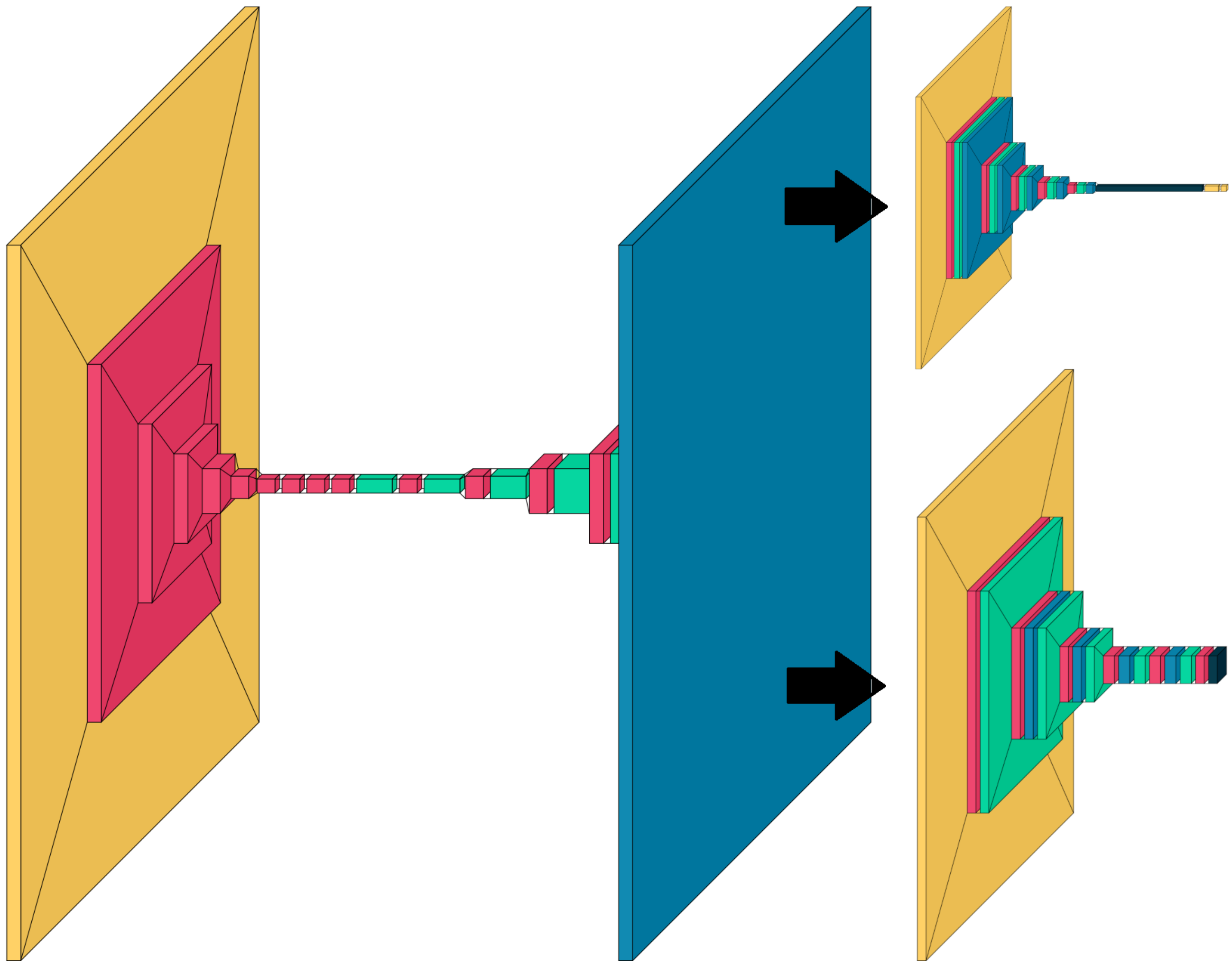

Fig. 2. GAN architecture model with global and PatchGAN discriminators

which sets inputs to have a mean of zero and a variance of one. This helps to prevent internal covariance shift, stabilizes the training process, and prevents mode collapse as shown in Figure 1. To retain important information during image downsampling, skip connections are employed. These connections link the encoder's previous layer to its corresponding upsampling layer in the decoder. By doing so, skip connections aid in recovering the original spatial resolution [31]. The generator network's final layer uses a hyperbolic tangent activation instead of a sigmoid activation. This is because it has a larger gradient, which allows for larger weight updates and mitigates the problem of a vanishing gradient. It also has symmetry around zero, leading to faster convergence. Table I shows the design of the generator layer.

- ***Global Discriminator***: To ensure that the generator network produces higher-quality images, we use a discriminator network that aims to trick the discriminators. Our approach involves using dual discriminators that assess images at both local and global levels. This enables us to generate finer details in images for better results. The global discriminator is responsible for classifying generated images as real or fake, and then relaying this information back to the generator network. By using the loss to adjust the generator's weights, we can improve the overall quality of the generated images. The global discriminator takes a 256 x 256 input from the generator network and uses strided convolutions to reduce the input to a 4 x 4 feature map before flattening and passing it to a single node fully connected layer with a sigmoid activation function. You can find the design of the Global Discriminator layer in Table II.
- ***PatchGAN Discriminator***: For the second discriminator, we use a PatchGAN [6] discriminator. Its purpose is to act as a local discriminator and improve the fine details of generated regions. This helps to reduce the blur

TABLE I
GENERATOR MODEL ARCHITECTURE

Kernel Size	Stride	Number of Filters	Activation
(5, 5)	(2, 2)	64	Leaky ReLU
(3, 3)	(2, 2)	128	Leaky ReLU
(3, 3)	(2, 2)	128	Leaky ReLU
(3, 3)	(2, 2)	256	Leaky ReLU
(3, 3)	(2, 2)	256	Leaky ReLU
(3, 3)	(2, 2)	256	Leaky ReLU
(3, 3)	(2, 2)	256	Leaky ReLU
(4, 4)	(2, 2)	256	ReLU
(4, 4)	(2, 2)	256	ReLU
(4, 4)	(2, 2)	256	ReLU
(4, 4)	(2, 2)	128	ReLU
(4, 4)	(2, 2)	128	ReLU
(4, 4)	(2, 2)	64	ReLU
(4, 4)	(2, 2)	3	Tanh

TABLE II
GLOBAL DISCRIMINATOR MODEL ARCHITECTURE

Layer Type	Kernel Size	Stride	Number of Filters	Activation
Conv2D	(5, 5)	(2, 2)	32	Leaky ReLU
Conv2D	(5, 5)	(2, 2)	64	Leaky ReLU
Conv2D	(5, 5)	(2, 2)	128	Leaky ReLU
Conv2D	(5, 5)	(2, 2)	256	Leaky ReLU
Conv2D	(5, 5)	(2, 2)	256	Leaky ReLU
Dense	-	-	512	ReLU
Dense	-	-	1	Sigmoid

in the generated regions. Since we are using irregular-shaped masks of varying sizes, it's necessary to evaluate generated images in patches to determine if they are real or fake. We use strided convolutions to reduce the input down to a 16 x 16 feature map, where each pixel represents a patch of the input image. You can find the design of the PatchGAN Discriminator layer in Table III.

- ***Loss Functions***: Loss functions play a crucial role in determining the performance of the generator model. There are two types of losses that affect it:
  - Adversarial loss, which is measured using binary cross entropy (BCE).
  - Reconstruction loss, which is calculated as the mean squared error (MSE) between the inpainted image and the ground truth.

TABLE III
LOCAL(PATCHGAN) DISCRIMINATOR MODEL ARCHITECTURE

Layer Type	Kernel Size	Stride	Number of Filters	Activation
Conv2D	(4, 4)	(2, 2)	32	Leaky ReLU
Conv2D	(4, 4)	(2, 2)	64	Leaky ReLU
Conv2D	(4, 4)	(2, 2)	128	Leaky ReLU
Conv2D	(4, 4)	(2, 2)	256	Leaky ReLU
Conv2D	(4, 4)	(2, 2)	256	Leaky ReLU
Conv2D	(4, 4)	-	256	Leaky ReLU
Conv2D	(4, 4)	-	1	Sigmoid

The discriminator models are trained by using batches of real and fake images. The real images are taken from the dataset, while the fake images are generated by the generator model. The global discriminator produces a single output of either 1 or 0, indicating whether an image is real or fake. The PatchGAN discriminator produces a 16 x 16 output, corresponding to 256 patches of the input image, where each patch is classified as either a 1 or 0.

$$\min_G \max_D \mathbb{E}[\propto \mathrm{MSE}(x, G(z)) + \log D(x) + \log(1 - D(G(z)))] \quad (1)$$

It is shown in Algorithm 1 how the general training procedure works. During the GAN training loop, the generator model's parameters are updated, while the weights of the discriminators are frozen. The generator produces damaged images, which are then passed to each discriminator and classified as either natural or fake. The loss from each model is propagated back through the GAN model and used to update the generator model parameters. To stabilize the GAN training loop and speed up the learning process, the MSE error is used. The generator and discriminators are playing a minimax game, where the generator aims to reduce reconstruction loss and adversarial loss, while the discriminators aim to increase adversarial loss as shown in (1). The weight of the MSE and adversarial loss is controlled by a hyperparameter, denoted by $\propto$.

IV. EVALUATION

The MSE loss decreased as the generator learned and was capable of producing images similar to the ground truth images. During the initial stages, the PatchGAN discriminator loss was high while the global discriminator loss was low. The models were not yet trained to differentiate between real and fake images, but by epoch 70, the discriminators learned to distinguish between them. As the generator started to produce believable images, the discriminator loss increased

Algorithm 1: Training process

Input : Iterations I_{Total}
Output: Trained generator and discriminator networks
while $iterationsI \leq I_{Total}$ **do**
  Sample a batch of real images and generate a batch of fake images;
  Update both discriminator networks D using real images and BCE loss;
  Update both discriminator networks D using fake images and BCE loss;
  Update generator network damaged images using weighted MSE/BCE Loss;
  if $I \mod 5 = 0$ **then**
    Sample a batch of real test images and generate a batch of fake test images;
    Evaluate both discriminator networks D using real test images and BCE loss;
    Evaluate both discriminator networks D using fake test images and BCE loss;
  end
end

again because it was being fooled. Figure 3 display the various losses.

TABLE IV
LOSS METRICS COMPARED TO OTHER MODELS

Model	L1 Loss	L2 Loss
Our Model	**5.55%**	**1.04%**
Context Encoder [4]	9.37%	1.96%
Patch-Based [6]	5.54%	1.19%
Multi-Scale Neural Patch Synthesis [9]	10.01%	1.82%
Contextual Attention [14]	8.6%	2.1%
Free Form with Gated-Convolution [21]	9.1%	1.6%

Due to the limitation of the available computing power, The model was trained for 40 hours period of 220 epochs using a batch size of 32 images and the results are shown in Figure 4. It can be noticed that the model is able to understand the context of the image and is able to restore the images with the correct colours and content preserving the overall image context. In Table IV, the comparison between the performance of our model is compared with models from other papers using L1 Loss and L2 Loss metrics. The L1 Loss of our model is 5.55% while the L2 loss is 1.04% which is exceeding and comparable to the performance of other models although the model has not been trained on a big dataset due to the limitations of the available computing power.

Our model, like many other machine learning applications, may encounter difficulties when presented with images that are not within the scope of the data it was trained on. Since the model was primarily trained on facial images, there may be little to no carryover when applied to landscape or urban environments. This results in the model failing to understand the face context and determine the location of the landscape features. However, if the system was fine-tuned on an out-

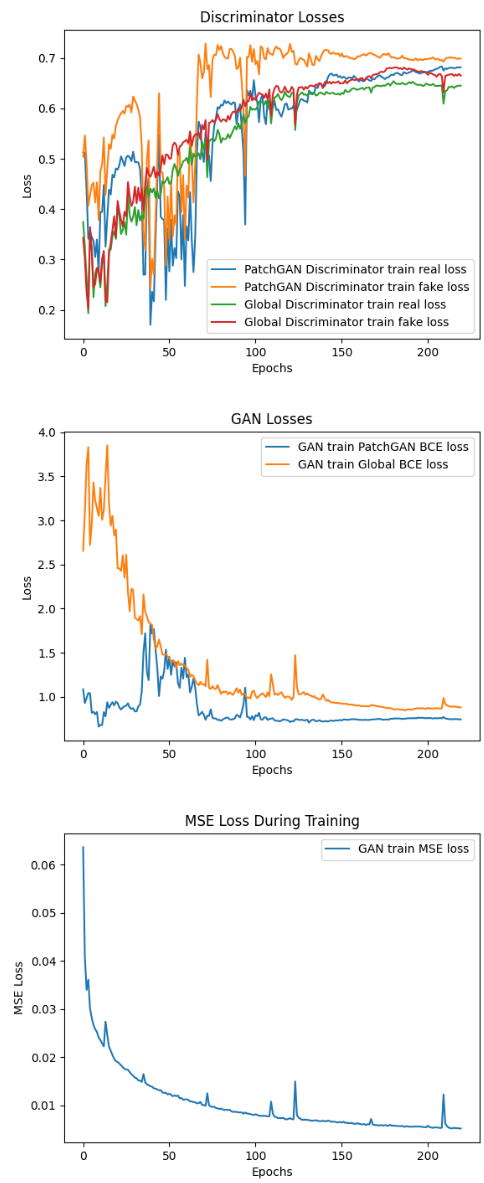

Fig. 3. GAN training loss plots for Discriminators, GAN, and MSE

of-scope dataset, it is possible that its performance would be improved. Additionally, if images are damaged with a mask that covers most of the input images, the model's ability to produce accurate results is limited as there would be little information left of the image to base the inpainted region off of.

The system is set up to only accept images that are 256 x 256 in size. If an image of a larger resolution is submitted, it would need to undergo further processing. However, this could lead to a loss of information in the final output image. Additional methods could be used in parallel with the existing system in order to upsample the output images to the original resolution but this is currently not an available option.

One limitation we faced was the insufficient computing

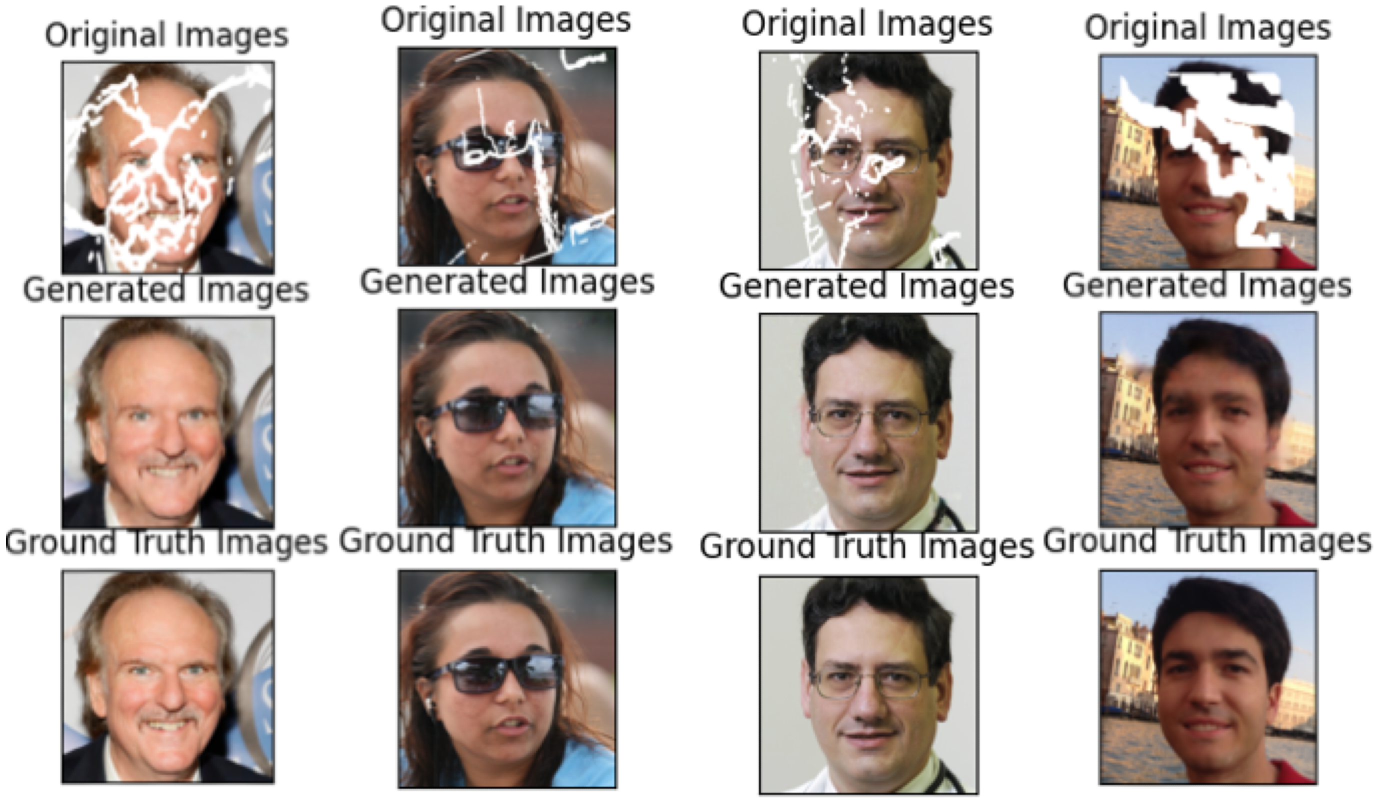

Fig. 4. Restored Images using L2 loss

power that prevented us from training the entire Flickr-Faces-HQ Dataset (FFHQ). Due to resource sharing during training and limited time, we had to select a subset of images from the dataset to evaluate our proposed approach. This restricted the scope of the final model.

V. Future Work

Although our approach has limitations, there is room for improvement. Time constraints and hardware availability are two factors that limit us. For future experimentation, we can perform additional training over a longer period of time using the entire FFHQ dataset [8], as well as more powerful hardware. This would result in a more robust and complete final model, capable of performing inpainting on a wider range of images. As well, we may add layers or adjust layers to be more complex in the generator in order to improve image quality or to better capture the long-range dependencies between input and output pixels.

Additionally, we may explore different network architectures, including the use of partial convolutions as they have been shown to improve inpainting results when dealing with irregularly shaped holes/masks [11] in addition to other kinds of losses to be incorporated in the loss functions like perceptual loss and style loss. This system may also be expanded to work for images of all sizes, including high-resolution images and images from different domains including landscape and urban images.

VI. Conclusion

We have developed a method to restore images that maintains global and local consistency. Our approach combines skip connections in the generator model and a PatchGAN-based [6] local discriminator to achieve the best results. While we were unable to complete a full training cycle due to limited computing power, the generated images demonstrate reasonably good outcomes. Inpainted regions show consistency in colour and produce promising results at a local level. The combination of skip connections and a local discriminator is effective in preserving image quality and reducing loss, resulting in good L1 and L2 loss outcomes. With additional training and more complex layers, we anticipate that this approach will produce higher-quality images with inpainted regions that are difficult to distinguish from the original unaltered images.

References

[1] K. M. T, M. C. C, and M. A. F. Majeed, "A Survey on Different Techniques used for Image Inpainting," International Research Journal of Engineering and Technology (IRJET), vol. 7, no. 12, pp. 1382-1387, 2020.

[2] C. Barnes and F.-L. Zhang, "A survey of the state-of-the-art in patch-based synthesis," Computational Visual Media, vol. 3, no. 1, pp. 3-20, 2017.

[3] A. Criminisi, P. Perez and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," in IEEE Transactions on Image Processing, vol. 13, no. 9, pp. 1200-1212, 2004.

[4] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell and A. A. Efros, "Context Encoders: Feature Learning by Inpainting," CoRR, 2016.

[5] Isola, Phillip, et al. “Image-to-Image Translation with Conditional Adversarial Networks.” 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.
[6] T. Nguyen and M. Yoo, ”PatchGAN-Based Depth Completion in Autonomous Vehicle,” 2022 International Conference on Information Networking (ICOIN), pp. 498-501, 2022.
[7] G. Liu, F. A. Reda, K. J. Shih, T.-C. Wang, A. Tao and B. Catanzaro, ”NVIDIA Irregular Mask Dataset,” in https://nv-adlr.github.io/publication/partialconv-inpainting, 2018.
[8] https://github.com/NVlabs/ffhq-dataset
[9] C. Yang, X. Lu, Z. Lin, E. Shechtman, O. Wang and H. Li, ” High-Resolution Image Inpainting using Multi-Scale Neural Patch Synthesis,” in Proc. International Conference on Computer Vision and Pattern Recognition (CVPR), 2017
[10] A. A. Efros and W. T. Freeman, ”Image Quilting for Texture Synthesis and Transfer,” in Proceedings of the 28th annual conference on Computer graphics and interactive techniques, New York, 2001.
[11] G. Liu, F. A. Reda, K. J. Shih, T.-C. Wang, A. Tao and B. Catanzaro, ”Image Inpainting for Irregular Holes Using Partial Convolutions” in Proc. European Conference on Computer Vision (ECCV), 2018.
[12] Z. Yan, X. Li, M. Li, W. Zuo and S. Shan, ”Shift-Net: Image Inpainting via Deep Feature,” in European Conference on Computer Vision (ECCV), 2018
[13] S. Iizuka, E. Simo-Serra and H. Ishikawa, ”Globally and Locally Consistent Image Completion,” ACM Transitions on Graphics, vol. 36, no. 4, pp. 1-14, 2017.
[14] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu and T. S. Huang, ”Generative Image Inpainting with Contextual Attention,” in Proc. Computer Vision and Patterns Recognition, 2018
[15] Z. Yi, Q. Tang, S. Azizi, D. Jang and Z. Xu, ”Contextual Residual Aggregation for Ultra High-Resolution Image Inpainting,” in Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, 2020.
[16] U. Demir and G. Unal, ”Patch-Based Image Inpainting with Generative Adversarial Networks,” 2018.
[17] K. He, X. Zhang, S. Ren and J. Sun, ”Deep Residual Learning for Image Recognition,” vol. abs/1512.03385, 2018.
[18] Y. Wang, X. Tao, X. Qi, X. Shen and J. Jia, ”Image Inpainting via Generative Multi-column Convolutional Neural Networks,” in Proc. Neural Information Processing Systems, 2018.
[19] L. A. Gatys, A. S. Ecker and M. Bethge, A Neural Algorithm of Artistic Style, 2015.
[20] K. Nazeri, E. Ng, T. Joseph, F. Z. Qureshi and M. Ebrahimi, ”EdgeConnect: Generative Image Inpainting with Adversarial Edge Learning,” in Proc. International Conference on Computer Vision (ICCV), 2019.
[21] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. Huang, ”Free-Form Image Inpainting with Gated Convolution,” in Proc. International Conference on Computer Vision (ICCV), 2019.
[22] M. Heo and Y. Choe, ”Single-image reflection removal using conditional GANs,” in Proceedings of the International Conference on Electronics, Information, and Communication (ICEIC), Auckland, 2019.
[23] R. Fujii, R. Hachiuma and H. Saito, ”RGB-D Image Inpainting Using Generative Adversarial Network with Late Fusion Approach,” CoRR, vol. abs/2110.07413, 2021.
[24] T. A. Mohite and G. S. Phadke, ”Image inpainting with contextual attention and partial convolution,” in Proceedings of International Conference on Artificial Intelligence and Signal Processing (AISP), Amaravati, 2020.
[25] Z. Wang and S. Ji, ”Smoothed Dilated Convolutions for Improved Dense Prediction,” in Pro. of 24th ACM SIGKDD International Conference, London, 2019.
[26] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville and Y. Bengio, ”Generative Adversarial Networks,” Advances in Neural Information Processing Systems, vol. 3, 2014.
[27]] M. Arjovsky, S. Chintala, and L. Bottou, Wasserstein GAN. arXiv, 2017. doi: 10.48550/ARXIV.1701.07875.
[28] Z. Wang and S. Ji, ”Smoothed Dilated Convolutions for Improved Dense Prediction,” in Pro. of 24th ACM SIGKDD International Conference, London, 2019.
[29] Gatys, Leon, et al. “A Neural Algorithm of Artistic Style.” Journal of Vision, vol. 16, no. 12, 2016, p. 326, https://doi.org/10.1167/16.12.326.
[30] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” 2015.
[31] M. Drozdzal, E. Vorontsov, G. Chartrand, S. Kadoury and C. Pal, ”The Importance of Skip Connections in Biomedical Image Segmentation,” 2016.
[32] B. Xu, N. Wang, T. Chen, M. Li, ”Empirical Evaluation of Rectified Activations in Convolutional Network”, 2015.

Code is made available at: https://gitlab.computing.dcu.ie/cobalic2/2023-mcm-master/-/tree/master/src