Context-aware Facial Inpainting with GANs

Jireh Jam
PhD 2021

Context-aware Facial Inpainting with GANs

Jireh Jam

A thesis submitted in partial fulfilment of the requirements of Manchester Metropolitan University for the degree of Doctor of Philosophy

Department of Computing and Mathematics
Manchester Metropolitan University

2021

Abstract

Facial inpainting is a difficult problem due to the complex structural patterns of a face image. Using irregular hole masks to generate contextualised features in a face image is becoming increasingly important in image inpainting. Existing methods generate images using deep learning models, but aberrations persist. The reason for this is that key operations are required for feature information dissemination, such as feature extraction mechanisms, feature propagation, and feature regularizers, are frequently overlooked or ignored during the design stage. A comprehensive review is conducted to examine existing methods and identify the research gaps that serve as the foundation for this thesis.

The aim of this thesis is to develop novel facial inpainting algorithms with the capability of extracting contextualised features. First, Symmetric Skip Connection Wasserstein GAN (SWGAN) is proposed to inpaint high-resolution face images that are perceptually consistent with the rest of the image. Second, a perceptual adversarial Network (RMNet) is proposed to include feature extraction and feature propagation mechanisms that target missing regions while preserving visible ones. Third, a foreground-guided facial inpainting method is proposed with occlusion reasoning capability, which guides the model toward learning contextualised feature extraction and propagation while maintaining fidelity. Fourth, V-LinkNet is proposed that takes into account of the critical operations for information dissemination. Additionally, a standard protocol is introduced to prevent potential biases in performance evaluation of facial inpainting algorithms.

The experimental results show V-LinkNet achieved the best results with SSIM of 0.96 on the standard protocol. In conclusion, generating facial images with contextualised features is important to achieve realistic results in inpainted regions. Additionally, it is critical to consider the standard procedure while comparing different approaches. Finally, this thesis outlines the new insights and future directions of image inpainting.

Acknowledgements

The fulfilment of this work over the last three years, created in this thesis is a result of help I have received from my support unit of Image Metrics Ltd and Visual Computing Lab at The Manchester Metropolitan University. First of all I would thank Prof. Moi Hoon Yap, Dr Kevin Walker, Dr Vincent Drouard and Dr Connah Kendrick for their immense support in the field of machine learning and computer vision with generative models in particular.

To my family, I express my heartfelt gratitude for always believing in me, even when I did not believe in myself. The people who have always been there for me when I needed them and who have always encouraged me to think optimistically are my father, Rev Ilaja Jam, my mother, Susan Jam, my daughter, Esther Roberts and my lovely friend Dr Cynthia Sirri, who has always been supportive and encouraging, and who has assisted me in improving my listening skills and attention to detail. I would like to thank my friends for always being there for me when I needed advice during the highs and lows of my life, especially Chiakom Patricia Ngoin, Dr Tom Agyme, Dr Irene Ambang and Dr Derick Yongabi. I would want to express my gratitude to the E121 Visual Computing Lab team, particularly Jhan Alarifi, Xulu Yao, Chuin Hong Yap, Dr Richa Agarwal, and Dr Manu Goyal. To the technical staff, I'd like to express my gratitude to everyone at Specialist Computer Sciences MMU, particularly Jay Woods and Anthony Walls, for constantly assisting me when my computer had difficulties. I am grateful to The Royal Society for providing a PhD studentship and Image Metrics Ltd for hosting me, which allowed me to complete this thesis and gain valuable industry experience.

Contents

List of Figures

Chapter 1

Introduction and Overview

In this chapter, the background, motivation, problem statement, aim and objectives are introduced to highlight key aspects that led to the development of this thesis. Also included in this chapter are the contributions and organisation of the thesis.

1.1 Introduction

Image inpainting is a widely researched topic in computer vision. The origin of inpainting came from an artistic process to conserve artwork (paintings) for historic purposes. This has evolved over the years due to the growing need of archiving digital images needing high quality restoration methods. With many techniques proposed, inpainting is an intriguing topic in research due to its impressive interpolation of pixels during the restoration process of an image. Inpainting originated from artistic conservation of paintings, which is based on the visual perception of the artist. Thus it is important to have a preliminary understanding of the Human Visual System (HSV). This is because the HVS is a useful tool in artistic work and visual perception of an image.

An image is a vast amount of pixels combined to form texture and structure to depict a visual perception of something. It can be recreated and stored in digital and non-digital format. The HVS has a good processing unit (cortex) that can remember and update images with natural eye movements as interpolation to complete an image [44]. It uses cognitive processes; a system that infers the nature of hidden structures given visible ones. With this in mind, it is clear that the cortex can interpret visual data in order to provide perception and create memories based on context and prior knowledge.. This perceptual ability in humans was put to use by artists in the past to restore damaged images from museums, churches, libraries and archive collections. In this way the artist would imagine semantic content based on overall scene, ensure texture and structural continuity of lines between regions of undamaged and damaged pixels with visually realistic content. To support this, a study by Ikkai et al. [83] and DiCarlo et al. [44] indicate that the dorsal visual stream guides the eyes towards spatial attention and movement task. However, using the artistic approach to restore images, often brought about substantial variations in details by various artist if given the same task, causing disagreements across the globe and poor restoration. For example the restoration of a damaged painting shown on Figure 1.1. Researchers achieved a breakthrough in image inpainting with the introduction of digital photography by developing computational techniques that are now regarded state-of-the-art. These algorithms are introduced and described in detail in Chapter 2. However, limitations persist across different methods, which may be the result of failures in the design stage to understand that inpainting is about information dissemination, feature extraction, and propagation. Another drawback might be the lack of a standardised testing protocol to guide research along a specified path, which will aid in the establishment of a benchmark for evaluating these algorithms rather than random mask application on images by different models. To shed more light on the key method that led to the design of all techniques proposed in this thesis, it is sufficient to introduce the encoder-decoder in comparison to the HVS.

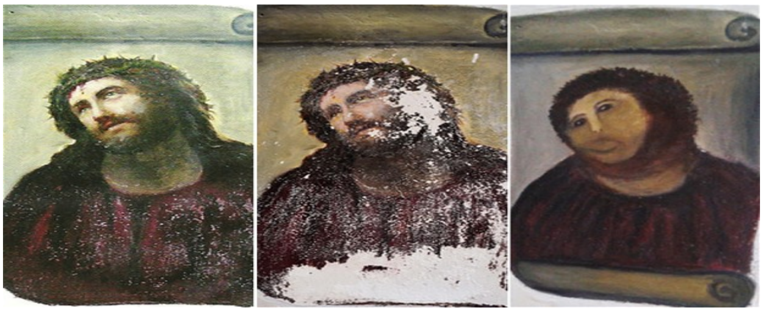


Figure 1.1: Example showing poor inpainting of the portrait of Jesus Christ. [63]

1.2 Background

Many computer vision algorithms are based on cognitive processes in the human brain that analyse and interpret various types of data. The use of artificial neural networks based on neurons in computer data analysis was intended to mimic human cognitive processes. According to Ogiela et al. [149], the stages in the human cognitive process are data analysis, features, data understanding, expectations, and knowledge. Despite the fact that neural networks attempt to mimic the brain by using mathematical frameworks to learn high-dimensional data representation, the human brain outperforms neural networks.

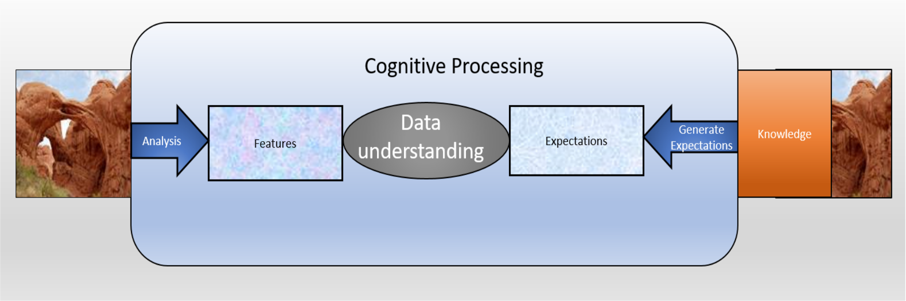


Figure 1.2: Cognitive resonance and data understanding of the human brain, which is similar to Ogiela et al. [149].

The Figure 1.2 depicts the analysis occurring in the brain as data is captured, analysed, and interpreted to generate expected outcomes using knowledge-based properties. The patterns defined in the knowledge-based properties must satisfy the set of expected outcomes. Based on this concept, it is obvious that given an image with missing information, a human can draw or complete the missing information of these regions to create an image that, when compared to the original image, is semantically consistent. Similarly, in the past, artists would restore degraded or damaged paintings. Since the beginning of the twentieth century, this has taken a digital turn, evolving from traditional inpainting methods to deep learning methods. However, the latter has emerged as the state-of-the-art restoration technique, employing convolutional neural networks within an encoder-decoder network to capture high-dimensional data abstractions based on image understanding for reconstruction best learned end-to-end. This network is not limited to images and can be extended to other data types. Figure 1.3 depicts an encoder-decoder for comparable understanding with cognition resonance in humans. The structure is comparable in that the features shown on Figure 1.2 correspond to the encoder, expectations correspond to the decoder, and data comprehension corresponds to compressed representations. In Figure 1.3, the encoder captures context (features) from an input image into latent feature representations, and the decoder reconstructs these features into an expected outcome; in the case of image inpainting, this is the generated missing content.

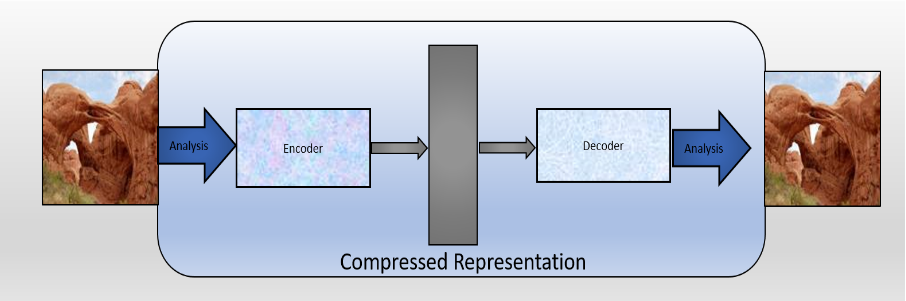


Figure 1.3: Encoder-decoder architecture.

The development of digital tools with image manipulation capability, as well as the evolution of computers in the twentieth century, has encouraged users to appreciate image editing, such as restoration, and the application of on-screen visual display and special effects to images. As a result, digital image inpainting (henceforth inpainting) has evolved into a state-of-the-art restoration technique. In a computer vision and graphics context, inpainting is a method that interpolates neighbouring pixels to reconstruct damaged, or defective, portions of an image with no discernible difference when compared to the rest of the image. On an image, these damaged portions/areas (masked-regions) are a collection of unconnected pixels surrounded by a collection of known adjacent pixels. The inpainting method uses known-information to fill the damage portions during the reconstruction of a damaged image (image with disconnected pixels). The goal of image inpainting is to produce realistic results that are coherent or consistent with surrounding contextual information.

Traditional inpainting methods accomplished this primarily by matching or copying background patches into missing regions. This was mostly accomplished by filling in the targeted patches with patch similarities from the local background regions. Some inpainting algorithms in this category use diffusion-based approaches, which are limited to narrow-hole (mask) regions, whereas another patch-based method [13] based on fast nearest neighbour field can fill-in larger masks regions. These methods, however, failed due to non-repetitive patterns on images and frequently generated verbatim copies. The most advanced methods are generative models like Variational Autoencoders (VAE) and Generative Adversarial Networks (GAN). Convolutional Neural Networks (CNNs) are used in these methods to hallucinate missing parts of an image using an encoder-decoder framework to learn and reconstruct features of the masked-image. Because of their fast inference, speed, and performance on high resolution images, generative models have become the best stage performing algorithm, attracting enormous amounts of research in this area of study.

1.3 Motivation

The breakthrough in computational methods in the 21st century spawned countless research in the area of image inpainting. The state-of-the-art techniques already exists, but visual perception differs from person to person and the continuous development of inpainting techniques is much needed to improve research in this area (more details in Chapter 2). There is a need for inpainted images with contextualised features of targeted regions in coherency with the entire image. Inpainting techniques that fall under learning based methods learn by hallucinating missing contents from images in a data driven manner [152]. Learning-based methods struggle to handle high-resolution images due to training difficulty, memory occupancy (which causes the GPU to slow), and a lack of high-resolution training data. As a result, there is still room for research to develop models that can address these limitations, allowing for training with GPU/NPU (Graphics Processing Unit/ Neural Processing Unit) devices to generate missing pixels with contextualised features.

1.4 Problem Statement

According to existing studies [209, 152, 52] in computer vision, inpainting is a learning problem that can be handled by encoding high-level features to a reconstructed output that is highly comparable to the input. Attempts were made to tackle the problem using traditional approaches [50, 18], however these methods encountered problems when dealing with complex textural and structural representations. In the case of GANs, the inpainting job is thought to have two components: an input source and a target image. The generated image is acquired by learning from the target image in the first scenario. The source image is often composed of a mask and an image, or a mask and an image that have been merged to form a single input. Because of the intricate nature of its characteristics, which can be easily noticed by humans, generating realistic face images with GANs is not a simple operation to do. One issue is the difficulty of the binary mask. In the literature, various types of masks have been identified, including square masks, irregular masks, masks at border and non-border regions [152, 137]. However, this is consistently ignored during the algorithm's design phase. Because neural networks learn relevant information by sliding a kernel across an image, the type of mask is important. As a result, a square mask will be consistent and in specific positions the majority of the time, making the network's job easier. On the other hand, an irregular mask will be applied to the entire image, making feature capture more difficult. A mask at the border region, as well as large missing textural patches, will make learning more difficult, resulting in an additional network challenge. As a result, it is critical to consider this when designing a network that can withstand any mask type. Failures may also occur depending on how the mask is applied to the image (failing to use an input image with normalised floating point representations). Information propagation is also important because the decoder requires it for high-quality reconstruction. Designing a model with efficient propagation mechanisms, either within the model or as a loss function, is thus advantageous to any inpainting GAN model.

This is due to the fact that the mask can be either square or irregular in shape, and the shape of the mask is very dependent on how it is applied on the image (failing to use an input image with normalised floating point representations).

1.5 Research Questions

In this thesis, GANs will be used to research into contextual-aware facial inpainting. In addition, methods for facial inpainting will be proposed, and the results will be evaluated and compared to the state-of-the-art methods. Three research questions will be addressed in this thesis:

- How do GANs grasp the context of high-resolution images in image inpainting?
- How can GANs comprehend the context of fidelity preservation image inpainting?
- How can GANs understand the latent space of irregular hole-regions in image inpainting?

1.6 Aim and Objectives

The aim of this research is to design novel facial image inpainting methods using GANs. The proposed method will capture contextual information to fill in missing contents of an image in a data-driven manner with contextualised features. The objectives are as follows:

- To conduct an informed study on the state-of-the-art algorithms of image inpainting, nested with comparative study of these techniques on facial images.

- To develop new deep learning architectures for image inpainting based on Generative adversarial networks (GANs).

- To propose new approaches to facial inpainting with capability of occlusion reasoning that can preserve fidelity of attributes even with large hole-to-image ratios.

- To identify potential biases in performance evaluation and conduct empirical study to compare the performance of the inpainting algorithms.

1.7 Thesis Contributions

The contributions of this thesis are newly designed methods to inpaint facial images and images of high quality with realistic features on the damaged regions. Inpainting high resolution images with fidelity preservation and contextualised features of a facial image are the main challenges to inpainting. Many algorithms have been proposed but there still persists some artefacts that are generated either due to failures in feature transfer or mask sizes being too large, thus leaving the model with totally no information. Other reasons for such failures are poor performances on images with masks at border regions, lack of edge preservation due to arbitrary masks sizes and poor performances on large irregular masks. In addition to this, the generated content of the masked regions comes from the interpolation of more than one possibility in latent space. It is challenging to reconstruct images with contextualised features because of such reasons. However, the state of the art models in face generation are GAN based, and have a high reputation for generating realistic images [98, 99, 100]. The application of GANs in inpainting is still closely studied and research is ongoing. Newly proposed inpainting GAN based models have changed the way the convolution layers are stacked. Many methods have applied different techniques to handle the mask during training. These methods have addressed most of these challenges with high-end contributions to the research community in image inpainting. To that end, this thesis adds the following contributions to the already existing methods. This thesis' main contributions are:

- A novel inpainting method namely Symmetric Skip Connection Wasserstein GAN for High resolution facial Image inpainting (SWGAN), is proposed to handle high-resolution images based on a new combination loss function that can preserve colour and luminance. Best performance was achieved as shown in Section 4.6 of chapter 4 [PUB2].

- A novel inpainting method namely RMNet, reverse masking, is proposed to explore only the missing pixels during back-propagation with the help of a reverse masking loss directly propagating gradients via a concatenated mask. The network architecture is presented in Section 5.3 of Chapter 5 [PUB3].
- A novel foreground-guided method is proposed based on a foreground loss model, where a segmented foreground mask is a feature representation of the face, thus assisting the network with occlusion reasoning of disentangle features. See Chapter 6 [PUB4].
- A novel feature representation method V-LinkNet capable of transferring high-level features to the decoder. A feature based loss function based on dual encoder and recursive residual pooling unit is proposed to assist the model during learning. See Chapter 7.

1.8 Thesis Organisation

The first part of this thesis presents the background to inpainting and highlights the concept of an encoder-decoder network, comparable cognitive reasoning of visual perception in humans in relation to image inpainting.

- Chapter 2 presents a comprehensive review of past and present image inpainting state of the art methods.
  - Section 2.1 introduces the chapter with an overview of image inpainting and how it started.
  - Section 2.2 presents traditional approaches to image inpainting represented in a hierarchical format illustrating different categories. These techniques are divided into five sub-categories, i.e. Exemplar-based texture synthesis, exemplar-based structure synthesis, Diffusion-based, Sparse representation and Hybrid methods each ending in conclusion with their limitations.
  - Section 2.3 presents deep learning approaches which includes convolutional neural networks and generative adversarial network methods to image inpainting.
  - Section 2.4 presents the datasets which is a key component to designing a generative neural network model. This section also provides a table showing an overview of the loss function and the dataset used for the state-of-the-art methods detailed in this thesis.

  - Section 2.5 presents the popular evaluation metrics used in image inpainting.
  - Section 2.6 presents the strengths and weaknesses of inpainting methods to provide new insights in the field.
  - Section 2.7 presents potential future works to address the challenges raised during research. Some but not all of these findings led to the proposed solutions presented in this thesis.
  - Section 2.8 is the summary of this chapter.

- Chapter 3 presents the research techniques and intuition that led to the development of the proposed methods presented in this thesis..

- Chapter 4 presents the Symmetric skip connection Wasserstein GAN for High resolution facial inpainting (SWGAN). This method a high resolution image inpainting method that uses a new combination Wasserstein-Perceptual loss function with colour preservation to optimise the end-to-end network.

  - Section 4.1 introduces the chapter and provides an overview of high resolution image inpainting algorithm.
  - Section 4.2 presents the connection to the relevant literature that inspired the work presented in this chapter.
  - Section 4.3 presents the architecture for the proposed inpainting solver and provides details of the design with reasons that led to the achieved model. Additionally this section describes the experimental steps and the loss functions used by the model during learning. Furthermore, quantitative measures show that our proposed SWGAN achieves the best Structure Similarity Index Measure (SSIM) of 0.94.
  - Section 4.5 presents the state of the art baseline comparison methods and the qualitative and quantitative evaluations from our model compared to state of the art.
  - Section 4.6 presents the experiments conducted to demonstrate the effectiveness of each component of the proposed design.
  - Section 4.7 is the discussion of the proposed method, the observation and findings to open the gap for an improved method.

  - Section 4.8 is the summary of this chapter and how it is related to the next chapter.

- Chapter 5 presents the RMNet, a perceptual adversarial network for facial image inpainting. This is a novel approach proposed that uses a reverse mask operator to address the limitation associated with blending missing pixels with the visible ones.

  - Section 5.1 introduces the chapter and provides an overview of the inpainting algorithm proposed in this chapter.
  - Section 5.2 connects the literature and techniques that inspired the design of the RMNet.
  - Section 5.3 presents the RMNet architecture and describes how the reverse mask operator transfers the reverse masked image to the end of the encoder-decoder network leaving only valid pixels to be inpainted. Additionally, this section introduces the new loss function computed in feature space to target only valid pixels combined with adversarial training. Furthermore, the reverse mask mechanism is compared to the state-of-the-art [124, 227] to demonstrate differences in novelty.
  - Section 5.4 describes the implementation and the parameters used during the experiments.
  - Section 5.5 presents the based comparison methods and displays the qualitative and quantitative results of the model.
  - Section 5.6 discusses our findings based on observation from the design stage to the experiments and results of the model proposed in this chapter.
  - Section 5.7 is the summary of this chapter and how it is related to the next chapter.

- Chapter 6 presents Foreground-guided Facial Inpainting with fidelity Preservation (FGAN), a facial inpainting model designed to preserve fidelity. It is a step towards preserving realism of facial features, though a very challenging task due to the subtle texture in key facial features that are not easy to predict.

  - Section 6.1 introduces the method with insights of what led to the proposed solution of this chapter.

  - Section 6.2 is a connection of relevant literature that links this chapter to the inpainting techniques that inspired the design of the proposed solution.
  - Section 6.3 shows the design of our model's architecture and outlines a detailed implementation. Included in this section is the newly proposed foreground loss, implemented to optimize our model.
  - Section 6.4 introduces the loss functions used during training.
  - Section 6.5 discuses the datasets used and provides details of the experiments. Also included in this Section 6.5, are comparable descriptions of our method with the state-of-the-art [152, 124, 227], followed by the model parameters during training and the quantitative evaluation.
  - Section 6.6 shows qualitative and quantitative results based on the inpainted regions. Also, more details on our findings and discussion of our model compared to the methods are highlighted in this section.
  - Section 6.7 presents our findings and discussion on the importance of preserving foreground features of the face on an image.
  - Section 6.8 is the summary of this chapter and how it is related to the next chapter.

- Chapter 7 presents a facial inpainting model designed to highlight high-level features for high quality predictions. It is a step towards generating realistic facial features, by introducing recursive maxpooling units connected with residual convolutions to merge two features from different encoders based on the morphological concept to erode low-level features, thus highlighting high-level features within the transition layer.

  - Section 7.1 introduces the chapter and provides reasons that led to the design of the proposed V-LinkNet model.
  - Section 7.2 links the chapter to related work that inspired the design of the model proposed in this chapter.
  - Section 7.3 shows the design of the full V-LinkNet model. The components of the model are explained in detail within this section.
  - Section 7.4 introduces the losses used to optimise the model.
  - Section 7.5 discusses the implementation and experiment parameters. This section also introduces the datasets as a standardised protocol for the evaluation of image inpainting algorithms.

  - Section 7.6 shows the qualitative and quantitative outcomes based on standardised protocol compared to the state of the art.
  - Section 7.7 presents the findings of different components of the model presented in this chapter and displays quantitative evaluations on the standardized protocol to demonstrate disparities in inpainting results.
  - Section 7.8 is a discussion section based on the proposed solution.
  - Section 7.9 is the summary of this chapter.

- Chapter 8 presents our findings, future works and conclusion for this thesis.

1.9 Summary of Publications

This thesis is based on the information included in the following publications:
The contents of Chapter 2 appears in:

- [PUB1]: A comprehensive review of past and present image inpainting methods: Jam, J., Kendrick, C., Walker, K., Drouard, V., Hsu, J.G.S. and Yap, M.H., 2020. A comprehensive review of past and present image inpainting methods. Computer Vision and Image Understanding, p.103147.

The content of Chapter 4 appears in:

- [PUB2]: Symmetric Skip Connection Wasserstein GAN for High-resolution Facial Image Inpainting: Jam, J.; Kendrick, C.; Drouard, V.; Walker, K.; Hsu, G. and Yap, M. (2021). Symmetric Skip Connection Wasserstein GAN for High-resolution Facial Image Inpainting. In Proceedings of the 16th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 4: VISAPP, ISBN 978-989-758-488-6; ISSN 2184-4321, pages 35-44. DOI: 10.5220/0010188700350044

The content of Chapter 5 appears in:

- [PUB3]: RMNet: A Perceptual Adversarial Network for Image Inpainting: Jam, J., Kendrick, C., Drouard, V., Walker, K., Hsu, G.S. and Yap, M.H., 2021. R-mnet: A perceptual adversarial network for image inpainting. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (pp. 2714-2723).

The content of Chapter 6 appears in:

- [PUB4]: Foreground-guided Facial Inpainting with Fidelity Preservation: Jam J., Kendrick C., Drouard V., Walker K., Yap M.H. (2021) Foreground-Guided Facial Inpainting with Fidelity Preservation. In: Tsapatsoulis N., Panayides A., Theocharides T., Lanitis A., Pattichis C., Vento M. (eds) Computer Analysis of Images and Patterns. CAIP 2021. Lecture Notes in Computer Science, vol 13053. Springer, Cham.

Open Source Contribution: The code and experiment developed for models on this thesis can be found in the following GitHub repositories:

- RMNet: A keras implementation of RMNet presented in Chapter 5.

Chapter 2

Literature Review

This chapter provides background information for this thesis. It presents image methods for both past and present techniques, as well as a summary of their strengths and limitations. It also categorises past and present techniques in a hierarchical structure for clarity, and it outlines the challenges with recommended guidance for future improvement and development. In subsequent chapters, a related context highlights the base literature relevant to the proposed algorithm. In subsequent chapters, we include a related context to form the base literature relevant to the proposed algorithm. This chapter appears in Computer Vision and Image Understanding Volume 203, February 2021 as "A comprehensive review of past and present image inpainting methods".

2.1 Introduction

Image inpainting originated from an ancient technique performed by artists to restore damaged paintings or photographs with small defects such as scratches, cracks, dust and spots to maintain its quality to as close to the original as possible. Figure 2.1 shows inpainting performed by hand.

The evolution of computers in the 20th century, its frequent daily use and the development of digital tools with image manipulation capability, has encouraged users to appreciate image editing, e.g. restoration, and the application of on-screen visual display and special effects to images. As a result image inpainting (henceforth inpainting) has become a state-of-the-art restoration technique. In a computer vision and graphics context, inpainting is a method that interpolates neighbouring pixels to reconstruct damaged, or defective, portions of an image without any noticeable

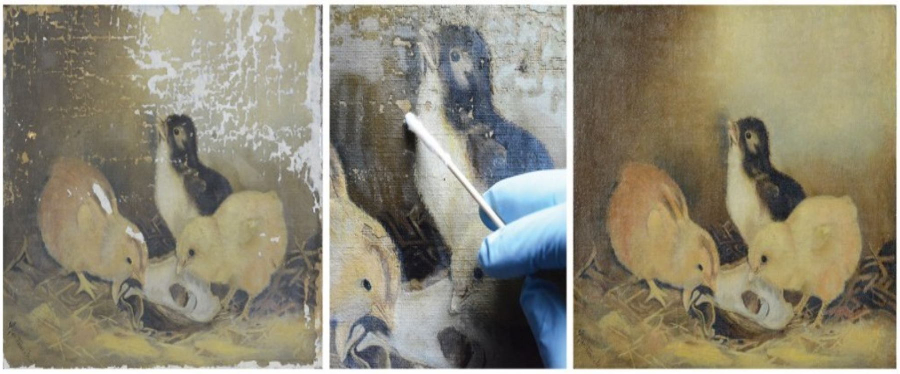
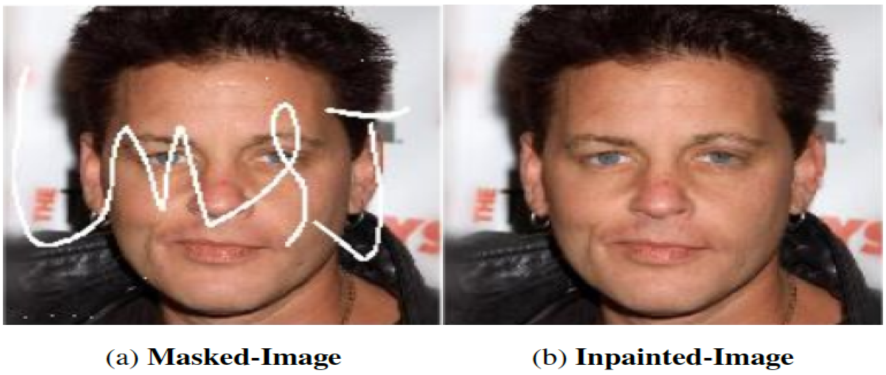


Figure 2.1: Hand inpainting performed by an artist. Image courtesy of [192].

change on the restored regions when visually compared with the rest of the image. These damaged portions/areas of an image are a set of unconnected pixels surrounded by a set of known adjacent pixels. During the reconstruction of disconnected pixels, the inpainting method uses known-information to fill unknown regions (disconnected pixels).

In this regard, Efros and Bertalmio are considered the pioneers [150, 19, 199] in this field and for advancing the research in texture synthesis and pixel interpolation respectively. In 1999, Efros et al. [50] proposed an advanced computational interpolation of pixels using Markov modelling. This novel concept is based on self-similarity to estimate a pixel value at the centre of a patch to synthesis a texture. This approach using image patches for texture synthesis, has largely influenced the success in developing image processing algorithms [23, 189, 25]. The method is a non-parametric approach to image synthesis, using an exemplar image as a source, and where pixel values are selected one pixel at a time. In this process, the chosen pixel merges and blends-in with the neighbourhood of the already synthesised output image.

In 2000, Bertalmio et al. [17] pioneered the introduction of a novel geometry-attentive approach for the interpolation of pixels on images. This novel method is based on Partial Differential Equations (PDE) and diffusion as a technique to propagate local features from surrounding regions into the damaged areas. PDE use isophotes (level lines with the same intensity on the surrounding area), e.g. Figure 2.2 shows the use of PDE for inpainting. The white mask regions denote the part or region to be filled-in, and the rest of the image is the source of propagated features. However, this technique is limited to small masks (unknown) regions. Then, in 2001, Efros et al. [49] introduced a stitching technique, known as quilting, that synthesised a smaller patch of an image to a more substantial textured outcome of the same texture and



(a) **Masked-Image** (b) **Inpainted-Image**

Figure 2.2: Inpainting on CelebA-Dataset image using the traditional method of inpainting by Bertalmio et al. [17]. Techniques under traditional inpainting methods are limited in terms of mask size, accuracy and sometimes efficiency.

structure of the initial image. This technique performs texture transfer on an initial seed (texture) through different stages, as shown in Figure 2.3.

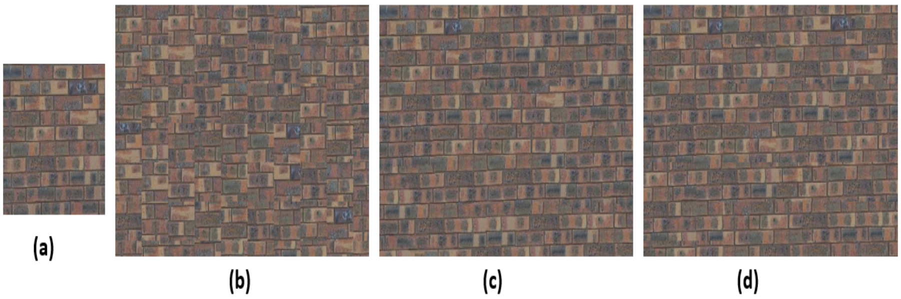


Figure 2.3: Image synthesis by Efros et al. [50] (a) Original texture (b) Synthesized texture from random patches. (c) Results of pixel-difference computed by the Sum of Square Differences (SSD) (d) A fully synthesised texture output with seam carving restoring similar visual appearance. Image courtesy of [107].

These pioneering works [50, 17], then considered to be state-of-the-art, caught the attention of the community [209, 21, 177, 92, 46, 40, 180, 36, 246, 109, 215, 64, 108, 26, 41, 72, 14, 4, 1, 52, 51, 186, 127, 93, 144] to further the research of these, “traditional” inpainting methods. Although these methods are reviewed in other literature, e.g. by Guillemot et al. [66] and Qureshi et al. [155], their scope is limited to traditional methods only. Yet, despite the advancements of these methods in the last decade, inpainting continues to remain a very challenging problem in computer vision. The purpose of this review is to bridge the gap in the previous literature [66, 155] and to include traditional and deep learning methods for the state-of-the-art algorithms. It should be noted that the few techniques reviewed here are just a handful of selected methods from inpainting techniques already in use. These methods are selected and summarised under different categories to illustrate the transition from traditional to the now state-of-the-art deep learning methods. Included as a summary under each category are the challenges and limitations of these techniques. Furthermore, the significance of datasets is considered, which include commonly used performance metrics as well as performance evaluations useful for inpainting methods. Figure 2.4 shows a hierarchical representation of the various categories of inpainting in their respective groups.

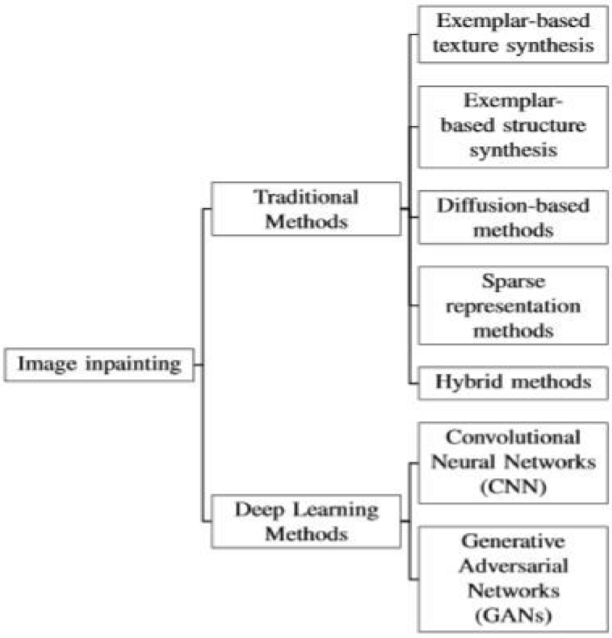


Image inpainting
Traditional Methods
Exemplar-based texture synthesis
Exemplar-based structure synthesis
Diffusion-based methods
Sparse representation methods
Hybrid methods
Deep Learning Methods
Convolutional Neural Networks (CNN)
Generative Adversarial Networks (GANs)

Figure 2.4: Hierarchical representation of image inpainting techniques in two main categories: Traditional Methods (the past) and Deep Learning Methods (the present). There are three sub-categories for Traditional Methods: Exemplar-based texture synthesis, Exemplar-based structure synthesis Diffusion-based, Sparse representation and Hybrid methods. Deep Learning Methods are sub-categorised into CNNs and GANs.

2.2 Traditional Image Inpainting Techniques

Since the evolution of digital technology, computer vision has experienced enormous research in transformations on images such as image-stitching [188], morphing [60], image swapping [30], registration [140], denoising [23] and inpainting [50]. Image inpainting experienced enormous amounts of research with considerable attention in the last few years, as researchers try to develop algorithms that are robust with less computational complexity. Various optimisation techniques are proposed to enhance the capability of these algorithms to handle more complex image structures. Because images are a visual representation in texture and structure, the image properties (patterns, corners, edges and changes in brightness) affect the performance of an inpainting algorithm. To understand the concept of inpainting in its fullness, texture and structure are defined in terms of image composition.

A **texture** is a visual pattern on an infinite 2-D plane with a stationary distribution at some scale [50]. This pattern refers to the feel (smooth, rough) of the image surface. Textures are either regular (repeated texels) or stochastic (imprecise texels) and can be synthesised based on the assumption that the sample is large and uniform with known statistics of regular patterns [156]. A geometric texture of an image is the entire representation as a texture based on statistical details of which a small patch is sufficiently a representative [110]. In textural inpainting, the available data considered for the inpainting task are exemplar textures. Textural inpainting uses statistical knowledge of patterns due to its stationary distribution of missing regions and known parts of the image, commonly modelled by Markov Random Fields (MRF) [50].

The **structure** of an image is a visual object constructed by distinct parts (global contour information) of the image texture [18]. The geometric structure of an image is a representation of composition and structure. During inpainting, the geometric structure has a low dimensionality representation in subspace. That is, the coordinates of the inpainted region are exact representations of the subspace and do not exceed its dimension. This is because it must satisfy the coordinate vertices of the image representation before decomposition to yield an approximate representation of the parent structure. With this technique, the target region does not exceed the parent structure, and the outcome is a good representation of the global context. In structural inpainting, taking account the nature of the smoothness in the missing regions and the boundary conditions is a precondition and which uses either isotropic diffusion or anisotropic diffusion to propagate boundary data in the isotropic direction [17]. The main categories of traditional methods of inpainting put forward in this review are as follows:

- Exemplar-based texture synthesis methods
- Exemplar-based structure synthesis methods
- Diffusion-based methods
- Sparse representation methods
- Hybrid methods

2.2.1 Exemplar-based texture synthesis

Exemplar-based texture synthesis methods are based on distance measuring tools and aim to generate new visually similar texture images from an input source whilst not being an exact copy of the input.

As already mentioned in Section 2.1, the Efros et al. [50] method laid the groundwork for exemplar-based texture synthesis. This method uses MRF modelling to locate pixel distribution and a new texture is formed in the unknown target region by querying existing texture to find blocks of similar pixels. This modelling process captures all neighbouring pixels to grow a new texture by synthesising the initial seed one pixel at a time. The process is iterative and uses a patch with known pixel values from a known patch of the previous step. The limitations are discontinuities, unwanted growings that do not respect statistics of a texture, thus causing the texture not to have uniformity.

Efros et al. [49] also used samples from a textured seed to form a similar patch with different dimension via quilting. This technique densely samples square patches from the initial seed, to assemble a single row of pixels in an order that forms the final image. To achieve this, the next patch to quilt into the image comes from a set of candidate patches. This method uses SSD to compute scores between the patches obtained from the patch overlap region to the left and above the patch in the quilted image. The limitation of this method is the enforcement of random patch selection, which may misalign with the rest of the texture and eventually form cascades on subsequent patch alignments. Also, due to disruptive coarse textures, smoothness is not often achieved because the patch size does not always complement the texture coarseness.

Le Meur et al. [112] used non-parametric patch sampling [50] to synthesize a coarse version as an input low resolution image for inpainting. The proposed solution is to use K-Nearest Neighbour (KNN), K-coherence candidate SSD and the Battacharya distance [24] metrics for priority selection of matches at different scales across a multi-resolution of selected patterns from the input low-resolution image. This allows the inpainting process to be less sensitive to noise and work with more enhanced oriented structures in the image. The authors use KNN to perform inpainting at coarse level, and apply single-image super-resolution to recover high-frequency details of the missing area from the inpainted low-resolution image. Therefore this technique reduces noise sensitivity and computational complexity allowing the algorithm to focus on the extraction of dominant orientations of textural image structures. This method can handle inpainting by filling in missing areas using different parameter settings to influence the patch size to better handle textures. A limitation is the speed during inpainting and quality of the resultant image, which highly depends on the selection approach for high-resolution patches from the dictionary based on the user (parameter setting).

In summary, exemplar-based texture synthesis methods are well-known to produce similar texture as the resultant image. These methods also preserve local textures and are capable of synthesising discontinuous textures. Exemplar-based texture synthesis exploit image statistics and assign a priority pixel based on surrounding similarity to inpaint a region on an image. Exemplar-based texture synthesis can produce a texture with perceptual similarity to the input sample. Methods in this category can synthesis small textures and rearrange neighbouring pixels in a consistent way, however, they can grow meaningless textures and verbatim copies. Another limitation of these methods is that the synthesised texture can have an unnatural feel due to limited samples in the data region caused by limited pattern similarity. Also noted is its inability to yield high quality results for highly structured examples.

2.2.2 Exemplar-based structure synthesis

Methods use statistical constraints to sample texture patch-wise instead of pixel-by-pixel, thus resulting in faster synthesis and realistic results with regular patterns. Under this category, image inpainting is structure consistent with the resultant image generated randomly with statistical constraints. Lou et al. [135] described exemplar-based structural synthesis as a search in the source image for a cluster of similar pixel patches to fill up missing pixels.

Criminisi et al. [40] proposed a filling order approach that is isophote-oriented. This technique is dependent on priority filling order of pixel values on structure continuation which favours the action of filling joints in the direction of incoming structures. The process starts by assigning pixels at the edge of the target region as priority pixels. Texture synthesis is performed during stage two of the process by replicating information from a source region in blocks based on the priority value that was initially assigned to each central pixel. Figure 2.5 illustrates the propagation of linear structures during the inpainting process from (a) to (d). Experimental observations shows adaptations to changes in structure due to its isophote preserving properties while propagating best match. The algorithm can handle both texture and structure during inpainting. However, it cannot handle curved structures, and its high dependence on priority pixel value may cause accidental priority pixel dropping, yielding visual inconsistencies on the inpainted region.

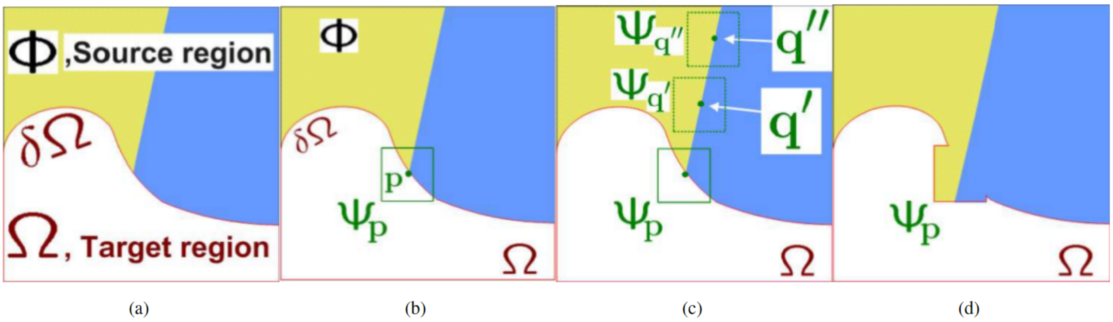


Figure 2.5: Image inpainting process in Exemplar-based inpainting task adopted from [40]. (**a**) The original image with contours showing source and target regions. (**b**) shows the chosen filled patch based on high pixel priority. (**c**) shows the most likely candidates for filling the patch. (**d**) shows best matching patch selected from candidate patch and copied to its occupied position.

Barnes et al. [13] motivated by Ashikhim [10], Sun et al. [187] and Simakov et al. [182], used Nearest Neighbour Fields (NNF) and proposed a fast randomised algorithm (PatchMatch). This method finds patches via random sampling with the help of prior information of random fields using NNF defined on a possible patch location. The process is iterative, and uses pixel propagation based on natural coherence where the randomly selected candidate uses adjacent pixels to improve a nearest neighbour search for new candidates. The advantage of this method offers is the fine-scale control to output pixels with the desired colour and restores both structure and texture simultaneously. It avoids dense computation in finding patch similarity because it applies arbitrary distance metrics which enabled local interactions as a constraint for completion. Furthermore, it performs well on images with texture and structure and large missing regions but limited in performance as it incurs additional memory overhead to store current best distance.

Ružić et al. [169] proposed to use textural descriptors to model and facilitate the search for candidate patches. This method splits the image according to context, thereby restricting the search for candidate patches to matching context. With the use of MRF as a prior to encode knowledge about consistency of neighbouring patches, the selection of candidate patches is accelerated based on contextual features with improved performance. This technique adaptively selects patches with more than half of the missing pixels in top-down procedure based on homogeneity in context. Thus, applying this technique, is an advantage because fixed patch sizes can be used even when missing pixels are not dominant. However, despite the improved performance it still faces challenges when shifting patches into unknown regions on images with complex scenes due to failures in translations associated with MRF-based methods [218].

Wang et al. [198] used a space varying update strategy powered by Fast Fourier transform for full image search. The base technique uses a standard deviation-based patch matching criterion and confidence term, that evaluates the spatial distribution of patches to measure the amount of reliable information surrounding the priority point against a known-priority point. This technique reduces the fast dropping effect seen in [40] and takes the distribution of patch differences into account. By eliminating fast priority estimation, a full search is achieved for better and agile matching results leading to improved visual consistency. The fill-in approach propagates linear structures surrounding the damaged area into the hole. These linear structures impose image constraints that influence the performance of exemplar-based texture synthesis efficiently and qualitatively. Also, this algorithm imposes a practical matching criterion of the region and priority function with high confidence in pixel and structure information. This method performs well in inpainting, but experiences discontinuities based on the coefficient value applied to adjust the weight on the standard deviation during the inpainting task.

Liu et al. [127] proposed to use multi-resolution for priority patch selection on high resolution images to complete an inpainting task. This method [127] uses similar patch selection to compute multiple candidate patches based on colour, gradients and boundaries. In this technique, the authors assumed that high-resolution imges are susceptible to high-frequency information (complex textures and noise) when extracting information around edges. The technique uses Structure Similarity Index Measure (SSIM) to select reasonable candidate patches and graph cut technique when filling the target region with the selected patch. To select a suitable candidate patch, the SSIM value of the known region is calculated to aid in selecting the best candidate patch. Using graph cut technolgy introduces smoothing, thus eliminating blockiness associated on the inpainted regions. The use of colour, gradients and boundary terms blends patchs well with improved inpainting effect. However, to obtain best results, the algorithm relies on the sample patch size to be manually selected, which may vary from user to user, thus may yield poor results.

In summary, methods in this exemplar-based structure synthesis category use similar patches from a known neighbourhood to recover the texture and structure of a missing region. This is based on learned pixel similarity by sampling texture sample from known parts of the image. The source patch is consistent with the geometric structure to fill in targeted regions during an inpainting task [25]. Furthermore, methods in this category overwrite missing pixels with corresponding pixels from patch to shrink the hole and update priorities. Also, limitations associated with speed, accuracy of texture (meaningless growing), and accurate propagation of linear structures are handled. Some limitations are lack of reasonable results when attempting to synthesis regions with no existence of similar pixels. Other limitations noted are failures in curved structures and depth ambiguity.

2.2.3 Diffusion-based methods

Methods in this category propagate image content smoothly from boundary regions into the interior of the missing region.

Bertalmio et al. [17] proposed to use isophotes and diffusion process (Laplacian) to propagate pixels automatically. This enables a simultaneous fill of missing regions in any direction with no limitation of the region to be inpainted. To achieve this, a smoothness estimator is introduced during computation and the propagated information is along the isophotes direction. Also, a time-varying estimator along the isophote direction determines the spatial change based on a discretization gradient vector. This algorithm is efficient on images with small cracks due to anisotropic diffusion, but leads to a blurring effect with slightly bigger mask regions. It is limited in the reconstruction of large textured regions or images with multiple damaged areas.

Oliveira et al. [160] introduced a fast approach that uses a diffusion kernel (gaussian) with considerations based on the tolerance of blur areas by human vision on regions with high contrast edges. The algorithm is mainly user specified as it involves repeated convolutions with a gaussian. This process involves computing a weighted average of neighbouring pixels when convolving an image with the gaussian kernel, which is equivalent to isotropic diffusion. That is, using a linear heat equation (isotropic diffusion) as diffusion barriers (two pixel-wide line segment) to handle edge re-connection. However, the results introduce some blur without user-entered diffusion barriers due to the low pass linear filtering suppressing high frequencies. Also, if the mask used is not exact on the region to be inpainted, false information propagates into the inpainted area. This algorithm only works well on filling locally small missing areas.

Tschumperlé [194] proposed a trace-based PDE as a regularization technique on multi-valued (multiple colour channels) images. This method is tensor-driven PDE based on heat flow constraints on integral curves to preserve curvature on images. By using this technique, isophote diffusion is minimised in all directions, yielding a low pass filter that suppresses high frequencies on the image. This process allows smoothing with edge preservation on curved edges. The capability of PDE to control geometrical features usually supports variational principles [54, 28]. Hence the use of the gaussian kernel on tensors to define orientation and strength of diffusion will fail to preserve curved structures. This method shows high performance on images with narrow damaged regions and small occlusions. The poor performance of this method is observed when filling large areas as it often results to image blur if the target region is large.

Daribo et al. [42] motivated by Criminisi et al. [40], proposed to use a depth aided texture (depth map) that considers background pixels as high priority over foreground pixels on an image during an inpainting task. This method [42] uses the depth map for image-based rendering to fill holes caused by disocclusion. Also, the distance measure of the depth minimises patch search with the same depth level. The depth map is smooth and disoccluded regions are easily verified. A smoothing strategy on the depth map correct deformities in disoccluded regions by correcting edges. This solves the problem of disocclusion occurrences, edge inconsistencies and overly smoothness during the process. However, this smoothing process is adaptive according to surrounding scene structures. Local smoothing of depth maps and edge correction are simultaneous during the inpainting process. Due to texture-less nature of depth maps, the assumption of a "virtual" image plane is where the depth map projects to perform the hole filling. This inpainting algorithm performs well on video inpainting tasks. However, this method lacks spatial and temporal stability and sometimes lead to more errors on the foreground depth map. This algorithm requires an iterative implementation of numerical methods, which has slow rendering speed, thus making it less robust on still images.

Le Meur et al. [111] used exemplar-based with PDE technique to compute patch priority on structure tensors to fill in missing regions. The main objective to employ PDE is to propagate information in in the direction of isophote lines to continue geometric and photometric information as described by Bertalmio et al. [17]. Li et al. [119] used PDE combined with smoothness constraints as regularisation technique to propagate local information. These constraints force the algorithm to follow directions given by local structure and are regularised iteratively, thus resulting in a sequence of continuous smooth image. Hence information showing a local pixel on an image contour propagates smoothness along contour direction and not across boundaries, thus addressing the limitations of previous methods [17, 16, 18]. Pixels located on uniform surfaces will spread smoothness in all direction. Based on this finding, using isophotes (line of constant intensity) alongside PDEs leads to an inpainted image with a continuous evolution in structure. It has shown great success on textured images with scratches/smaller gaps but poor results on large gap regions. The disadvantage of these methods is computational cost due to slow or prolonged arrival of structures at border regions.

Sridevi et al. [186] proposed to use fractional-order derivative (integer-order derivative) with Discrete Fourier Transform (DFT) for inpainting task. The authors [186] used this method to achieve a good trade-off between the restored region and edge preservation, and also because DFT are easy to implement. Using fractional order derivative, pixel level on the whole image is considered instead of just considering neighbouring pixel values. The technique employs fractional-order nonlinear diffusion model (difference curvature driven [34]) to handle gap regions and fractional-order variational model for denoising and de-blurring of the image. The advantage of this method is its ability to preserve edges during an image restoration task. Also, it ies effective in eliminating noise and blur without affecting the edges. The disadvantage of this model is that it relies on user interaction for manual selection of fractional order, which may lead to poor results on the inpainted region.

In summary, methods in this category push good pixels from boundary into the gap region, filling in altered pixels to set a colour similar to or the same as the source region. They are suitable for inpainting with scratches, straight lines curves and edges. However they turn to blur large textured regions due to prolonged arrival of pixels to fill in gap regions. Therefore it is not well suited for textured images with large gaps/regions. Also, the iterative implementation of numerical methods that will eventually render it slow. Therefore this algorithm is not robust on still images.

2.2.4 Sparse representation methods

The sparse-based methods assume that images contain natural signals that admit a sparse decomposition over a redundant dictionary leading to efficient algorithms that can handle sources of such data [141].

Inspired by Shih et al. [181], Chang et al. [29] proposed to use colour space in facial images to correct facial images which are overly-exposed in digital photography. This method uses multi-resolution [181], to consider level details in a suitable colour space for layer separation and fusion layer during an inpainting task. This process exploits the characteristics and level-by-level features of the image and segments the skin region on facial image. The multi-resolution technique uses mean colour (average of a group of pixel colours) and neighbouring pixels to perform an inpainting task based on the percentage of damaged pixels. A sliding window, based technique is utilised to select bright spots (reflection artefacts) on the face. Based on this method, there is evidence that a considerable pixel variation contains detailed shapes of an image which supports the claim by Shih et al. [181], for a multi-resolution inpainting strategy. However, this algorithm is highly effective on facial images with high accuracy, but cannot generalize well on other images. It is used to correct facial images with too much light exposure in digital photography. Although, this algorithm is highly effective and accurate on facial images, it has not been tested on natural scene images.

Kawai et al. [101] proposed an approach that examines two sample textures by considering the change in brightness and spatial locality of texture patterns. Energy minimisation is the key technique to this method because initial values are assigned to missing regions and targeted regions completed by minimising the energy function. This technique also checks the data region for image patterns, and uses a sliding window to match fixed pixels on that region. The preceding step uses a central pixel of the window overlapping the expanded area and the missing region fills-in with the reference pixel inwardly during inpainting. The energy function is the weighted SSD representing the pattern similarity in conjunction with the change in brightness and the similarity difference representing the spatial locality. This method allows a change in intensity with spatial locality as a constraint during inpainting, but shows poor performance when single weight coefficient is used.

Shen et al. [175] proposed to use the sparse representation of image signals over a redundant dictionary with the assumption that the image is thinly distributed on the basis of wavelets. This method relies on discrete cosine transform to build a redundant dictionary of patch observations. The inpainting task performs sequential computation iteratively over these sparse representations, completing every uncompleted patch at the boundary of the target region. In this method, the user is allowed to specify the area to be inpainted, which eliminates the problem of finding the corresponding input signal to the corrupted area of the known pixel specified by the user. The image is inpainted inwardly from the boundary of the targeted region with the priority pixel given the most probable chance. At each iteration, the pixel closest to the target filling region, has the maximum priority since the patch is the current iteration centred at the boundary of the target region. Overall, the algorithm recovers incomplete image signals with each signal corresponding to a patch, and the target region is filled based on the patches for each sparse patch representation.

He et al. [74] used a dual-phase algorithm (Thiele's rational interpolation function and Newton-Theile's function) for adaptive inpainting. This method uses continued fractions to update pixel intensity during the reconstruction of damaged portions based on the surrounding pixel information of known regions along the target region. That is, if the damaged pixel points are vertical, the selected points for interpolation of pixels are in the horizontal direction. The masked image is scanned line by line to locate and adopt information of known pixel points to perform interpolation of pixel intensity. The first phase repairs the damage and the second phase refines the restored image to closely resemble the original image. This second phase updates the intensity of all the damaged pixels in the reconstructed image by using the masked image to locate damaged positions originally corresponding to the previous damaged repaired regions. Whilst this method performs well, its limitation is that the damaged pixels need to be in the vertical direction.

In summary methods in this category assume that the known and unknown regions share similar sparse representation. Also, another assumption is the images are represented in sparse linear combination as a complete dictionary which can be adaptively updated to target inferred pixels. Methods in this category help to improve the visual quality of the image.

2.2.5 Hybrid methods

The continued success of exemplar-based texture synthesis and exemplar-based structure synthesis methods in inpainting tasks, has motivated researchers to explore the combined capabilities of these methods. Where some are suitable for small gaps, or structures with curvatures and edges, others work best with texture restoration. For this reason, combining two methods to handle images with composite structures and texture has become an area of great interest in inpainting.

Bertalmio et al. [18] used PDE to determine synthesis ordering [17] and texture synthesis by Efros et al. [50] to recover geometrical structures and small textured regions. This method decomposes an image into texture and structure layers for inpainting. After decomposition, the energy function for texture synthesis is applied to the texture layer and diffusion-based method for inpainting applied to the structure layer. This method breaks down the inpainting into two. It uses the diffusion based technique in Bertalmio et al. [17] in the structure layer and adds synthesised textures derived by using Efros et al. [50] method to the in-filled region. The energy function utilised in patch stitching minimises the seam area during the inpainting task. The energy function measures self-similarity, coherence and diffusion with each measure

Table 2.1: Analysis of the state-of-the-art traditional methods on image inpainting.

Category	**Advantage(s)**	**Disadvantage(s)**	**Prior(s)**	**Application(s)**
Exemplar-based texture synthesis[50, 49, 182, 112]	Preserves artefacts, No occurrence of blur. Preserves both structural and textural information, Used in wireless transmission for lost block retransmissions.	Failures in the reconstruction of large textured regions or images with multiple damaged areas. Can lead to repetitive patterns.	Smoothness.	Image restoration, editing, disocclusion. & concealment
Exemplar-based Structure synthesis[17, 13, 169, 127]	Performs better during inpainting of large textured regions. Inpainted task is copy/paste fashion with almost verbatim copies. Restores texture, structure and colour.	It is time consuming and exhaustive for methods in this category to generate different candidate patches. Can mix several verbatim copies with distinctive overlaps. Incurs additional memory overhead to store the current best distance.	Priority assignment. Best patch selection.	Image restoration, editing, disocclusion. & concealment.
Diffusion-based Methods [17, 16, 176, 177, 128, 66, 119]	It generates good results when filling in small or gap regions. Preserves edge information. Suitable for completing lines and curves. Does not produce verbatim copies in the synthesized textured region. Maintains the structure of the inpainted region.	Fails to inpaint large textured regions, resulting in blurry artefacts on image.	Smoothness	Image restoration
Sparse representation Methods [181, 141, 29, 101, 175, 74]	Inpaints facial images with high exposure to light. Allows change in light intensity.	May not work well on natural scene images. . On image reconstruction,the damages pixels must be in the vertical direction.	Colour space, self-similarity & sparsity.	Image restoration
Hybrid Inpainting[231, 66, 198]	Preserves edge and restores smoothness. Impressive results on the linear structure of the image improve the speed.	Computational complexity with no guarantee in convergence.	Smoothness, Similarity & Sparsity.	Image restoration, editing, disocclusion. & concealment

having a role during the corresponding mapping of pixels. The similarity between the patch of central filling pixel and the known pixel of the source region of the image is computed by self-similarity measure. A discrete Laplacian equation calculates the corresponding map of the image region to be inpainted for diffusion. This method overcomes the limitation by diffusion-based methods which causes overly smooth outcomes. The method is computationally expensive and fails in some cases of large missing structures.

Allène et al. [5] used a combination of variational [49]and statistical [17] methods to propose a graph-based approach based on the concept of progressive stitching. This technique uses MRF-based cost function to find similarity of the existing image and patches that are needed as well as the measure of the boundary to the inpainted region. The concept of progressive stitching consists of a selection of pixel values, corresponding to selected patches from possible image patches that best approximates the original data. The technique of Allène et al. [5] uses MRF-based cost function to find similarity of the existing image and patches that are needed as well as the measure of the boundary to the inpainted region. To achieve this, they introduced constraints to minimise the distance between chosen patches and existing data to capture properties of the area surrounding the missing content. With these constraints, multiple candidate patches are then superimposed over the target region, forging a multi-source label solvable by graphs to perform inpainting. However, because discontinuities are created during the stitching of selected patches, local smoothness on the final texture is not always achieved. This method also incurs a computational cost due to the particular attention required for selecting the right patches, to best approximate the corresponding pixels.

Zhang et al. [231] proposed an algorithm that decomposes an image into structure and texture using wavelet transform with the aim being to capture the image texture and structures without loss of information with wavelet transform. In structural propagation, the patches copied are specified by missing structures of the unknown regions and are in the same direction of similar curves in the known area. The use of curvature driven diffusion also applies to structural reconstruction, while exemplar-based texture synthesis is the fill-in process during textural reconstruction. The two inpainted components are combined into a plausible outcome with similarity compared to the input image. However, this method is computationally demanding for large fill regions.

Ghorai et al. [58] proposed to use patch selection and refinement method based on joint filtering alongside a modified MRF to enhance optimal patch assignment to perform an inpainting task. This technique uses subspace clustering to select target patches from boundary regions into groups, which are refined via joint patch filtering to capture patterns and remove artefacts. The selected patches are targeted in sequential order from the interior regions based on neighbouring patches from candidate patches along the boundary regions. The subgroup of similar patches are merged into larger groups, alongside ensuring deselection of any patch that is too different from the boundary patch. However, despite a faster patch selection in reduced search space compared to global patch selection by other methods, the limitation is in the cost of group formation and grouping of each target patches. Quantitative evaluations using PSNR showed the algorithm's best at 29.7db for regular-shaped mask and 24.23db for irregular-shaped mask.

In summary, methods in this category can remove text overlays, fill in regions with complex textures and structures. It is noted to handle discontinuity in boundary regions and blur, and produce images with local coherency in visual quality. Despite the excellent results, methods in this category still perform poorly in some disocclusion and object removal task. Also, there is no guarantee in convergence and they cannot be applied to error concealment applications due to time constraints (computationally expensive).

2.2.6 Summary

Some of the inpainting methods described above are summarised in Table 2.1. These have shown great success in nearest neighbour searching, i.e. using patches or pixels to synthesise images. However, the challenge to inpainting is maintaining a realistic structure and texture in the output image. For example, traditional methods attempt to fill in missing pixels using image patches from existing regions or use the diffusion mechanism to propagate pixels into a hole region from high pixel similarity areas. Whilst, these methods can propagate vivid textures for background-inpainting, they often failed to capture high-level semantics, yielding non-realistic images with repetitive patterns. Moreover, these methods do not yield plausible outcomes for inpainting tasks on complex mask regions such as a face or objects with non-repetitive structures. Generally, and despite their success in generating high-frequency seamless textures, they continue to fail in the generation of structures that are globally consistent. However, the advent of the generative neural network, inpainting algorithms can be taught to learn meaningful and high-level semantics which have proven to generate coherent structures for missing regions. This is discussed in the next section.

2.3 Deep Learning methods

In more recent research, the use of Convolutional Neural Network (CNN) [56, 116, 196] and Generative Adversarial Network (GAN) [152, 217, 223] have become the state-of-the-art methods used to perform image inpainting task. These methods use CNN as a feature extraction method through the process of convolution to capture abstractions. The use of CNN combined with adversarial training [62] has produced excellent results on inpainting task, with perceptual similarity to the original image. It is advantageous to use CNN combined with GAN because a CNN has an encoder that is used as a feature extractor to capture high dimensional data abstractions; and a decoder that reconstructs these learnable features in an end-to-end fashion, while the GAN enhances the sharpness of the image [152].

The performance of algorithms in this category depends on the datasets used. Datasets are valuable in research as it consists of ground-truth images and information [161]. Different types of datasets have been used to train and test image inpainting algorithms. These datasets are made up of images with diverse textural and structural information, which mainly test the robustness of the algorithms to learn different image features.

2.3.1 Convolutional Neural Networks

Jain et al. [86] pioneered the use of CNN to inpainting, by framing the computational task within a statistical framework of regression rather than density estimation. The authors used an image denoising task which was formulated with parameter learning for back propagation. The image denoising thus becomes the learning problem in the CNN with noise integrated clean images during training. However, this method is restricted to greyscale (one colour channel) images, removal of "salt and pepper" noise, whilst also requiring substantial computation cost. The method of Jain et al [86] was improved by Xie et al. [212], who proposed combining sparse coding and deep neural networks as a denoising auto-encoder, to handle inpainting of inconsistent localities of corrupted pixels. The use of sparse coding and deep neural network overcame the limitation of computational cost, and eliminated noisy pixels supplied to the algorithm as the regions required for inpainting. However, the use of stacked sparse denoising auto-encoders strongly relying on supervised training and can only handle images with small denoising tasks, such as the reconstruction of images with controlled procedural pixel corruption.

2.3.2 Generative Adversarial Network

GANs refers to a two model framework of unsupervised learning algorithms that estimate adversarial processing. Inpainting methods using the GAN process aim at generating a conditional image for high-level recognition, based on low pixel synthesis formulated into a convolutional network (encoder-decoder). The trained adversarial network enhances coherency between generated and original pixels. For example, Figure 2.6 shows the GAN framework for estimating generative models via adversarial networks. GANs were first proposed by Goodfellow et al. [62], as a two model network; the generative and discriminative model. The generative model (generator) of the neural network captures data as a random input noise and transforms into a fake image, intending it to look like the real image from the training set. The discriminative model (discriminator) tries to distinguish this generated image from the training set, by estimating the probability that it came from the training set rather than from the generative model. Equation 2.1 shows optimisation of the loss during the combined training of the discriminator (D) and generator (G) network, where **z** is sampled from a prior distribution p_z and **x** is the sample from p_{data} distribution. G maps the random vector **z** and D discriminates between images generated from G and real images **x** sampled from p_z [6].

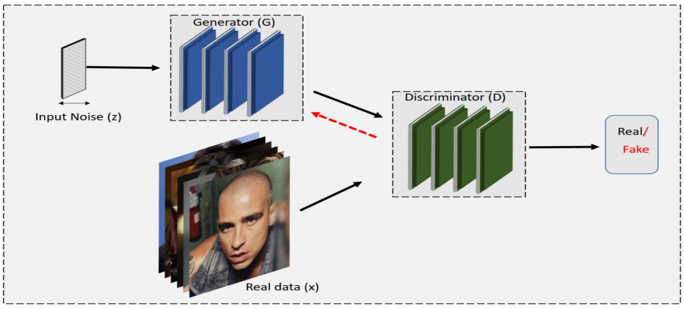


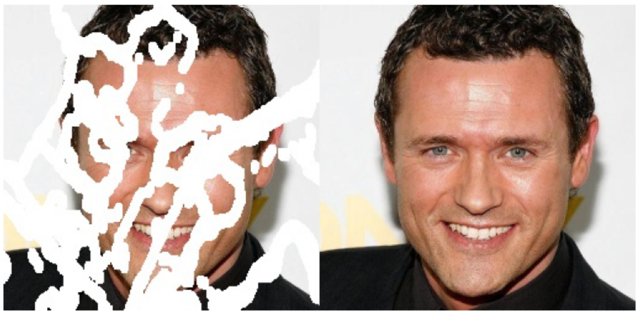
Figure 2.6: An example GAN block. The generator input (z) is sampled from a random noise vector and the Discriminator input (x) is sampled from real data distribution

$$\min_G \max_D V(D,G) = E_{x\sim P_{data}(x)}[log(D(x)]+ \\ E_{z\sim P_z(z)}[log(1 - D(G(z)))] \tag{2.1}$$

However, the difficulty during the training process increases uncertainty, and the generator can improve, leading to a vanishing gradient of the discriminator, thus making it difficult to converge. Figure 2.7 shows an inpainting by deep learning method.

For simplicity and readability deep learning based algorithms are categorised here based on how efficient each network is to the key properties necessary when considering designing a deep learning model. These are feature extraction, feature propagation and feature attention. Note that each model is assigned under a category where it performs best. Table 2.2 shows a summary of the deep learning methods, models, description, the loss function and datasets used for evaluation.

Feature Extraction methods, style transfer, Feature attention, and Feature propagation are all categories for the state of the art deep learning inpainting models.



(a) **Masked-Image** (b) **Inpainted-Image**

Figure 2.7: Inpainting task on CelebA-HQ Dataset [98] shows the performance of deep learning methods. The slightly thicker mask obtained from Nvidia Mask Dataset [124].

These are the many properties of inpainting, and each model has been assigned to a category that corresponds to the proposed method's main contribution.

Feature Extraction

Pathak et al. [152], pioneered adversarial training [62], an end-to-end network based on CNN [76, 15, 106], to predict the missing content of an arbitrary image region conditioned on its surroundings with realistic output. This technique captures semantic visual structures aided by $\boldsymbol{\ell}_2$ reconstruction and adversarial loss. The $\boldsymbol{\ell}_2$ loss is capture the overall structure of the missing region with regards to context and coherency, but tends to average the multiple modes in prediction. Introducing adversarial loss [62] enables the network to predict more realistically by picking particular patterns from the distribution. Overall, performance evaluations using PSNR show that adversarial training with reconstruction loss, yields higher PSNR value of 18.58 with [45] compared to 14.70 & 12.79 obtained by Hays et al. [70] model, suggesting a more accurate pixel inference of missing content to the entire image. It is, however, limited to small image sizes of low resolution due to training with regards to $\boldsymbol{\ell}_2$ loss. It also lacks spatial support with more substantial inputs, and often produces images with considerable amounts of implausible results that are overly-smooth (blurry) and which lack edge preservation. Furthermore, the discriminator focuses on the missing region and does not take into account the global context of the image. Thus, this method cannot guarantee structural cohesion nor a harmonious texture between the inpainted region and the image context.

Iizuka et al. [82], motivated by Pathak et al. [152], proposed the use of dilated convolutions [225], as part of the encoder-decoder, combined with two discriminators. Dilated convolutions increase the input area of each layer without loss of resolution or parameter accretion. The use of dilated convolutions increases the receptive fields for neurons at the output, thus replacing the channel-wise fully connected layer in [152]. The global discriminator for assessing the entire coherency of the reconstructed image and a local discriminator assesses the area of the completed region to ensure consistency within the entire image. The input to the global discriminator is a 256 × 256 resolution image, while the input to the local discriminator is 128 × 128 on the centre of the completed region. The authors used subjective evaluation reporting 77% approval rating on naturalness of inpainted images against the state-of-the-art [13, 152] and 96.5% for ground-truth images. However, this method fails to capture long-ranged textural information (limited to image completion with mask at border region) and relies heavily on post processing using fast matching [190] and Poisson blending [154].

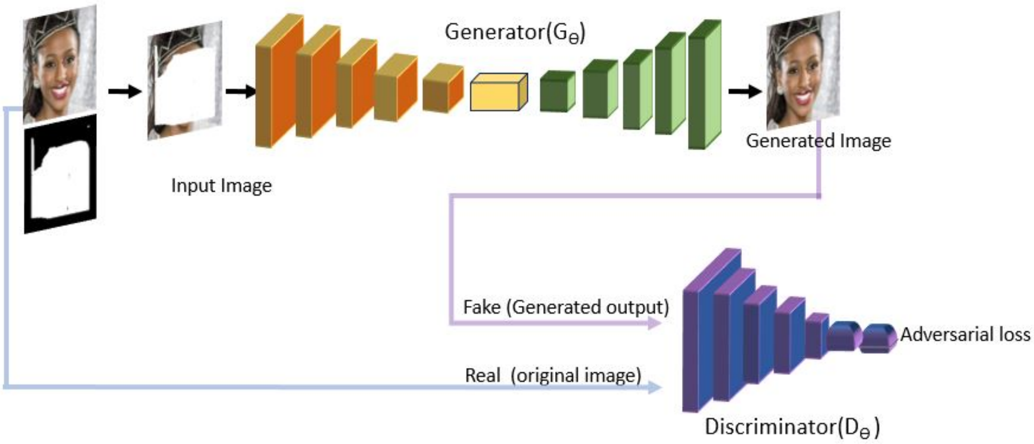


Figure 2.8: An overview of encoder-decoder architecture; a backbone of some inpainting networks. For example, an encoder-decoder framework used by [87]

In a different study, Yan et al. [216] used the U-Net [163] to introduce a shift-connection layer combined with guidance loss to inpaint images using deep feature rearrangement. The shift-connection layer handles images with sharp structures and fine texture details. The technique concatenates the encoder feature of the first convolutional layer to serve as an estimator of the missing parts on the last decoder layer after the fully connected layer. This approach uses a guidance loss function, ℓ_1 loss and adversarial loss to obtain photo-realistic textures. The guidance loss implemented based on the shift-connection layer, uses SSD on concatenated features of first convolutional layer of encoder and features of last convolutional layer of decoder. The end recovery used the encoded features to approximate the missing portion based on the ground-truth. Overall, the performance evaluation scored high values quantitatively and are shown in Table 2.4. However, this method may experience poor performance due to the parameter value of the guidance loss chosen to perform the shift operation. A limitation is that a smaller parameter value may lead to a more extensive feature map size that will increase computational time to 400ms per image. Also, a more substantial parameter value may lead to a more modest feature map, leading to a loss in image detail information. Thus, the best trade-off reported a computational time of 80ms per image, compared to 40ms per image which results to a generated image with less texture and coarse details. Although their shift-connection implementation of the U-Net structure has shown excellent results, it struggles in terms of efficiency and computational speed due to network parameters that do not make it suitable for most applications.

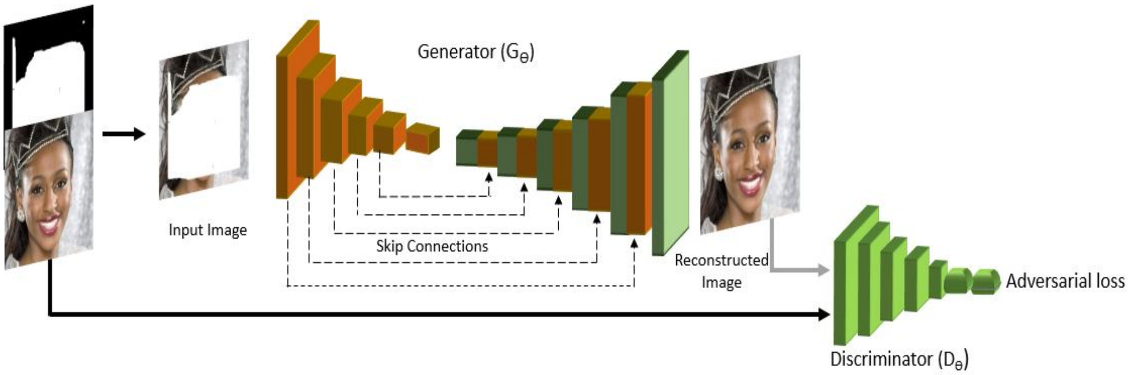


Figure 2.9: An overview of U-Net combined without any attention layer as described in some state-of-the-art methods. This U-Net is similar to [216] without the shift-connection layer.

Huang et al. [79], motivated by Goodfollow et al. [62], Mirza et al. [143] and Ronneberger et al. [162], studied the network structure of the encoder-decoder and introduced padding and pooling operations to avoid edge disappearance. The proposed completion network uses adversarial training with a new loss function based on SSIM and ℓ_2 loss. SSIM loss works as an authentication mechanism on the reconstructed image to improve the structure and texture. This method also introduced the use of a mini-batch discriminator to optimise training, thus increasing diversity of the generated sample. This loss enables photo-realistic images which are further judged by the adversarial network to obtain the output as close as possible to the original image. The in-house dataset dataset contains 2015 images for authentic street images by Huang et al.[79]. The images are 256×256 with a variety of situations such as foggy, rainy, day and night. The mask is rectangular, with various sizes randomly generated and applied to the image. The randomly generated binary masks are applied to the images and randomly shuffled with 80% used for training and 20% for testing. The data distribution is similar for both training, and test sets for the filling task handled in various situations. Both qualitative and quantitative evaluations carried out on the in-house dataset shows this algorithm performs with good results, scoring $\boldsymbol{\ell}_2$ (8.99), PSNR (39.63) and SSIM (0.97) values, compared to Pathak et al. [152] (11.02, 37.36 & 0.95).

Style Transfer

Yang et al. [217] used style transfer [94, 196, 116] to propose a multi-scale neural patch synthesis approach combined with adversarial loss. Yang et al. [217] used the context-encoder [152] to captures image content, texture and to preserve contextual structures, thus producing images with high-frequency detail. The style transfer network ensures the context-encoder predicts the global content with the local patch similarity of the predicted region. A texture network, pre-trained on image classification, takes the output image of the prediction network as an input. The architecture uses a local texture loss function computed VGG19 [183], pre-trained on ImageNet and a "holistic" loss based on $\boldsymbol{\ell}_2$. The joint loss function is $\boldsymbol{\ell}_2$-norm and the texture term (loss function) computed on extracted feature maps from feature layers (relu3 and relu4) of the VGG19 block. Quantitative results based on Paris Street View [45] shown in Table 2.4, show that inpainted images performed better compared to the state-of-the-art [152, 59]. However, the proposed algorithm suffers some limitations with the content and texture networks failing to guarantee correct image structure, blurry images whilst being computationally expensive with high-resolution images (taking longer to inpaint). Another limitation is the difficulty in hallucinating suitable texture for larger irregular mask regions since it is designed for rectangular holes.

Zheng et al. [239] proposed to use a two stage probabilistic distribution framework, combined with an attention layer (short+long term-based), both using GANs for image inpainting task. The first network uses a Variational Autoencoders-based model to reconstruct an image based on prior distribution of missing parts given the ground-truth. The second network uses a conditional completion coupled with information obtained from the first network to predict the missing regions based on the visible pixels. The short and long term layers are used to improve appearance and consistency by measuring the distance between related features of both encoders and decoders of the two networks. However, both frameworks sample from a probabilistic distribution of the masked image with ground-truth visible pixels, and the complement of the masked image with ground-truth missing regions (the reverse of the masked image). To achieve this, a conditional variational autoencoder is employed to estimate the parametric distribution in latent space, where sampling is possible. Therefore, a lower bound conditional log-likelihood is the probability of observed training data given the deep network parameters that generate the missing data. To optimise both networks, two L1-Norm based reconstruction loss of which one is geared towards reconstructing the entire image and the other focused on valid (visible) pixels combined with adversarial loss are employed. For evaluations, quantitative comparisons with the state-of-the-art Iizuka et al. [82] and Yu et al. [226] showed the model's superiority based on $\boldsymbol{\ell}_1$ (12.91), PSNR (20.10), Total Variational (TV) Loss (12.18). For IS [184], the model's performance rate was 24.90 conducted on 20000 test images from ImageNet [168] using 128×128 centre binary mask. However, the authors state that evaluations were carried out based on a selection of samples since the goal was not to achieve a single solution. Zhao et al. [238] used three network modules namely; a conditional encoder module, manifold projection module and generation module combined with cross semantic attention for image inpainting. The authors used a jointed probability distribution analysis to come up with a hypothesis to solve the inpainting task. That is, given that a set of reconstructed images generated from a set of masked-images is expressed as conditional probability distribution, then the set of masked-images is expressed as marginal probability distribution, then the training data is a joint probability distribution. This means that in an image inpainting task, finding the conditional probability distribution depends on the marginal probability distribution and joint probability distribution. Therefore borrowing information from the ground-truth (training data) by traversing an image completion space is in a sense using marginal and joint probability distribution to obtain conditional probability distribution. The architecture is a dual encoder network with different inputs, where one branch takes an instance image (ground-truth) and the other a masked-image to perform a one-to-one mapping in the same low-dimensional space in order to reconstruct an image. Within this network, are a set of instance images (ground-truth) corresponding to the masked-image used for guidance during training. The network uses cross-space translation to learn one-to-one mappings between the instance image and the masked-image. Therefore, the two spaces (instance and conditional completion) are associated in one latent space by one-to-one mapping, where the instance images corresponding to the mapped restored images have the same representation in low dimensional space. To optimise the network, a conditional constraint loss handles appearance and perceptual features extracted from VGG16 [94] using the $\boldsymbol{\ell}_1$ as base. Both appearance and feature loss use the instance and masked image expressed as a function of the network and the mask. Other losses used are the KL divergence, reconstruction and ongoing adversarial loss. The cross semantic attention layer uses 1×1 convolutions to transform feature maps obtained by instance and masked images to evaluate cross attention before adding them to feed the decoder. Comparative evaluations were carried out using the baseline models Pathak et al. [152], Yu et al. [226], Liu et al. [126], Ren et al. [159], Song et al. [185], Yan et al. [216], Sohn et al. [184], Zhu et al. [245] and Zheng et al. [239]. Quantitatively, the performance on 1000 CelebA-HQ images using centre mask of size 128×128 were better than the state-of-the-art and are shown in Table 2.4. The limitation of this network is that there is a possibility to suffer from mode collapse (i.e poor diversity in generated images) during training if trained in an unsupervised manner. Figure 2.8 is a typical encoder decoder network with ongoing adversarial loss for image inpainting.

Feature Attention

Yeh et al. [223] used the model architecture from Radford et al. [157] and introduced a spatial attention mechanism that searches for encodings of the corrupted image in latent space to recover the lost area based on the surrounding image features as reference; thus, with this encoding, the generator reconstructs the original image. This algorithm uses context loss (ℓ_1-norm) and adversarial loss information to search for closest encoding on the trained generative model regardless of the structure (mask) of the missing content. The context loss is a "weighted context" considered close to the corrupted pixel region while the prior loss corrects unrealistic images. Iterative optimisation of the objective function is through back-propagation in combination with prior and context losses. The algorithm scored high PSNR values (22.8, 33.0 & 18.9) compared to PSNR (20.6, 24.1 & 16.1) using Pathak et al. [152] across the various datasets for this method as shown in Table 2.2. However, despite excellent performance, the algorithm struggles with misalignment on images and some failures in finding "closest" encoding to the corrupted image in latent space. This may be as a result of the difficulty in training GANs, which can lead to poor data distribution capture increasing its inability to handle high resolution or complex scene images.

Yu et al. [226] introduced a coarse-to-fine network and used the contextual attention module by Iizuka et al. [82] to design their network. This model uses cosine similarity to learn the relationship between background and foreground feature patches. The contextual attention module is redesigned to use dilated convolutions and the model optimised using a reconstruction loss and two Wasserstein GAN losses by Arjovsky et al. [9] and Gulrajani et al. [67]. The two-stage model produces a roughly restored intermediate image with filled predictions and refines this result using a refinement network designed with dilated convolutions. The contextual attention module includes spatial propagation layers to encourage spatial coherency and fuse attention scores for more realistic outcomes. The contextual layers refine the image and alienate the idea of Poisson blending in [82]. The final reconstruction performing convolutions on foreground patches and background patches relies on the attention score for each pixel value, obtained using Softmax. These are then propagated channel-wise to reconstruct layer. The quantitative performance are reported on rectangular mask only and are shown in Table 2.4. However, despite great results, it also lacks fine textural details and inconsistencies with the background pixel-wise on high resolution images.

Zeng et al. [228] used the U-Net (Ronneberger et al. [163]) and proposed a network that learns high-level semantic features from region affinity to fill in missing regions in a pyramid fashion. The authors introduced a cross-layer attention and pyramid filling mechanisms in each layer, referred to as Attention Transfer Network (ATN) with each layer being derived from region affinity between patches. The ATN transfers relevant features outside missing regions and makes use of softmax and cosine similarity between patches inside/outside missing regions extracted from patches. With softmax applied, the attention scores obtained are used as the valid pixel to fill in the missing region. A multi-scale decoder using dilated convolutions at different rates acts as a refiner during the filling-in of missing regions. The loss function used is ℓ_1 loss combined with GAN loss [62] for realistic images. Masks sizes of 32×32, 64×64 and 128×128 are used for the evaluation of this method. Overall, and based on the qualitative evaluations by the authors, high quality images are obtained with smaller non-border size masks. Also included in the evaluation analysis are random mask used for visual comparison . The overall performance using non-border mask sizes of 128×128 show good results compared with the state of the art [13, 82, 124, 217] as shown in Table 2.4. Also, the authors do not show detailed results for images with border mask regions.

Zhou et al. [243] used the U-Net architecture to learn facial textures at multiple scales with help of seven discriminators. The proposed method uses a Dual Spatial Attention (DSA), that learns correlations between facial textures based on two inputs (masked-image and ground-truth image) to obtain attention scores for foreground and background pixels for reconstruction. The attention layer is applied to multiple layers within the decoder, with foreground attention scores from softmax layer, acting as direct supervision to the inpainted regions. Within this layer, the masked regions are the foreground and the unmasked region is the background. The DSA works has foreground-background cross-attention and foreground self-attention units within its module. The first unit uses the mask to segment the input feature into foreground and background features and uses 1×1 convolutions to rebuilt the original foreground features based on correlations with background features. The second unit is similar, without the foreground features taken into consideration. The attention maps are learned from the ground-truth to ensure high quality filling of missing regions during training. Four discriminators ensure realistic features of the left-eye, right-eye, nose and mouth. The other discriminators are the global and local discriminators that look ensure consistency on the entire image and local masked region. The authors used facial landmarks to locate the eyes, nose and mouth. These locations are cropped using a mask with fixed size corresponding to the landmarks. This model uses the $\boldsymbol{\ell}_1$ and perceptual loss [94] to optimise the generator combined with ongoing adversarial loss based on the PatchGAN discriminator [85]. Segments of the face (eye, nose, mouth) are cropped and each passed through a discriminator to authenticate its generated features. Qualitative and Quantitative evaluations compared the effectiveness of the model with the state-of-the-art [13, 226, 239, 228, 227]. Quantitatively, the $\boldsymbol{\ell}_1$, PSNR, SSIM and Learned Perceptual Image Patch Similarity (LPIPS) [233] were used and shown in Table 2.4. The advantage of this network is that it uses ground-truth as a direct supervision to obtain high fidelity features for the masked regions on the input masked-image. The limitation of this model is that if learned attention is insufficient or not accurate, poor quality filling will result in the generated image due to unsuitable features filling in the missing regions.

Feature Propagation

Liu et al.[124] used the U-Net [163, 85] and replaced convolutions with partial convolutional in inpainting task. The partial convolutional operation has an automatic mask update step. The re-normalized masked-convolutions operations focus on valid pixels, followed by an automatic mask generation to the next layer as a forward pass. The loss functions used to handle pixel-reconstruction accuracy of the hole region are, $\boldsymbol{\ell}_1$ loss, perceptual loss [55] and style-loss. The mask updating step is non-learnable and with a fixed convolutional layer with a kernel size that matches that of the partial convolutional operation with weight initialised to 1 and no bias layer. The partial convolution layer, with automatic mask-update mechanism, undergoes a sufficient number of continuous updates to remove any masking on the unmask value in return for accurate feature maps. The comparative evaluation against the state-of-the-art

Barnes et al. [13], Iizuka et al. [82], Yu et al. [226], using a binary mask of hole-to-image area ratio of [0.5,0.6]. The performance evaluation of this method [133] by the authors compared to the state-of-the-art showed high PSNR and SSIM values for all mask sizes. This method suffers from the reliance on initial hole values, that causes the algorithm to produce images that lack plausible output texture. Also, it struggles with sparsely structured images and binary masks with larger hole-to-image-ratio. This is because neurons with receptive fields cover valid or invalid pixels at different spatial locations. The invalid pixels disappear following the rule-based mask layer by layer leading to some missing information in deeper layers that may be needed to synthesis pixels in mask regions.

Wang et al. [202] introduced a Laplacian-pyramid based GAN to inpainting. Using a modified ResNet block by He et al. [73], the aim is to propagate high-frequency details from the surrounding to predict precise missing information while eliminating colour discrepancy. The modified ResNet block, implemented with dilated convolutions, implies a larger receptive field with batch normalisation layers and rectified linear unit for speed convergence. The Wang et al. [202] introduced a combined representation learning with reconstruction and residual learning in the generator network to extract predicted missing regions and therefore combine features of low middle level of fine layers. The generator model captures the image content and compresses it to a latent representation. These are feature extraction progressively predict missing regions while the residual learning phase learns the difference between visible colour similarities of predicted pixels and surrounding pixels. The loss function uses trained-VGG model to extract features and uses feature space, combined with pixel-wise and adversarial loss to learn photo-realistic images, therefore maintaining the natural artefacts of the original image while completing the missing region. The masks used are rectangular and of size 128 × 128, and are randomly positioned on the input image. The Wang et al. [202] method achieved the best performance scores compared to [152, 82, 122, 226, 131, 223]. This model performed particularly well with regularly shaped binary mask, but was not experimented with irregularly shaped mask. Also, the model output had colour discrepancies, hence the model cannot be generalised to natural scene inpainting.

Li et al. [120], proposed the use of visual features and structures to restore or inpaint missing parts of an image. This model introduces a visual reconstruction layer to the U-Net [163] combined with partial convolutions [124], and a bottleneck residual block. The encoder uses upper bound additional visual reconstruction layers to estimate the edges of the missing structure before passing these to partial convolution layers. Within the decoder are lower-bound additional visual reconstruction and convolution feature reconstruction layers. The visual reconstruction layers entangle the reconstruction of visual structures and features of an image . The masks regions are progressively filled-in with meaningful content based on the reconstructed edges and the input image. The use of Patch-GAN discriminator [85] with slight adjustments to include spectral normalisation controls the generalisation error. The network is end-to-end with detailed generation of restored missing structure assisted by adversarial loss combined with loss functions from [124]. It should be noted that parameter fine-tuning is required before training the network. Across all hole-to-image ratios used during the evaluation, the network had a slight edge in performance compared with the state-of-the-art [145, 124]. The results for 10%-30% hole-to-image ratio are shown on Table 2.4. However, it is time consuming for the visual reconstruction layers to learn structural parts, thus increasing the time to filter out unwanted structures not needed for image reconstruction.

Yu et al.[227] used gated convolutions combined with contextual attention layer and Spectral-normalized Markovian Discriminator (SN-PatchGAN) for inpainting task. The backbone of the network is an encoder-decoder stacked with gated convolutions, contextual attention layers and a refinement unit of dilated convolutions. Gated convolutions allows the network to learn soft mask from input data. Within these convolutions are processes that learn features from input data progressively for each channel. At each spatial location the prediction of missing pixels are conditioned on the valid pixels in the input image. During the process, each gating block produces two outputs that go through different activation mechanisms producing gating values and learned features. The contextual attention layer enables the network to capture long-ranged features from distant spatial locations. This is assisted by dilated convolution blocks used as refining mechanisms within the network. The discriminator as part of the combined network, outputs a 3D-shape feature based on three inputs (image, mask and guidance channel). Based on the report by Yu et al [227], the performance of this algorithm as shown in Table 2.4 is better with free-form mask than rectangular mask, though the sizes of the masks are not detailed in their report. However, gated convolutions will perform better with free-form than rectangular masks, which limits the algorithms performance to generalize larger masks or large images with hole-to-image ratio. Additionally, with gated convolutions, correlation between valid features is not guaranteed and may lead to colour discrepancies on the completed image. Furthermore, this network is computationally expensive due to the gating within convolutions and the three stage network which has to be trained end-to-end.

Liu et al. [126] proposed the use of semantic relevance between hole and non-hole regions for effective prediction of the hole, aided by a contextual structure preserving mechanism known as Coherent Semantic Attention Layer (CSAL). The design is a two stage network based on the U-Net [163], with ongoing adversarial training. The CSAL is an embedding within the refinement block of the two-step U-Net architecture, implemented to elevate the quality of reconstructed images. A proposed consistency loss is used, combined with feature patch and patch discriminator [85] to improve stability during training and maintain the natural statistics of the image details. The consistency loss computes the error between corresponding CSAL layers of the encoder-decoder block and VGG features. The feature and patch discriminator combined introduces ongoing adversarial training, based on the relativistic average adversarial loss [95]. The patch discriminator evaluates the pixels values on the final output compared to the input image. Introduced within the consistency loss is the reconstruction loss ($\boldsymbol{\ell}_1$), as a constraint to assist the model in learning meaningful parameters that can approximate the ground-truth image. Note, these results shown in Table 2.4 are based on a hole-to-mask ratio of 10%-20%. However, despite the high performance of this algorithm, the CSAL may fail due to the nature of the network. That is, if the network is too deep or too shallow, loss of information may occur leading to increased time overhead.

Yi et al. [224] modified gated convolutions [227] into light-weight model and introduced high-frequency residuals to generate rich and detailed textures for high resolution images. Motivated by the size of images captured by mobile phone cameras, the authors proposed a contextual residual aggregation mechanism that borrows contexts, features and residuals. The scores computed are between patches inside/outside the missing region within a specific region that has high affinity of similar patches. Gated convolutions are modified into depth-separable, pixel-wise and single-channel variants. Depth-separable uses depth-wise convolutions followed by 1×1 convolutions as a gating mechanism. Pixel-wise uses 1×1 convolutions as gating mechanism. Single-channel broadcast a single mask to all channels as a hard mask, similar to partial convolutions [124]. The network is a two-stage network that uses single-channel for all layers in a coarse-network and depth-separable or pixel-wise in the refinement network. The loss function is a reconstruction loss based on $\boldsymbol{\ell}_1$ in the generator and adversarial loss. The qualitative results gives a high-visual quality of the images compared to the state-of-the-art [82, 226, 227, 228, 124]. In quantitative analysis, there is not a significant effect in the measurement compared to the state-of-the-art. From the table of result presented by the authors, it is observed that four performance evaluation metrics are used in addition with a time factor as a measure to rate the algorithms performance. Different resolutions on the places2 [240] dataset were compared against the state-of-the-art, with higher resolution images of size 1024 × 1024 the algorithm performed overall best with a time difference of -696ms. The results on Table 2.4 shows the results for 1024 × 1024 image sizes. The algorithm induced a large effect on high resolution images, proving its ability to generate high-quality contents for missing regions such images. However, with the poor performance in low resolution images, this area of studies still remain a challenge.

Li et al. [121] proposed a recurrent learning approach, where feature maps are inferred in shared recurrent units. The approach uses partial convolutions [124] to identify target regions and use the output as input to an encoder-decoder generator with skip connections. The mask updating mechanism within partial convolutions is exploited during each recurrence as a prerequisite to identify the target regions for subsequent recurrences. Within this network, encoded features undergo a series of recurrence to maximise inference capability to obtain high-quality features during an inpainting task. This means that the hole regions shrink with each recurrence until a high-quality feature is achieved. The mean of the various feature outputs of the network is the final output the decoder. The authors also proposed an attention layer that uses prior knowledge of background pixels to assist the model to obtain best patches at different occurrences that are consistent with the predicted regions and the image. The authors used perceptual and style loss [94] formulated using feature maps from i^{ith} pooling layer extracted from VGG-16 network. Other loss functions used calculate the $\boldsymbol{\ell}_1$ of unmasked and masked regions respectively as valid and hole loss. Quantitative evaluations were conducted compared to the state-of-the-art approaches [239, 124, 227, 145, 120] and are shown in Table 2.4. The limitation of this method is that some boundary artefacts may occur due to inconsistencies with feature maps posing as shadow-like regions during feature merging process.

In summary, the use of deep learning methods in inpainting produces plausible results when compared to the original image. However, the limitation thus far by the state-of-the-art is the failure in reporting appropriate analysis of results. So far, the best evaluation and analysis of results is by Liu et al. [124], Nazeri et al. [145], Xie et al. [211] and Li et al. [120]. These authors provide details of the various hole-to-image ratios used for qualitative and quantitative evaluation. For example, Liu et al. [124] report details of non-border mask and mask at border regions across evaluation

Table 2.2: Summary of reviewed literature on deep learning methods in image inpainting.

Method	Model	Description	Loss function	Dataset
[86]	CNN	Auto encoder. Formulated for image denoising and extended to image inpainting.	Reconstruction loss.	In-house 100 greyscale images [86]
[212]	CNN	Sparse coding and deep neural networks as a denoising auto-encoder and extended to image inpainting.	Reconstruction loss.	In-house images [212]
[152]	GAN	Encoder-decoder architecture as the generator and Discriminator network. For example Figure 2.8	ℓ_2 reconstruction, Adversarial loss.	Paris Street View [152, 45], ImageNet [168, 167], PASCAL VOC2007 [53]
[82]	GAN	Encoder-decoder generator with a refinement network based on dilated convolutions combined with Global and local discriminators.	weighted ℓ_2 Adversarial loss	Places2 [241], ImageNet [168], CMP Facade [195].
[217]	GAN	Uses style transfer network and context-encoder to ensure Multi-scale neural patch synthesis to preserve contextual structure with local patch similarity on images with high-frequency details.	Adversarial loss, ℓ_2-based texture loss computed with features from VGG19.	Paris Street View [152, 45] and ImageNet [168].
[223]	GAN	Searches latent space encodings assisted by spatial attention mechanism to reconstruct the original image.	Weighted context ℓ_1 based loss and Adversarial loss.	CelebA [133, 132], SVHN [223], Standford Cars [223].
[226]	GAN	A two stage model network (encoder-decoder) that uses cosine similarity assisted by contextual attention layers, redesigned to use dilated convolutions	Reconstruction loss and two Wasserstein GAN losses.	CelebA [133, 132], CelebA-HQ [98], DTD [226], ImageNet [168] and Places2 [241].
[124]	GAN	A U-NET architecture, that uses partial convolutions instead of normal convolutions.	Perceptual loss, style loss, adversarial loss.	CelebA [133, 132], CelebA-HQ [98], Places2 [241] and ImageNet [168].
[216]	GAN	Uses a U-NET architecture to introduction Shift connection layer to transfer fine texture details. For example Figure 2.9	Guidance loss, ℓ_1 and adversarial loss.	Paris Street View [152, 45] and Places2 [241].
[203]	GAN	A Laplacian pyramid GAN, reconstruction and residual learning in generator.	VGG-Feature loss, Adversarial loss.	CelebA [133, 132] and Paris Street View [152, 45].
[79]	GAN	Introduced padding and pooling operations in an Encoder-decoder, to avoid edge disappearance. The model also uses a mini-batch discriminator for realistic photo completion.	ℓ_2-based SSIM loss and Adversarial loss.	In-house dataset containing 2015 images [79].
[228]	GAN	A U-NET architecture that uses a cross-layer attention and pyramid filling mechanism.	ℓ_2 and Adversarial loss.	Facade, DTD [239], CelebA-HQ [98] and Places2 [241].
[120]	GAN	U-NET, visual reconstruction layers, residual block, partial convolutions , patch discriminator.	Perceptual loss, style loss, adversarial loss.	Places2, Paris Street View [152, 45] and CelebA [133, 132].
[227]	GAN	Gated convolutions, contextual attention layer and SN-PatchGAN	Pixel-wise reconstruction loss (ℓ_1) and adversarial loss.	Places2 [241].
[126]	GAN	A two-stage network that uses Coherent semantic attention layer that preserves the spatial structure of the image within the refinement network.	Consistency loss (ℓ_2), feature loss (VGG) and adversarial loss.	CelebA [133, 132], Places2 [241] and Paris Street View [152, 45].
[87]	GAN	An encoder-decoder network that uses a reverse masking mechanism to enforce prediction only on missing pixel regions with contextualised features on the unmasked regions, to improve the quality of the inpainted image.	Reversed-mask loss (ℓ_2-base), feature loss (VGG) and adversarial loss (WGAN).	CelebA-HQ, Places2 [241] and Paris Street View [152, 45].
[238]	GAN	A trio-network network that uses joint probability distribution combined with a cross-semantic attention layer	conditional constraint loss (ℓ_2-base), feature loss (VGG) using ℓ_1-base and adversarial loss (WGAN).	CelebA-HQ [98], Places2 [241] and Paris Street View [152, 45].
[243]	GAN	A U-Net architecture that uses a dual spatial attention layer	ℓ_1, feature loss (VGG) and adversarial loss (PatchGAN).	Flickr-Faces [102].

metrics compared with the state-of-the-art. This provides the reader with a true picture of the performance of the algorithm. However, with the details of various mask sizes and mask regions provided by this author [124], the training dataset is not publicly available. Also, on a subjective evaluation, the authors used a wider audience compared to Li et al. [120], and Yu et al. [227] to judge the quality of the images. Additionally, randomised missing data (random mask) is more difficult to learn compared to missing data in a central region of an image. The difficulty for an algorithm to capture semantic information for images with masks at border regions and preserve edges still remain challenging. For example, Liu et al. [124] and Yu et al. [227] introduced algorithms that uses masks to infer missing pixels compared to all other methods. In terms of quantitative evaluations, report by [223] point out that quantitative results do not have a true representation for different methods.

2.4 Datasets

With the wider use of deep learning in present inpainting research, the data and the masks are two essential components to train and evaluate the performance of the methods. The following discuss some popular mask datasets and image datasets in image inpainting.

2.4.1 Nvidia Mask Dataset

The Nvidia Mask dataset proposed by Liu et at. [124], Figure 2.10 has six categories of masks of different hole-to-image ratios. This dataset contains 55,116 training masks and 24,866 testing mask, and of 512 × 512 resolution.

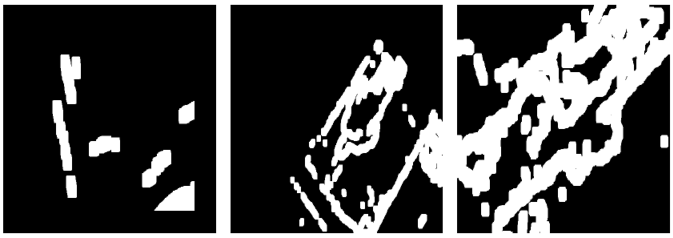


Figure 2.10: Examples of binary masks from Nvidia Mask Dataset [7].

2.4.2 Quick Draw Irregular Mask Dataset

The quick draw mask dataset proposed by Iskakov et al. [84], Figure 2.11 contains 50,000 train and 10,000 test masks. The samples are of size 512 × 512 resolution and used for image inpainting task [84, 88].

2.4.3 Caltech Faces

Caltech Faces [7], Figure 2.12 is a sample from the Caltech dataset, containing 450 face images, from 27 different people, at 896 × 592 resolution in JPEG format under different lighting conditions, expressions and background.

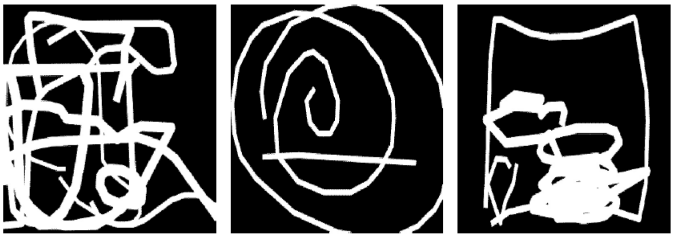


Figure 2.11: Mask Dataset [84].

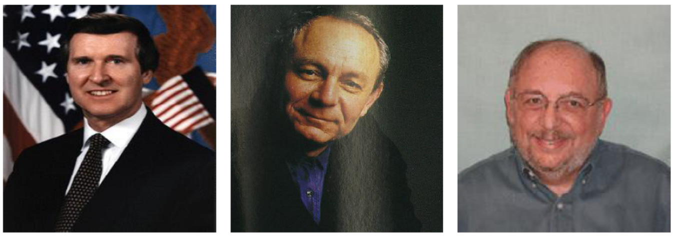


Figure 2.12: Caltech Dataset [7].

2.4.4 Places2

Places2 is designed following the principles of HVS [240, 241] containing images of diverse scenery used for high-level visual understanding task. Consisting of more than 10 million images and containing more than 400 unique scene categories, it has 5,000 to 30,000 training images consistent with real-world occurrences. It is used for learning in-depth scene features using CNN for various scene recognition task, e.g. Figure 2.13.

Figure 2.13: Places2 Dataset [241].

2.4.5 Paris Street View

Doersch et al. [45] developed the Paris Street View dataset from Google Street View [65, 8] to examine which specific algorithms would work on a computational geographic task, and therefore enable automatic location of geoinformation features for a particular place or city. The images are distinctive and geographically informative, being based on a variety of architectural correspondences and geospatial scales (summarised appearance on one specific scale) of different cities from around the world. Two perspectives of images of 936 × 537 resolution are scraped automatically from a dense sampling of panoramas [65]. Approximately 10,000 images per city were downloaded from 12 cities across the world, with a focus on Paris and suburban areas. A sample of the dataset is shown in Figure 2.14.

Figure 2.14: Paris street view Dataset [45].

2.4.6 CelebA

CelebFaces Attributes Dataset (CelebA) is a collection of 202,599 facial images of celebrities [133, 132] containing 10,177 identities, five landmark locations and each with 40 binary attribute annotations cropped to size 178 × 218 resolution as of 2015. This dataset makes it an appropriate test set for facial image synthesis [178] since there are considerable pose variations and background clutter associated with the database alongside a broad diversity and rich annotations.

2.4.7 CelebA-HQ

The CelebA-HQ dataset, developed by Karras et al. [98] is developed from the from the CelebA dataset consisting of 30,000 high-quality images images of 1024 × 1024, 512 × 512 and 128 × 128 resolution. The original image resolution in the CelebA dataset varied from 43 × 55 to 6732 × 8984 with various backgrounds and processed by different image quality measures to ensure the image is on the central region [98]. To obtain the high-quality images for the CelebA-HQ, each JPEG image was processed using two pre-trained neural networks. The authors then used the model proposed by Mao et al. [142] to remove JPEG image artefacts which was combined with an adversarially-trained 4x super-resolution network for high-resolution images similar to that in [114]. Padding and filtering were applied to extend the dimension of the images. The authors then used facial landmark annotations included in the original CelebA dataset to orientate and crop the images. The 202,599 subjects in the dataset were processed and analysed, resulting in the best 1024 × 1024 resolution image, and sorted to estimate the best quality images to select 30,000 images. To obtain the rest of the image sizes, the GitHub repository resizing tool by [98] is implemented. For example, Figure 2.15 shows a sample from the dataset.

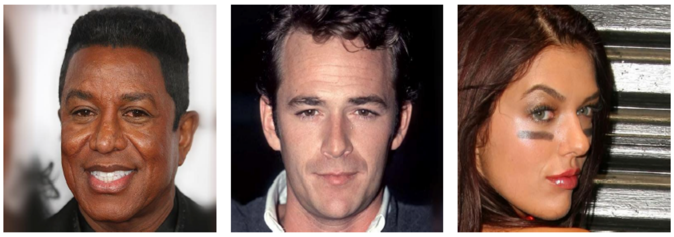


Figure 2.15: CelebA-HQ Dataset [98].

2.4.8 ImageNet

The ImageNet Large scale visual recognition challenge (ILSVRC) has collated millions of images classifying hundreds of different object categories [168, 167]. It is large ground-truth annotated dataset of images put together for object recognition, detection and classification for the comparison of state-of-the-art algorithms for computer vision accuracy with human accuracy. It contains over 10,000 categories with more than 8 million images of variable resolution [106], e.g. Figure 2.16 show samples from three classes.

Figure 2.16: ImageNet Dataset [106].

2.4.9 PASCAL Visual Object Classification (PASCAL VOC)

The PASCAL VOC visual object classes consists of two components: a publicly available and an annual competition datasets (PASCAL VOC2005, PASCAL VOC2007, PASCAL VOC20012). Established in 2005, it provides a standardised dataset closest to ILSVRC for object detection, image classification, object segmentation, person layout and action classification [53] for the annual competition. As of 2010, the PASCAL

VOC dataset has a total of 19,737 for 20 object categories organised into train, validation and test sets; Figure 2.17 shows a sample taken from three different categories of this dataset.

Figure 2.17: PASCAL VOC Dataset [53].

Table 2.3: Datasets used in Image Inpainting.

Datasets	Total Images	Purpose	Resolution
Nvidia Mask [124]	79,982	masks	512 × 512
Quick Draw Mask [84]	60,000	masks	512 × 512
Caltech Faces[7]	450	Various	896 × 592
Places2 [241]	30,000	Urban	Variable
Paris street view[45]	14,900	Urban	936 × 537
CelebA[131, 133]	202,599	Various	178 × 218
CelebA-HQ [98]	30,000	Inpainting	1024 × 1024
ImageNet [168]	>8 Million	Classification	Variable
PASCAL VOC[53]	14,974	Classification	Variable

In summary, the challenges to developing applications rely on the mathematical equations, optimisation parameters and dataset used to test the robustness. Table 2.3 shows a summary of the popular datasets used by researchers for the evaluation of inpainting algorithms.

2.5 Performance metrics for image inpainting algorithms

To ease readability, it is good to first highlight the performance metrics to quantitatively evaluate the performance of the state-of-the-art methods. This is because inpainting algorithms generate images which are distorted or show changes in appearance. To evaluate the performance of these algorithms, different performance metrics are used to quantify the generated images. Methods, based on the highly developed Human visual system (HVS), have mostly used qualitative questionnaire evaluation to extract structural context without need for a large dataset, making this both time consuming and costly. However, some authors use both qualitative and quantitative performance metrics with most commonly used being $\boldsymbol{\ell}_1$ (Mean Absolute Error), $\boldsymbol{\ell}_2$ (Mean Square Error), Peak signal to Noise Ratio (PSNR) and Structure Similarity Index Measure (SSIM). These tools measure the perception of an image to quantify the quality of the error between the distorted pixels of the reconstructed image and the (original) reference image. Other evaluation metrics (Visual information fidelity [174], universal quality index [206], inception score [171], Multi-scale SSIM [208],Frechet inception distance [75] and LPIPS [233]) have been reported in literature, however, the focus is mainly on the most used ones in this review. The quantitative measure, or score, of the generated image need change only by a few pixels to validate the effectiveness of an algorithm. Given the ground-truth image and inpainted image, $\boldsymbol{\ell}_1$ is the total value of the absolute difference between the pixel values of the predicted image and the actual pixel values of the ground-truth image.

$$\boldsymbol{\ell}_1(\mathbf{x}, \mathbf{y}) = \frac{1}{N} \sum_{i=1}^{N} |x_i - y_i| \tag{2.2}$$

In Equation 2.2, $\boldsymbol{\ell}_1$ gives an overview of the average error for a predicted image. A low computed $\boldsymbol{\ell}_1$ indicates that the quality of the image is good [134]. In Equation 2.3, $\boldsymbol{\ell}_2$ averages the squared intensity difference between the reference image and the reconstructed image [69].

$$\boldsymbol{\ell}_2(\mathbf{x}, \mathbf{y}) = \frac{1}{N} \sum_{i=1}^{N} (x_i - y_i)^2 \tag{2.3}$$

However, the error in Equation 2.3 may not match the perceived visual quality of the image.

PSNR is the ratio of the maximum possible value (power signal) to the power of distorting noise that affects the representation of quality based on two images (reconstructed/original) of the same kind.

$$PSNR = 20log_{10} \frac{(MAX_I)^2}{\sqrt{MSE}} \tag{2.4}$$

Equation 2.4 computes PSNR (dB), well known for assessing the quality of noisy images [77], and an approximate value of 48dB is considered good [12]. The higher the PSNR value, the better the quality of the reconstructed/generated image.

The SSIM [207] has become a good correlator for quality perception that discounts aspects of an image not important to the HVS. The SSIM models three factors (loss of correlation, luminance distortion and contrast distortion) of two images based on neighbouring and corresponding pixels. Given the input signals (x,y), SSIM computes the combination of luminance, contrast and structure to output a similarity measure expressed in Equation 2.5;

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{({\mu_x}^2 + {\mu_y}^2 + C_1)({\sigma_x}^2 + {\sigma_y}^2 + C_2)} \tag{2.5}$$

where C_1 and C_2 are constants. This method assesses the quality of the image based on the degradation of structural information of the reconstructed image. However, the above quality measures quantify the image whereas subjective assessments depends on the HVS to extract the structural information of the image. This method is, again, both time-consuming and costly.

Fréchet Inception Distance (FID) [75] measures the quality of reconstructed images by computing the distance between two distributions (ground-truth and generated image). The metric quantifies the quality of generated samples by embedding features to a specific layer of the Inception Net [75, 171] feature space. Based on the assumption that the embedding layer is a continuous multivariate Gaussian, the mean and the covariance are measured for both the ground-truth and generated image. The distance between these two Gaussians is expressed in Equation 2.6;

$$FID(x,y) = ||\mu_x - \mu_y||^2 + Tr(\Sigma_x + \Sigma_y - 2(\Sigma_x\Sigma_y)^{1/2}) \tag{2.6}$$

where the means and covariance matrices of the ground-truth and generated image distributions are given by μ_x, Σ_x and μ_y, Σ_y. A lower FID score means that the quality of the reconstructed image is close to the ground-truth indicating better. Also, the FID score is consistent with human judgement and robust to noise [75, 138], which makes it a good metric for images generated by inpainting algorithms. The limitation with this method is that it lacks the capability to detect overfitting.

2.6 Discussion

In this review, it is observed that inpainting remains an important, yet challenging, research area in computer vision. Another observation is that traditional approaches [18, 5, 231, 74, 58] can handle textural and structural target regions, and are suitable for disocclusion or object removal. A further observation and point to note is that inpainting methods in this category (traditional inpainting methods) use various techniques e.g. [17, 13, 119, 128] and have shown exceptional performance in linear structures using diffusion. However, some methods e.g. [49, 182] in this category

Table 2.4: Summary of quantitative results of some deep learning methods for image inpainting on Places2 [240, 241], CelebA-HQ [98] and Paris Street View [45] datasets. The performance evaluation vary from method to method and are approximated to 2 decimal places. The results included are for distortions (image-to-mask ratio) between 10%-20% on image sizes 256 × 256. † Lower is better. ⊎ Higher is better.

Method	Mask Type /	Image	**Places2**						**CelebA-HQ**						**Paris Street View**					
	Image-to-mask Ratio	Size	MAE†	MSE†	FID†	PSNR⊎	SSIM⊎	IS ⊎	MAE†	MSE†	FID†	PSNR⊎	SSIM⊎	IS ⊎	MAE†	MSE†	FID†	PSNR⊎	SSIM⊎	IS ⊎
[152]	Square		—	—	—	—	—	—	—	—	—	—	—	—	0.10	0.23	—	17.59	—	—
[217]	Square (64 × 64)	128 × 128	—	—	—	—	—	—	—	—	—	—	—	—	10.01	2.21	—	18.00	—	—
[226]	Irregular (10%-20%)	256 × 256	8.6	2.1	—	18.91	—	—	—	—	—	—	—	—						
[124]	Irregular (10%-20%)	256 × 256	0.49	—	—	33.75	0.94	0.05	—	—	—	—	—	—	—	—	—	—	—	—
	Irregular (30%-40%)	256 × 256	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	Irregular (40%-50%)	256 × 256	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	Irregular (50%-60%)	256 × 256	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
[216]	Irregular (10%-20%)	256 × 256	—	—	—	—	—	—	—	—	—	—	—	—	—	0.02	—	26.51	0.90	—
[202]	Square (128 × 128)	256 × 256	—	—	—	—	—	—	—	—	—	23.45	0.86	—	—	—	—	—	—	—
[228]	Square (128 × 128)	256 × 256	9.94	—	15.19	—	—	50.51		—	—	—	—	—						
[120]	Irregular (10%-20%)	256 × 256	0.012	—	—	28.87	0.95	—	—	—	—	—	—	—	—	—	—	—	—	—
	Irregular (20%-30%)	256 × 256	0.022	—	—	25.66	0.91	—	—	—	—	—	—	—	—	—	—	—	—	—
	Irregular (30%-40%)	256 × 256	0.033	—	—	23.46	0.86	—	—	—	—	—	—	—	—	—	—	—	—	—
	Irregular (40%-50%)	256 × 256	0.046	—	—	21.74	0.79	—	—	—	—	—	—	—	—	—	—	—	—	—
	Irregular (50%-60%)	256 × 256	0.068	—	—	19.51	0.67	—	—	—	—	—	—	—	—	—	—	—	—	—
[227]	Irregular (10%-20%)	512 × 512	9.1	1.6	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	Square	512 × 512	8.6	2.0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
[126]	Irregular (10%-20%)	256 × 256	—	—	—	—	—	—	0.72	0.04	—	34.69	0.98	—	—	—	—	—	—	—
	Irregular (20%-30%)	256 × 256	—	—	—	—	—	—	0.94	0.07	—	32.58	0.98	—	—	—	—	—	—	—
	Irregular (30%-40%)	256 × 256	—	—	—	—	—	—	2.18	0.37	—	25.32	0.92	—	—	—	—	—	—	—
	Irregular (40%-50%)	256 × 256	—	—	—	—	—	—	2.85	0.44	—	24.14	0.88	—	—	—	—	—	—	—
	Square (32 × 32)	256 × 256	0.01	—	—	27.75	0.93		1.83	0.27	—	26.54	0.93	—	—	—	—	—	—	—
[121]	Irregular (10%-20%)	256 × 256	0.014	—	—	27.75	0.93	—	0.007	—	—	33.56	0.98	—	0.011	—	—	31.71	0.95	—
	Irregular (30%-40%)	256 × 256	0.038	—	—	22.63	0.81	—	0.02	—	—	27.76	0.93	—	0.027	—	—	26.44	0.86	—
	Irregular (50%-60%)	256 × 256	0.076	—	—	18.92	0.59	—	0.047	—	—	22.88	0.81	—	0.054	—	—	22.40	0.68	—
[243]	Square (128 × 128)	256 × 256	—	—	—	—	—	—	1.46	—	—	26.36	0.91	—	—	—	—	—	—	—
[224]	Square (128 × 128)	512 × 512	5.43	—	4.89	—	—	17.72	—	—	—	—	—	—	—	—	—	—	—	—
	Square (128 × 128)	1024 × 1024	5.43	—	4.89	—	—	17.72	—	—	—	—	—	—	—	—	—	—	—	—
	Square (128 × 128)	2048 × 2048	5.49	—	4.89	—	—	17.85	—	—	—	—	—	—	—	—	—	—	—	—
	Square (128 × 128)	4096 × 4096	5.50	—	4.890	—	—	17.81	—	—	—	—	—	—	—	—	—	—	—	—
[87]	Irregular (10%-60%)	256 × 256	0.27	—	4.47	39.66	0.93	—	0.31	—	3.09	40.40	0.94	—	0.33	—	17.64	39.55	0.91	—
[238]	Irregular (10%-20%)	256 × 256	—	—	—	—	—	—	1.51	—	—	26.38	0.88	3.01	—	—	—	—	—	—

perform poorly on mask regions along curves and edges. In contrast, some methods e.g.[17, 13] handled such regions with plausible outcomes, but are slower and can only perform inpainting on single images. A number of these [17, 160, 194] methods produce good results in propagating linear structure using diffusion, but also introduced blur into the target region, making them unsuitable for highly textured images with large missing regions. These methods whilst showing excellent performance have a shortcoming of preserving image realism.

Datasets, e.g. [45, 131, 98], which often contain thousands or millions of images, are crucial to deep learning methods and enable an algorithm to learn features and complete target regions to produce semantically plausible outcomes. The wide diversity in the complexity of structural and textural information of images has a significant impact on the results. To test the robustness of an algorithm, a complex dataset is a requirement since it provides a model with multitude of patterns (sophisticated features) to learn and output plausible satisfactory results. Furthermore, deep learning methods e.g [152, 82, 124, 226, 227, 79] have produced plausible outcomes in inpainting when compared to the original image, due to their ability to extract features in an end-to-end fashion. However, the weakness encountered is the difficulty in reproducibility across papers. Generally, most authors do not appear to share their code and therefore meaningful comparison is both made difficult and is not progressive. A fair comparison between the deep learning techniques is challenging due to the lack of a dedicated and standardised benchmarking system. Such as system would allow newer models to be compared against an accepted set of models acting as the baseline. A further observation is that not all proposed algorithms using similar parameters use the same dataset for training and testing. Also, most algorithms do not disclose parameter fine-tuning, data-preprocessing steps and complexity of the model, and for codes made publicly available, not all are complete. This definitely leads to poor evaluation of different codes if all the information is not available, which may or may not be robust, making it harder to obtain a progressive trajectory in research direction. Additionally, hardware is a hindering/limiting factor in the progress of newly proposed algorithms in this field. However, a good inpainting algorithm should have qualities which, in addition to robustness, also embody high performance with low computational cost.

Computational analysis is a key factor that needs addressing in detail to support the quality of an inpainting algorithm. This will aid the reader to understand the quality of different methods based on some factors such as; running time, training speed, inference time and training time as mention, also highlighted by [52]. To this effect, it will be advantageous for future works to consider the type of mask used (difficulty in terms of image-to-hole ratio), the dataset, the image sizes and the type of GPU machine used as reported by Liu et al. [126] to perform computational analysis. Table 2.5 shows a summary of methods that have some form of record

Table 2.5: Summary of the type of GPU and record on experimental details for evaluation computational time by some deep learning image inpainting methods. Note that the record on this table has been extracted from the proposed method and are not based on our evaluation.

Method	Type of GPU & computational analysis	Image resolution	Batch size	Dataset
Pathak et al. [152]	The training time took 100,000 iterations is 14 hours.	256 × 256	—-	Paris Street View [152]
Iizuka et al. [82]	The training took 500,000 iterations on a single machine with four K80 GPU took two months. Also,further evaluations using GeForce TITAN X GPU reports a drop in computational time to 0.141s per 512 × 512 image. 0.141s per image	512 × 512	96	ImageNet [168]
Yu et al. [226]	Initial reported time was 11,520 GPU hrs on K80. Improved training time to 120GPU hrs using GTX 1080TI. The full model runs at 0.2s per frame	512 × 512	—-	Places2 [241]
Yang et al. [217]	The time taken to fill in 256 × 256 hole on an image size of 512 × 512 using a TITAN X GPU which is slower compared to Pathak et al. [152]. It takes 1 minute for a single image as reported by the authors.	512 × 512	—-	CelebA [133, 132]
Liu et al. [124]	The network inference time is 0.029s using NVIDIA V100 GPU (16GB), regardless on the mask-to-image ratio on the image.	512 × 512	6	CelebA-HQ [98]
Yan et al. [216]	IT takes one day on a TITAN X Pascal i.e. 24 hours for 30 epochs.	256 × 256	1	Paris Street View [152]
Huang et al. [79]	GeForce GTX 1070Ti	256 × 256	4	In-house road images. [79]
Zeng et al. [228]	The model runs at 0.19s per frame on a TITAN V GPU.	256 × 256	—-	CelebA-HQ [98]
Li et al. [120]	The training time is 3 days using RTX 2080TI 11G GPU for CelebA and two weeks for Places2 [241].	256 × 256	5	CelebA [133, 132] Places2 [241]
Yu et al. [227]	During testing, it takes 0.21 secs per image using NVIDIA(R) Tesla(R) V100 GPU.	512 × 512	—	CelebA-HQ [98]
Liu et al. [126]	It takes 9, 5 and 2 days for Places2 [241], CelebA, and Paris Street View using NVIDIA 1080TI GPU(11GB). Overall inference time is 0.82s per image.	256 × 256	1	Places2 [241] CelebA [133, 132] Paris Street View [152].
Yi et al. [224]	Trained using two NVIDIA GTX 1080TI GPU.	512 × 512	8	CelebA-HQ [98]. Places2 [241].
Zheng et al. [239]	—-	256 × 256	1	CelebA [133, 132]
Zhao et al. [238]	It takes 500,000 iterations to train the model.	256 × 256	8	CelebA-HQ [98]
Zhou et al. [243]	It takes 4 days to train using NVIDIA TITAN Xp (12GB).	256 × 256	16	CelebA-HQ [98]

with regards to computational analysis. This is not in full detail because it does not have all the relevant information that can determine the best algorithm. However, it will be important to provide the full specifications of the machine, measure in terms of the number of epochs, batch size, training data, and provide information on dataset preprocessing. Other methods [152, 216, 124] have reported some training time. Another analysis measure to consider is the time taken to compute the metric score based on a single image resolution. Despite the great success of deep learning methods, there are still drawbacks in terms of computational complexity and failures in preserving image realism.

2.7 Research Direction

Image inpainting, from traditional to deep learning methods, has achieved immense, and continued, success. A comprehensive review for a range of methods from the perspective of the algorithm (it's development and how it is used) for inpainting tasks, datasets, performance evaluation and limitations of the methods are presented in this chapter. It is observed that the poor(er) performance of traditional methods on images with more extensive binary mask and facial images due to complexity in features on the image. Remedying this limitation, deep learning methods have developed to become state-of-the-art, showing great success on images containing intricate patterns, but also with shortcomings. This thesis will focus on the following gaps identified in this literature:

- The inpainting process in GANs is challenging. This could be due to the network design being used as part of the generator or discriminator. Pathak et al.[152] pioneered the use of GAN in image inpainting to generate realistic images. The channel-wise fully connected layer, which they proposed, is intended to propagate information to the decoding layers. However, flaws such as blurry artefacts and failures with high-resolution images were observed. This has been addressed in research by introducing skip connections [216] to fuse features extracted from encoder layers with those of decoding layers. This method, however, necessitates careful consideration of computational resources, and there is still room for improvement in this area.

- Another factor to consider is the loss. Different losses compute gradients in different ways, which affects backpropagation performance in various ways. It makes no difference how good a loss is; what matters is the combination and how it is used during training. Feature losses have already been used to solve blurry artefacts and overly smooth edges on generated images. However, a more effective technique for using features as a loss model has been introduced in the contribution chapters of this thesis, demonstrating that there is always room for improvement.

- The techniques and applications of the inpainting algorithms proposed in the literature vary; while some [226] have introduced layers to address the inpainting problem, there are still limitations. To generate high-quality images, the inpainting algorithm must be capable of extracting subtle and high-level features while also transferring these features to the decoding layers. An attempt to address this gap in Chapter 7 by introducing a module to highlight high-level features with good transition to the decoding layers.

Research on image inpainting using deep learning has witnessed good progress in recent years. Many datasets and the masks region for inpainting were generated but lack of standardisation. Some potential works to advance the field of inpainted are outlined below:

- **Datasets**. To track and determine which model is the state-of-the-art in inpainting, and curtail the propagation of weak baselines into research, standardized training and testing datasets of images and masks, should be established.

- **Algorithms**. To enable reproducibility of the work, there is need for the transparency of the experiments by reporting the number of epochs and parameters for each method. This will allow future work to benchmarking against the baseline models. Current algorithms train on specific dataset and only work on the data with similar nature. Future work should explore a generalisable algorithms that can work on any image type.

- **Necessity of standardised performance metrics**. There are no standardised metrics to evaluate the performance of the algorithms. A recommendation will be for future research to report $\boldsymbol{\ell}_1$, $\boldsymbol{\ell}_2$, PSNR and SSIM as these metrics reflect different aspects of the performance.

- **Human Visual System**. The human visual system should be taken into consideration in more subjective evaluations to evaluate the perceptual quality of inpainted regions. For example, this can be in the form of Mean opinion score or measuring direct gaze information by direct bearing and the subjective rating of the observer.

Finally, deep learning methods have proven to be extremely powerful when compared to traditional methods. It is expected that research efforts to propose novel inpainting algorithms will increase in this area of study in the future.

2.8 Summary

As previously said, deep learning approaches have shown to be highly effective when compared to older inpainting methods. It is expected that research efforts in this area of study will continue to develop in the future, since it is a steadily expanding field with a large number of research studies offering various approaches to image inpainting. To generate realistic high-resolution face images, the work presented in this thesis will primarily focus on overcoming the limitations identified in this chapter. As a result, the ideas and approaches that are important in the field of GANs to complete this demanding task successfully are carefully considered. The primary focus will be on the generation of synthetic faces through the inpainting of masked images using the deep learning approach. However, the proposed models will be extended to inpainting on natural scene images in order to generalise them and demonstrate their robustness in inpainting. The preliminaries, ideas, and procedures that served as the foundation for the models that you will learn about in the subsequent chapters will be discussed in the following chapter 3. To summarise, the research community in painting should embrace more rigorous and improved practices in terms of reproducibility, assessment, and computing complexity reduction in order to attain more efficacy.

Chapter 3

Preliminaries, Background Techniques and Intuition

This chapter describes the preliminary steps, background information, theory and techniques that led to the design of the proposed algorithms in the contribution chapters. Furthermore, it helps the reader have a better grasp of what is going on behind the scenes of the proposed models.

3.1 Introduction

Our assumptions include prior knowledge of deep learning algorithms that utilise adversarial training techniques (GAN) [62]. In a wide range of significant machine learning and computer vision techniques, generative models are a critical component. Recently, generative models have been increasingly popular for inferring the statistical structure of high-dimensional data, which may be used to generate many types of fake data, including realistic images, movies, and audio signals, among others. These models are currently being investigated for semi-supervised and representation learning [157, 171] conditional GAN [148], text-to-image synthesis [158], image super-resolution [114, 204], sketch-to-image [137], inpainting [152, 124], image enhancement [230], style transfer [57, 94], image-to-image translation [85, 244], amongst other applications. GANs [62] are vastly studied type of generative models in computer vision tasks because of their superiority in producing high-quality images. During this study, several GAN algorithms were investigated, both in terms of network design phases and network architecture.

3.2 Preliminary Research

First, facial inpainting methods are investigated in detail from Chapter 2, with our main focus on the network design and architecture.

This section provides an understanding of the preliminary considerations that lead to the choice on how the models proposed in this thesis are designed. The contributions of this thesis are designed to focus on end-to-end learning using GANs, based on the original introduction to inpainting by Pathak et al. [152], and to build on that foundation. As previously said, the model [152] is the first to use GANs to generated realistic inpainted images, and it is also the most widely known inpainting method in this area of research. A variety of optimization theories based on the context-encoder [152] and prior inpainting models have been applied in this study. This research, however, did not stop with the study of inpainting GAN models; it also investigated the foundations of various architectures in order to gain a thorough understanding of the methodologies used within CNN layers. Because inpainting is a process of restoring damaged pixels, it is important to consider designing a deep learning model capable of generating the missing regions in a way that is consistent with the rest of the image while retaining the realism of the original.

It was observed throughout our research that various models stack their networks in a different way and utilise different parameters to obtain the best possible outcomes. This is a significant finding. I delve a little further into the convolution block of a CNN model, using the MNIST dataset [113] shown on Figure 3.1 to help us through the design process. This chapter provides an optimization view of common activation

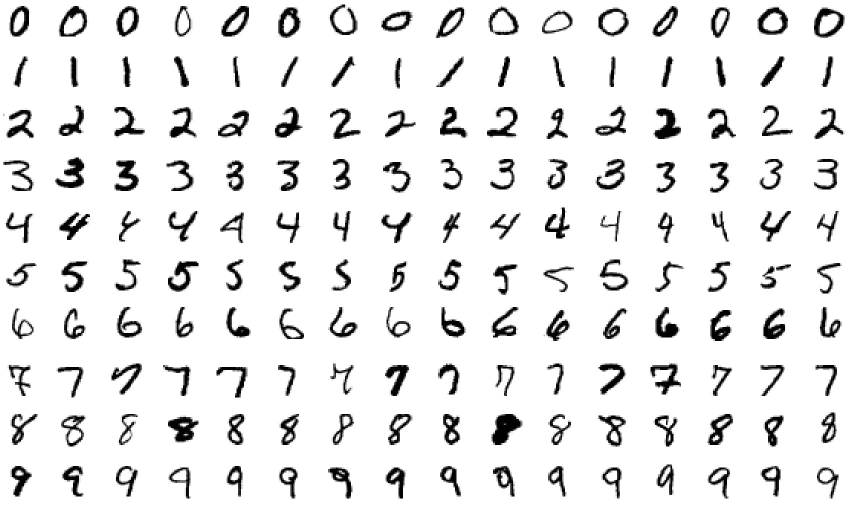


Figure 3.1: Examples from MNIST test dataset [43].

functions used by deep learning models using the MNIST dataset. It also investigated the batch normalisation layer within the convolution and tested various parameter settings. It then investigates the effect of layer pooling in greater depth, and apply the findings to one of the proposed approaches in order to improve its contribution. Before proceeding with the experimental analysis, a brief review of some cutting-edge methods used in machine intelligence performance for the generation of synthetic data is provided. It will look at three designs that served as inspiration for the design of the proposed contributions to this thesis, which will be discussed further below.

3.3 Learning to Generate Synthetic Data

Machine learning refers to a variety of techniques that use a dataset to make intelligent predictions [146]. A excellent example of one type of technique is a predictive model, in which a well-designed algorithm is trained to generate predictions using sampled data, i.e. data having a known link between the input and output. Predictions are made here by utilising new data for which the output is unknown [22]. Another example of machine learning are generative models [104]. As already mentioned in Chapter 1 and 2, generative models are a critical component of machine learning and computer vision techniques. There are two major kinds of generative models: explicit and implicit methods. The former class presupposes access to the model likelihood function, whereas the later class generates data via a sampling method. Explicit models include variational auto-encoders (VAEs) and PixelCNN whereas GANs(as shown on Figure 3.2) are examples of implicit generative models. Typically, explicit models are trained via maximization of a lower bound or likelihood. Using a parameterized model distribution Q, GANs attempt to approximate a data distribution P. They do it by optimising two adversarial networks concurrently: a generator and a discriminator. The generator G through training learns to generate images that are near to the real data distribution from a random noise vector. The discriminator D role is to differentiate properly between the generated images and the ground-truth images from the data distribution. Empirically, the training procedure in GANs is a minmax game with two players and can be summed up with the objective function below;

$$\min_G \max_D V(D,G) = \mathbb{E}_{x \ P_{\mathrm{r}}}[\log D(x)] - \mathbb{E}_{z \ P_{\mathrm{g}}}[\log(1 - D(G(z)))]$$

where x is real data distribution from ground-truth data P_r and z is the noise vector, P_g is generated data distribution sampled from a uniform or Gaussian distribution. GANs have demonstrated a remarkable capacity for generating realistic high-resolution images [98]. For this reason, many variants of GANs have been introduced to target specific task. GANs is the state-of-the-art method for predicting the facial appearances at all stages (baby, young and old) on a facial image [235, 11, 221]. All of these designs use compressed data in latent space to achieve a desired outcome based on a hypothesis. The postulation is that in a defined latent space, there are possibilities within the designed model where a specific series of tensor operations map input to output data either in a supervised or unsupervised manner. Therefore it is possible to perform challenging tasks using guidance from signal feedback using data to learn useful representations to obtain an expected outcome. This explains why it is possible to perform challenging tasks that could have only been done by humans using guidance from a feedback signal (e.g. A painter, hand painting a drawing from a reference image or deteriorated painting). The problem most methods have is that the cost function needs to be manually specified or fine-tuned during training to achieve the desired outcome. This makes it very difficult for latent representations because it can either slow down learning or result in exploding gradients. As a result, caution must be exercised during the network's design phase. However, the handler and the task determine the network's design, which can be either supervised or unsupervised learning.

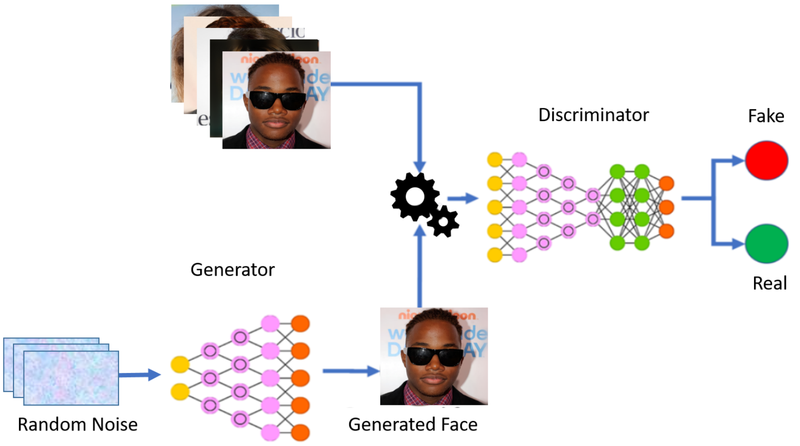


Figure 3.2: The GAN framework composing of a generator and discriminator. The generator input is random noise vector and the generated samples are fake synthetic samples. The discriminator takes both real and fake samples to judge whether they are real or fake.

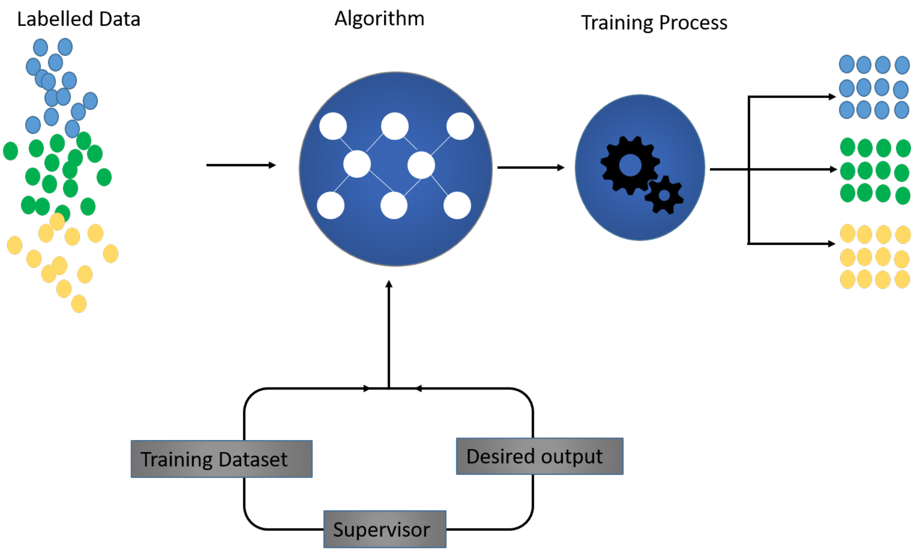


Figure 3.3: Supervised machine learning.

3.3.1 Supervised Learning:

As the name implies, supervised machine learning (as shown on Figure 3.3) involves training a model using a set of input data that already has been linked with the correct output [146]. The algorithm makes predictions based on the training dataset and it is corrected by the desired output data or label. A label refers to a tag attached to each example in a dataset. This tag now becomes the answer the algorithm should produce on its own. E.g a labelled data of air-planes will tell the algorithm which images were Concord, Boeing, etc. If the algorithm is shown an image, it compels the model to compare with the training set to predict the correct label. In image classification, regression and image inpainting tasks supervised machine learning is increasingly often employed. In classification, an algorithm predicts a discrete value, that identifies a member of particular class from an input data. i.e from a dataset, of class animal, each animal has a label, e.g elephant, lion, zebra etc. So the algorithm correctly classifies the images into their different classes. Regression problems are continuous data i.e. given a value x, provide the expected value of the y variable. In inpainting task, the input image is a masked image and the reconstruction task is to train the algorithm to generate images with predictions of the missing regions that match the ground-truth counterparts and the entire image. The pioneer approach to inpainting was by [86] who employed autoencoders to solve the inpainting problem as already mentioned in Chapter 2. Autoencoders use stacked CNN layers to extract relevant features needed for image reconstruction. Deep learning models, which as the name implies mean more layers of CNN blocks employ the same technique within the generator and some combined with a discriminative model to complete an inpainting task. In addition, the convolutional operation is explained, and some relevant techniques that are critical to the design of our models are explored in Chapters 4, 5, 6, 7. Furthermore, all techniques are not included here because they are unrelated to the contribution of this thesis. This is not to say that other techniques not mentioned here are irrelevant or ineffective in inpainting.

3.3.2 CNN Layers

Deep learning methods use convolutional operations for feature extraction through the process of convolution within a CNN block as shown on Figure 3.4. A convolution is a mathematical filtering operation that maps features based on two functions, with a third function as an expression of how the shape of one modifies the other. The convolutional operation is capable of computing a multidimensional array of input data using a multidimensional array of parameters. This is accomplished by the use of a kernel that implements an infinite summation over a finite number of array members. This is achieved as the Convolutional layers convert these three pieces information as tensors. That is the input tensor, filter tensor and the output tensor as shown on Equation 3.3.2.

$$S(i,j) = (I * K)(i,j) = \sum_{m}^{+\infty}\sum_{n}^{+\infty} I(m,n)K(i-m,j-n)$$

where S is convolution operation, I is a 2D image and K is the filter (kernel). The convolution operation does not use linear operations on the whole image at once but selects small sections of the image to condense and detect patterns. As a result different types of patterns are detected based on the type of kernel used. Thus the reason for this preliminary work as the purpose of this thesis is to inpaint missing portions with contextual information which is perceptually consistent with the entire image. Another importance of this study is the parameter accretion in deep generative models. For example the size of the kernel matters and the total number of filters to be learned in each convolution operation combined with other parameters (stride and padding size) decide the model weight which is critical to computational resources.

It is important to consider the convolutional operation parameters stride and padding while constructing image inpainting methods. As a result, the stride parameter defines how many pixels the filter will slide across during each pass. The filter will move one pixel at a time if stride is set to 1. When stride equals 0, the filter

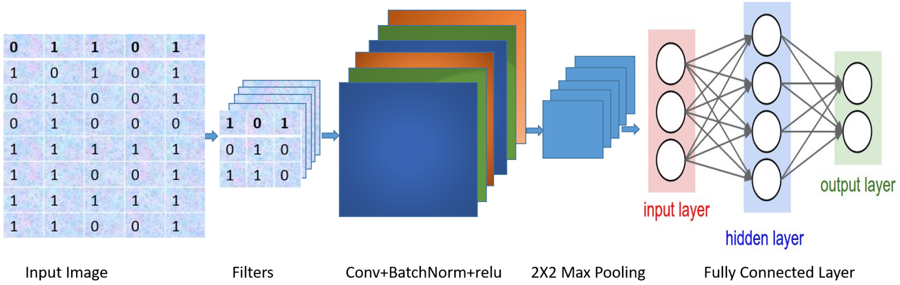


Figure 3.4: Visualisation of simple CNN. The layers are stacked in the format presented on here. The fully connected layer in our proposed models are removed. Please see chapters 4,5,6 and 7 for full details of our models.

does not move at all, resulting in an error. Padding, on the other hand, is employed to cushion the input volume all round the border in order to control the output volume's size. Thus, zero padding will pad zeros around the output volume's boundary, providing for greater control over the output volume's size.

3.3.3 Dilated Convolution

The dilated convolution operation was introduced by [225] as a modified kernel of the convolution operation. The dilation operation supports exponential expansion of the receptive field without loss in resolution and without parameter accretion. Based on the modification, the receptive field are set to grow exponentially while the number of parameters grow linearly. For more detail refer to the original work by Yu et al. [225].

Figure. 3.5 shows an illustration of dilated convolution. The yellow dots indicate the dilation rate and the green area is the receptive field at various sizes based on the dilation rate (r_d).

Figure 3.6 shows the process of dilated convolution. Dilated convolutions [225], combined with skip connections, are critical to the design of the contribution chapters of this thesis as:

- It broadens the receptive fields to capture more contextual information without parameter accretion and computational complexity, which are preserved and transferred by skip connections to corresponding deconvolution layers.

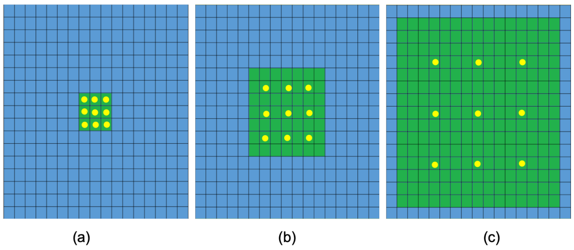


Figure 3.5: An overview of the systematic operation of dilated convolution with the expansion of receptive field without loss in resolution. (a) is has a dilation rate of 1 showing each element with a receptive field of 3×3. (b) shows a receptive field of 7×7 based on $r_d = 2$. (c) has a receptive field of 15×15 on $r_d = 4$. The accretion of receptive field is in linearity with the parameters. [225]

- It detects fine details and maintains high-resolution feature maps, and achieves end-to-end feature learning with a better local minimum (high restoration performance).
- It has shown considerable improvement of accuracy in segmentation task [225, 31, 32].

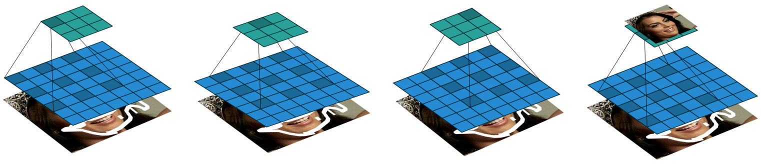


Figure 3.6: Illustration of dilated convolution process. Convolving a 3×3 kernel over a 7×7 input with a dilation factor of 2 (i.e., $i = 7$, $k = 3$, $d_r = 2$, $s = 1$ and $p = 0$) [48]. The accretion of receptive field is in linearity with the parameters [225]. A 5×5 kernel will have the same receptive field view as over a 7×7 input at dilation rate=2 whilst only using 9 parameters over a 512×512 input.

Because the goal is to achieve realistic results on high resolution images, dilated convolutions are investigated and incorporated into the design of proposed models in the contribution chapters 4, 5, 6, 7. This, however, is easily accomplished if a function is included that can preserve some features and transfer them to corresponding layers within the network.

3.3.4 Skip Connections

Many deep learning methods [163, 73, 216, 166, 78, 47, 117] use skip connections. Skip connections (Figure 3.7) as the name implies is an implementation within the network

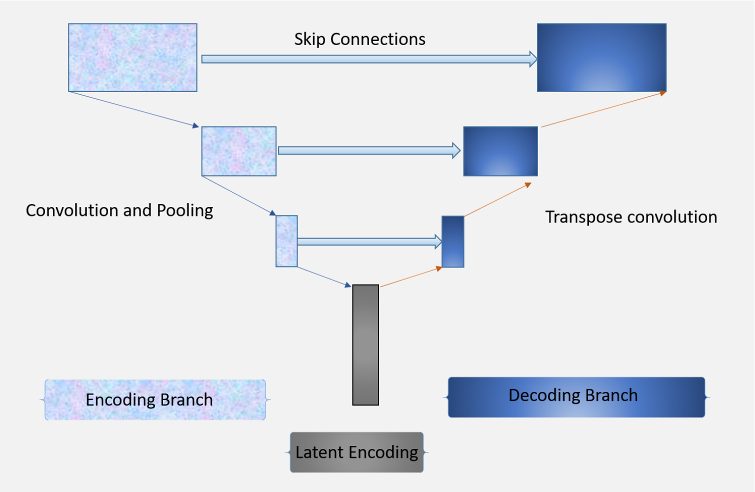


Figure 3.7: Illustration of skip connection between blocks.

that skips some layers to feed the output of one layer as input to the next layer. Skip connections are commonly used in Encoder-Decoder architectures, and they assist the model in generating near accurate results by transferring appearance information from shallow layers of the encoder to the corresponding deeper layer of the decoder (generator). The UNet is the most widely used Encoder-Decoder architecture, and the LinkNet is also widely used. The manner in which these structures combine the appearance of information from the encoder layers with that of the decoder layers differs slightly. In the case of the U-Net, incoming features (from the encoder) are concatenated in the decoder layer. LinkNet, on the other hand, performs addition. As a result, LinkNet structures require fewer operations in a single forward pass and are significantly faster during training than U-Net structures. Skip connections stepped into deep learning models as a result of vanishing gradient problem. The vanishing gradient problem results from when the training loss stops decreasing when it is a long way from the desired value. The vanishing gradient problem may occur from the backpropagation algorithm (loss). The backpropagation algorithms' goal is to assist the model during learning by optimizing a number of parameters iteratively with respect to the training loss function. The loss function is usually defined based on the problem to solve and it is the quantitative measure of the distance between two tensors [166, 2]. These tensors can be a representation of an image, label or translated text or numbers depending on the task. Thus backpropagation helps to gradually minimize the loss as it updates the weights of the network. Backpropagation calculates the gradient in the loss function by using the chain rule to compute the with respect to a neural network parameter. Skip connections provide an alternative path to backpropagation. According to recent approaches [238, 229], skip connections are additional paths that benefit network convergence. There are two ways to use skip connections; either via addition or concatenation. The alternative path to backpropagation with skip connection via addition is to use the identity function by simply adding a vector. Thus the identity function preserves the gradient as the layers go deeper. The alternative of skip connections is by concatenation of previous feature maps. This is usually because of the low-level information that is shared between the input and output. Thus passing this information using a concatenated feature map of the previous layer ensures maximum information flow between layers of the network. There are two types of skip connections via addition namely long skip connections and short skip connections. Short skip connections are feature map additions within consecutive convolution layers that have no impact on the input dimension. A good example of short skip connections are proposed in ResNet [73]. Long skip connections are associated with encoder-decoder frameworks, such as U-Net [163]. With this type of feature map addition, the global information is preserved while the local information provides intuition to the model in the form of image patch details (e.g in the case of inpainting). Based on research in 2, it is observed that skip connections provide uninterrupted gradient flow between convolution layers and enables feature reusability.

3.3.5 Unsupervised Learning:

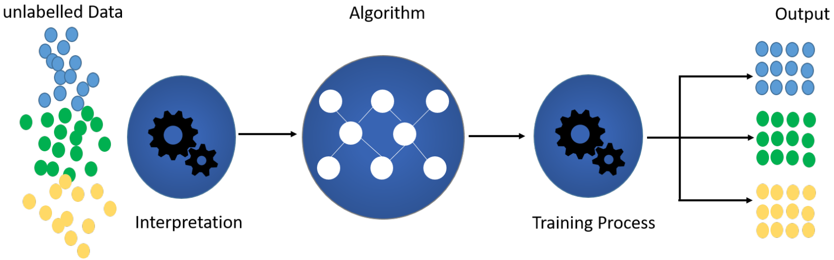


Figure 3.8: Unsupervised machine learning.

In this type of learning problem, the algorithm is presented with input data with no instructions on what to do as illustrated on Figure 3.8. However, two independent sets of data with no paired example of the data can be translated to a corresponding output. This means that a collection of examples from the training set have no desired outcome or correct answer. This is where the neural network automatically extracts useful features to analyse and produce structural information or data similar to the structure represented on the input data. For example the use of autoencoders to compress input data into code and recreate the input from summarized code in latent space. This makes training difficult but more applicable in real life scenarios.

3.3.6 Siamese Network

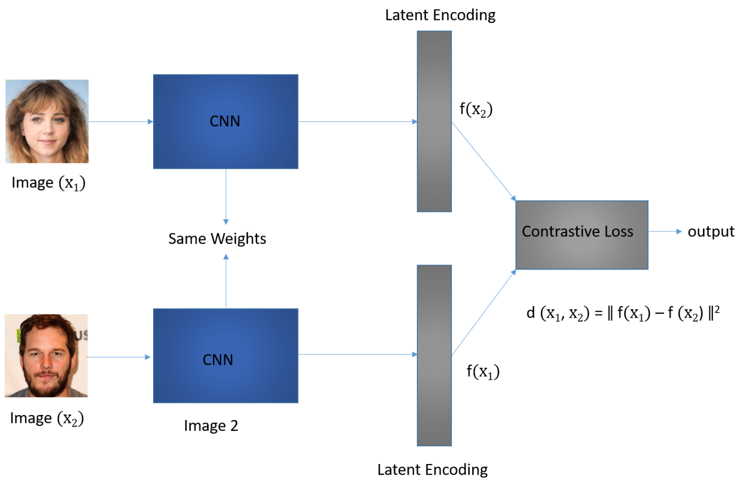


Figure 3.9: Siamese Network.The function d is used to tell how similar or different the two faces are.The latent encoding of the fully connected layers are denoted f(x1) and f(x2) which are good representations for the two images.

Learning similarity metric from data is well known in computer vision [38, 71, 139]. Figure 3.9 shows similarity learning between two encoders based on a contrastive loss function. This type of model can be used for recognition or verification of applications. Traditional approaches use discriminative techniques in neural networks or support vector machines for classification of a number of categories from a known training sample. However, these methods have a limitation which is their inability to handle samples with large categories, thus making it unsuitable for applications with very large categories.

3.4 Preliminary Experiments

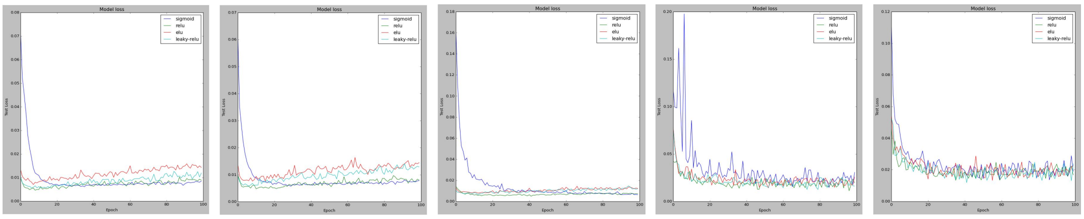
3.4.1 Batch Normalization (BN)

In previous approaches for facial manipulation algorithms, Batch Normalization (henceforth BN) is introduced within the hidden layers to normalize the input layer, thus making sure the hidden layer is on the same scale. This proved to speed up learning during training [152]. Another advantage for using BN is that it tackles the mean and variance for each feature during training. In a more recent approach to image inpainting, Facial attribute manipulation, it is observed that BN are not part of the implementation. An explanation for this is that noise introduced by BN layers reduced the performance of the model [105]. To address this, most of these methods[238, 243] have used Instance Normalization or not introduced BN at all. However, it is observed that BN layers if introduced after the activation layer is harmful during training. Also, another observation was that the negative impact on BN on the performance of the model is during testing not training when introduced after the activation layers. BN layers deteriorate colour coherency [82] during testing, hence should be used in the upper convolutional layers to reduce computational load. The slow-moving statistics in BN layers can cause the output of each layer to shift slightly during evaluation, thus making the output to be slightly different. This problem is handled by modifying the momentum parameter of BN layers to allow for running statistics to catch-up with batch statistics. Also, a regularization technique is utilized to scale the parameters in BN layers to correct small mismatch in statistics so that are not amplified by BN layers.

3.4.2 Activation Functions

GANs are best described as optimization problems. This means that GANs are designed with the capability to find the optimal or best mathematical solution to a task. For better optimization of neural networks, one would require an explicit closed-formed solution to constrain the weights of the network during training.

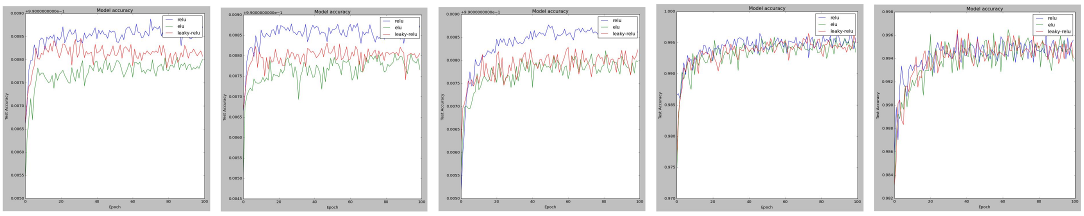
- **NoBN**: One convolutional layer with maxpooling, fully connected layer and no BN.
- **WBN**: Two convolution layers with BN, momentum=0.8. Note Activation function is before BN layer
- **AABN**: Activation function after BN with two layered convolution. momentum=0.8
- **FAABN**: Full model with activation after BN. Default values of BN are used within three layers of convolution, and maxpooling in between.
- **BN=**0.8: Full model with activation after BN. BN momentum=0.8 and three layers of convolution, and maxpooling in between.



(a) **NoBN** (b) **WBN** (c) **AABN** (d) **FAABN** (e) **BN**=0.8

Figure 3.10: Loss variation depending on number of layers and position of activation function within the model.(a) BN is before the activation. Note that in the second experiment, (b) the BN layer is stacked below the activation function. BN momentum set at 0.8 (c) Two convolution layers. BN momentum set at 0.8 (d) four convolution layers with default parameters of BN. categorical cross entropy

The goal of this section is not to learn or rediscover existing activation functions but to provide an inside to the decisions that led to the design of the algorithms proposed in subsequent chapters. An activation function can be described as a transfer function that determines the output node of a convolution layer or neural network in values ranging between [0,1] or [-1,1], based on a weight sum of the input [97, 3, 173]. That is, it limits the output of a neuron by squashing the amplitude values between 0 and 1 or -1 and 1. Activation functions are either linear or non-linear functions. However, different parameters have been used within some functions based on the study conducted on activation functions with the MNIST dataset. I have follow similar method as in [153] to have a guide on what to expect. The experiment conducted here is an integration of the first three traditional convolution layers mostly used by Facial Inpainting algorithms, combined with the traditional fully connected layer to solve the MNIST problem. Figure 3.11 shows that test accuracy of the

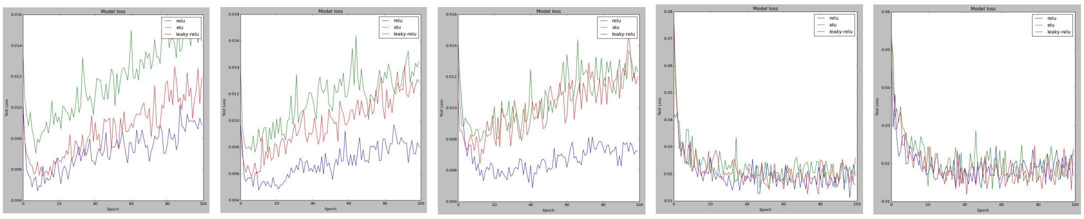


(a) **NoBN** (b) **WBN** (c) **AABN** (d) **FAABN** (e) **BN**=0.8

Figure 3.11: Comparison of accuracy between activation functions (ReLU, ELU, LeakyReLU) (a) BN is before the activation. Note that in the second experiment, (b) the BN layer is stacked below the activation function. BN momentum set at 0.8 (c) Two convolution layers. BN momentum set at 0.8 (d) four convolution layers with default parameters of BN.

MNIST. With a full model as in (d) with default parameters of BN, an accuracy of

0.99 is achieved. With fewer layers and BN=0.8 the accuracy is slightly lower. With changes to BN momentum=0.8, the test accuracy is 0.99 though not very stable as compared to the (d). The results obtained show very marginal improvement on the different activation functions. The main point of the comparison is to illustrate how these layers can be stacked within our proposed networks to assist with efficient propagation of the weights through the layers. Examples of activation functions mostly used by Facial manipulation algorithms are ReLU, ELU, LeakyReLU [39, 147]. For example, SoftMax is specifically used as the output of the fully connected layer and varied LeakyReLU(alpha=0.2) and changes to BN layer between experiments as well. As the experiments continue, the convolutional layers are increased to observe the performance of the model. The choice depends on the task and how the algorithm designer have stacked the network. This is usually based on what they want to achieve in the convolutional output or what optimization they do during training. Based on our experiments, some hyperparamters (value used to control the experiment) are used within the convolution layers and stack in a way that could change the activation function during training. These network variants are trained with the different activation functions mentioned here and used the history generated by each is used to plot its accuracy over epochs. Figure 3.12 shows that test loss results are similar for the first two experiments. There is a slight error in fourth experiment in (d). This may be due to the default parameters of BN used. In (e) and (c) the same BN parameter is used but increase the convolution layers in (e).



(a) **NoBN** (b) **WBN** (c) **AABN** (d) **FAABN** (e) **BN=**0.8

Figure 3.12: Comparison of losses on between activation functions (ReLU, ELU, LeakyReLU). Note that in the second experiment, (b) the BN layer is stacked below the activation function. BN momentum set at 0.8 (c) Two convolution layers. BN momentum set at 0.8 (d) four convolution layers with default parameters of BN. A good evaluation is expected to have a constant curve of the losses lower across all activation functions for all transformed samples with a corresponding high accuracy in performance of the model.

3.4.3 Objective functions

One of the hardest task for deep learning based models is to find a suitable objective learning function needed to backpropagate the weights in order to find an optimal solution. This is because the neural networks can be highly confident even when it is wrong. In this sense, an objective function is computed with model parameters as arguments to evaluate and return a number. By computing this function, the model finds the values of these parameters that either maximize or minimize the returned number. Consider the model parameters to be weights; the purpose of the algorithm designer is to employ an objective function capable of effectively exploring the latent space with a range of potential values that can steer the model to an optimal solution. At each stage of the network during training, the weights needs correction to indicate to the model how far it is from the results. Hence the importance of the objective function which provides the basics and formal specification of the problem. With most GAN task, the goal is to minimize the error. Hence when minimizing, the term loss function, or error function [61]. As previously described, the objective function is determined by the problem formulation, thus in the case of our study, the objective function is termed loss function. For facial manipulation algorithms a suitable content loss function is needed to assist the model during training. This is because the model needs to gear towards perceptual similarity. In image inpainting, large hole region need filling-in and needs a much deeper semantic understanding of the image, hence a reconstruction loss is needed. This is because during back propagation, weights need to be updated properly to allow the model to learn from nearby pixels. Thus, when training a model, it is important to provide the model with statistical understanding of the image so that it can learn to generate plausible hypothesis for the missing regions. Hence the introduction of GANs to inpainting by Pathak et al. [152]. With GANs, the reconstruction loss captures structural and contextual information about the missing region, whilst the adversarial loss selects specific modes from the distribution to provide realistic results. This will provide the model the ability to be able to synthesize images with high-level features. For more details on the loss functions used in this thesis, see subsequent chapters.

3.5 The use of Wasserstein Distance as a GAN loss function

In the field of generative models, the log-likelihood (or, more precisely, Kullback-Leibler divergence) has been considered the gold standard in the training of GANs [62, 193, 20]. It assesses the likelihood of the real data occurring under the generated distribution on a number of samples (N) taken from the data set. Mathematically, this can be represented as $L = \frac{1}{N}\sum_i logP_{model}(X_i)$. Because it is not possible to estimate likelihood in higher dimensions, generated samples can be utilized to estimate anything about a model's log-likelihood in one dimension. A model with maximum likelihood (zero KL divergence) is thought to yield flawless samples. However, the log-likelihood is very intuitive but its stability has been questioned by [191]. The authors showed that, the probability does not tell anything about the sample's quality. In other words, sample quality and log-likelihood are unrelated. That is a models's log-likelihood might be bad while producing outstanding samples and vice versa. A more detailed proof can be seen in [191], where a mixture of Gaussian distribution training on images will generate amazing samples but have poor log-likelihood. Furthermore, if the GAN discriminator operates as predicted, the gradient of the loss functions begins to decrease and finally approaches zero. As a result, the model has difficulty updating the process loss, resulting in sluggish training and the model being stuck. Another possibility is that the discriminator behaved poorly, resulting in erroneous feedback to the generator and leading the loss function to not correctly represent reality.

GANs struggled to find an objective function that could better evaluate the training process. However, a good evaluation measure was needed, and the Wasserstein Distance [9] was sought to solve it. The Jensen-Shannon divergence optimised in GANs is not continuous with respect to the generator parameters, as proven by [9] when the model distribution and data distribution have disjoint supports. Instead, they recommended that the Earth Mover distance or Wasserstein-1 distance be used as a substitute. The Wasserstein critic [9] was proposed to estimate the Wasserstein distance between real and generated data distributions. The Wasserstein distance (Earth Mover's Distance (EMD) is the minimal mass displacement required to change one distribution to another [20]. Empirically, the WGAN shows improved stability during optimization of a neural network. The approximation is on the norm of the discriminator (critic) with clipped weights considered to have bounded derivatives based on a Lipschitz continous function. It can be derived using the equation below.

$$W(P_r, P_g)\alpha \max_f \mathbb{E}_{x\ P_r}[f(x)] - \mathbb{E}_{z\ P_g}[f(z)], \tag{3.1}$$

where $f : \mathbb{R}^D - > \mathbb{R}$ is the Lipschitz continous function, f is the neural network, P_r is the real data distribution and P_g is the generated data distribution. To be more elaborate, a dual formulation by [9] based on discriminator D and Generator G

$$\min_G \max_D V(D, G) = \mathbb{E}_{x\ P_r}[D(x)] - \mathbb{E}_{z\ P_g}[D(G(z))]$$

where D is set on 1-Lipschitz function and z is the noise vector. The Wasserstein distance computes the statistical similarity between local patches derived from Laplacian pyramid representations of real and generated images [98, 99]. The advantage of the Wasserstein GAN is that it solves the problem of both overfitting and mode collapse. Thus, provided the generator remembers the training set, the trained critic would be able to distinguish the generated samples from the real ones. Furthermore, when two distributions do not overlap, it does not saturate. The distance that separates the samples and the data demonstrates how simple it is for the critic to differentiate between the samples and the data. When computing the base distance in an appropriate feature space, the Wasserstein distance is a reliable metric. The high sample and time complexity of this distance is a significant limitation to the WGAN model.

3.6 Datasets

In this section, the datasets used for the contribution chapters are introduced. All the contributions of this thesis are evaluated qualitatively and quantitatively against the state of the art methods. The most commonly used publicly available datasets for facial image inpainting are the CelebA [131] and CelebA-HQ [98]. To create a damaged image or images with missing regions, a binary mask must be applied on the image to simulate the damage. Usually this is done by an external dataset or function to create these masks regions. In this thesis, an external dataset namely Quick Draw Mask dataset [84]. To utilize semantic segmentation mask, the CelebAMask-HQ dataset proposed by Lee et al. [115] is considered for one of the models in the contribution chapters. Our experiment focuses on high-resolution face images and irregular binary masks. The benchmark dataset for high-resolution face images is CelebA-HQ dataset [98], which was curated from the CelebA dataset [133]. Figure 3.13 shows a few samples from the CelebA-HQ dataset. In this thesis' experiments, high-resolution images from the CelebA-HQ dataset [98] were resized to 256 × 256. For masking method,

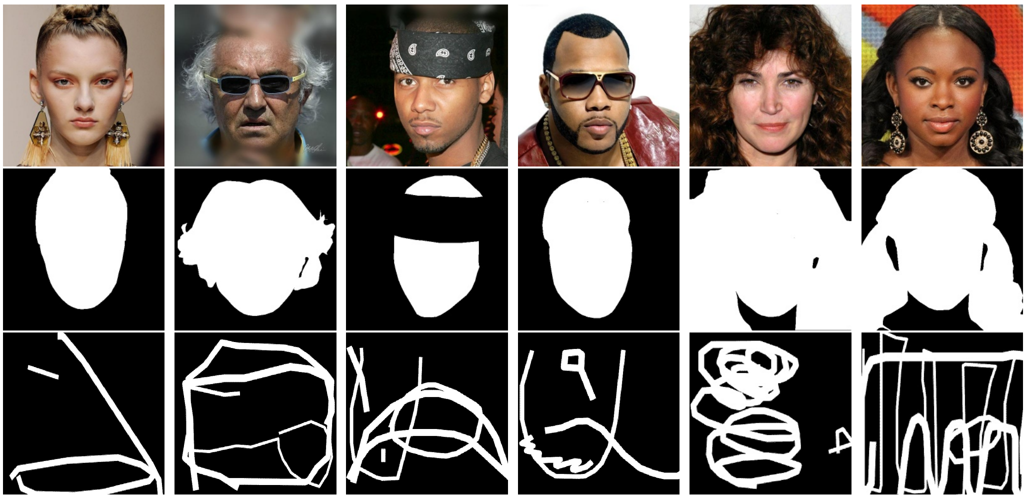


Figure 3.13: First row are images from CelebA-HQ [98]. Second row shows the segmentation masks of face and hair regions from CelebAMask-HQ [115] used as our foreground masks in Chapter 6. The skin region without hair indicates the subject's hair is short or has no hair. The third row is the mask dataset from [84] used as masking method for training the models in the contribution chapters

the Quick-Draw irregular mask dataset proposed by Iskakov et al. [84] which has up to 50000 binary masks samples for training with hole-to-image ratios ranging from [0.01, 0.60] are used. Both the masks and the images are resized and merged together by matrix operations to obtain an input masked image. The CelebAMask-HQ dataset is used for the extraction of foreground segmentation masks to compute the loss function for our model. The semantic segmentation maps are fined-grained mask annotations of all the facial attributes including the skin and cloth making up 19 labels. This dataset is used in the case of the model proposed in Chapter 6, but only the skin and hair regions are extracted to form the foreground mask. Figure 3.13 shows the foreground regions of drawn from the CelebAMask-HQ dataset.

The Places2 [241] dataset contains more than 10 million images with more than 400 unique scene categories with 5,000 to 30,000 train images. The training and testing sets are split based on the state-of-the-art [152] and trained our network to understand scene category. The Paris Street View has 14,900 training images and 100 validation images, with the same testing set as described in the state-of-the-art [152] for the experiments.

3.7 Summary

This section compares the various techniques that are critical to the proposed contributions in the following chapters. The parameters that comprise the CNN layer depicted in Figure 3.4 were investigated. To learn how activation functions behave, different experiments were carried out by stacking the convolution layers with different parameters. It has been discovered that a neural network may require more parameters within the convolution layer, and that the results are influenced by where the activation is stacked. Another finding is that small parameter values are required to tune the neural network. Furthermore, depending on the algorithm designer and the task at hand, different activation functions can be applied in each feature map during neural network training. So far, from Chapter 2 to the present, measures introduced for GAN inpainting algorithms have been investigated. Techniques were also investigated to aid in the design of the models proposed in this thesis. However, the best way to design these algorithms remains an unsolved problem in this field of study. As a result, our proposed solutions are designed to be open to criticism and fair comparisons in order to improve and develop new GAN inpainting models. Finally, the datasets used to run the experiments in this thesis' contribution chapters have been discussed. It is important to emphasise that the lack of a universal metric for evaluating GANs has hampered the elimination of weak baselines. A proposed protocol in Chapter 6 could, however, contribute to a universal empirical evaluation technique for GAN inpainting models. The chapter that follows will describe the first contribution to the research in this thesis, and subsequent contributions will build on this work to demonstrate gradual improvement to the inpainting problem throughout all stages of this research.

Chapter 4

Symmetric Skip Connection Wasserstein GAN for High-Resolution Facial Inpainting

This chapter describes SWGAN, a high-resolution face inpainting network architecture. An architecture (encoder-decoder) that incorporates dilated convolutions with skip connections at multiple levels and is trainable end-to-end with a Wasserstein discriminator as a deep network. The feature extractor (encoder) extracts abstractions from the masked image to learn end-to-end mappings. The learned abstractions are used in the decoder to reassemble high-resolution images as generated output. In addition, dilated convolutions and skip connections are investigated to demonstrate how they complement each other in inpainting. This chapter appears in ***Symmetric Skip connection Wasserstein GAN for High Resolution Facial Inpainting*** *VISAPP 2021.*

4.1 Introduction

This chapter focuses on high resolution facial image inpainting. Instead of improving on traditional inpainting methods that deploy PDEs to interpolate neighbouring pixels for patch transfer or diffusion, the approach with GANs is considered. A perceptual objective learning function is contrained to target the pixel values during optimisation. The state-of-the-art facial image inpainting methods achieved promising results but face realism preservation remains a challenge. This is due to limitations, such as failures in preserving edges and blurry artefacts. To overcome these limitations, the Symmetric Skip Connection Wasserstein Generative Adversarial Network (S-WGAN) for high-resolution facial image inpainting algorithm is proposed. The architecture is an encoder-decoder with convolutional blocks, linked by skip connections. The encoder is a feature extractor that captures data abstractions of an input image to learn an end-to-end mapping from an input (binary masked image) to the ground-truth. The decoder uses learned abstractions to reconstruct the image. With skip connections, S-WGAN transfers image details to the decoder. Also, skip connections ensures high quality local optimum and enables the model converges faster with deeper networks, thus making them easy to train in consequence to achieving high restoration performance. Additionally, a Wasserstein-Perceptual loss function to preserve colour and maintain realism on a reconstructed image is introduced to minimise the error between the real and reconstructed image. This model is evaluated and compared with the state-of-the-art methods on CelebA-HQ dataset. Our results show S-WGAN produces sharper and more realistic images when visually compared with other methods. The quantitative measures show our proposed S-WGAN achieves the best Structure Similarity Index Measure (SSIM) of 0.94. However, these approaches often fail to produce images with plausible visual semantics. With the evolution in research, deep learning-based methods [152, 124, 82, 226, 216, 217, 151] encode the semantic context of an image into feature space and fill in missing pixels on images by hallucinations [217] through the use of generative neural network. Although deep learning approaches achieve excellent performance in facial inpainting, there are some limitations of state of the art as illustrated in Figure 4.1. These are cases, where Figure 4.1(a) shows poor performance on holes with arbitrary sizes; Figure 4.1(b) illustrates the lack of edge-preserving using the existing technique; Figure 4.1(c) depicts the blurry artefacts; and Figure 4.1(d) demonstrates the poor performance on high-resolution images and image completion with mask at border region.

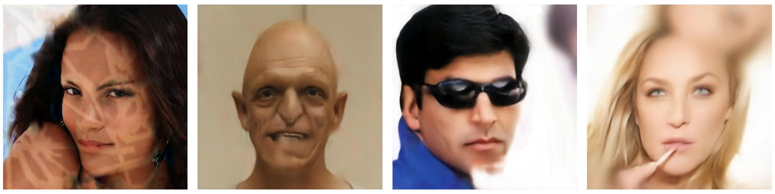


Figure 4.1: Images showing some issues by existing methods [152, 137]: (a) Poor performance on holes with arbitrary sizes; (b) Lack of edge-preserving technique; (c) Blurry artefacts; and (d) Poor performance on high-resolution images and image completion with mask at the border region.

To correctly predict missing parts of a face and preserve its realism, the S-WGAN has the following contributions:

- A newly designed framework with Wasserstein Generative Adversarial Network (WGAN) that uses symmetric skip connection to preserve image details.
- A newly defined combination loss function based on RGB and feature space.
- Experiment results show that the new combination loss trained with the S-WGAN model outperforms state-of-the-art algorithms..

4.2 Connection to Related Work

The closest to this proposed method is context-encoder approach by [152] and the inpainting algorithm in [82]. The context-encoder is designed to learn from context inspired by the introduction of the encoder-decoder framework with adversarial training [152]. The context-encoder uses an AlexNet [81] encoder with ℓ_2-loss and adversarial loss. However, despite the achievement of impressive inpainting results, this model still struggles with blurry artefacts, poor performances on masks of arbitrary structures, failures in edge preservation and poor performance on high-resolution images. This is because the context-encoder focuses on feature representation for inpainting and does not take into consideration these feature representations when differentiating real and fake inpaintings. The method introduced in [82] uses the concept of the context-encoder with structural changes to the encoder network designed to include dilated convolutions to expand the view of the receptive fields to capture more contextual information. The authors [82] also introduced two discriminators that differentiate localised inpainted regions of the generated and compared to its real counterpart, and the global fake and real regions.

4.3 Proposed Method

Our proposed model uses skip connections with dilated convolution across the network, to perform image inpainting. Because convolutions view missing pixels and valid pixels as one pixel level during inpainting, the important concept is to collect more contextual information. Furthermore, since most earlier studies on inpainting presume that missing parts are regular (e.g., a central mask or numerous rectangular holes), the convolutions as they glide across may readily remember specific windows for filling in missing pixels. As a result, the present design addresses this issue, despite the fact that it is just one level of pixel information. This is accomplished by conditioning the current convolution to capture large amounts of contextual information and using skip connections to retain valid pixels for corresponding mask regions during backpropagation. The architecture and loss function of S-WGAN model are discussed in the following sections.

4.3.1 Architecture

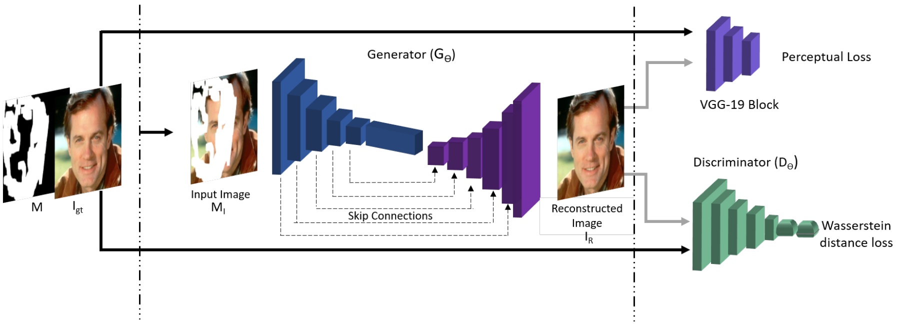


Figure 4.2: S-WGAN framework. The dilated convolution and deconvolution with the element-wise sum of feature maps (skip connection) combined with a Wasserstein network. The skip connections in the diagram ensure local pixel-level accuracy of the feature details to be retained.

Generator (G_θ) The effectiveness of feature encoding is improved by having an encoder of ten-convolutional layers, with a kernel size of 5 and dilation rate of 2, designed to match the size of the output image. This technique enables our model to learn larger spatial filters and help reduce volume [165]. Each block of convolution in exception of the final layer has Leaky ReLU activation and max-pooling operation of pool size 2×2. A dropout regularisation with a probability value of 0.25 is applied in the 4th and final layer of the encoder. The dropout layer randomly disconnects nodes and adjust the weights to propagate information to the decoder without overfitting.

Decoder The decoder are five blocks of deconvolutional layers, with learnable upsampling layers that recover image details using the same kernel size and dilation rate of the generator. The corresponding feature maps in the decoder are asymmetrically linked by element-wise skip connections to reach an optimum size. The final layer in the decoder is Tanh activation. Following the procedure mentioned in Chapter 3, dilated convolution is expressed based on network input. Using Equation 4.1, this equation can be stated in terms of the mask, the input image, and the dilation ratio.

$$I'_{(m,n)} = \sum_{i=1}^{M} \sum_{j=1}^{N} M_I(m + d_r i, n + d_r j) * \omega_{(i,j)} \tag{4.1}$$

where $I'_{(m,n)}$ is the output feature map of the dilated convolution from the input masked image M_I, M is the binary mask and I_{gt} is the groundtruth image, the filter $\omega_{(i,j)}$ and the convolutional operator $*$. The dilation rate parameter (d_r) reverts to normal when $d_r = 1$.

It is advantageous to use dilated convolution compared to using typical convolutional layers combined with pooling. The reason for this is that a small kernel size of $k \times k$ can enlarge into $k + (k-1)(d_r - 1)$ based on the dilated stride d_r, thus allowing a flexible receptive field of fine-detail contextual information while maintaining high-quality resolution.

The inpainting solver G_θ may result in predictions $G_\theta(\hat{\mathbf{z}})$ of the missing region, that may be reasonable or ill-posed. As part of our network, D_θ adopted from [9] is used to provide improved stability and enhanced discrimination for photo-realistic images. With ongoing adversarial training, the discriminator is unable to distinguish real data from fake ones. Equation 4.2 shows the reconstruction of the image during training from G_θ:

$$I_R = I_{gt} \odot M + (1 - M) \odot G_\theta(\hat{\mathbf{z}}) \tag{4.2}$$

where I_R is the reconstructed image, I_{gt} is the ground-truth, $G_\theta(\hat{\mathbf{z}})$ is the predictions, $\odot$ is the element-wise multiplication and M is the binary mask, represented in 0 and 1. In our case 0 is the context of the entire image and 1 is the missing regions.

Equation 4.3 adopted from [9] refers to the Wasserstein discriminator.

$$\max_D V_{WGAN} = E_{x \sim p_r}[(D_\theta(I)] - E_{z \sim p_z}[D(G_\theta(I_R))] \tag{4.3}$$

where D is the discriminator and P_r is real data distribution. G is the generator of our network and P_z is the distribution.

4.3.2 Loss Function

Perceptual loss Instead of using the typical $\boldsymbol{\ell}_2$-loss function used in [152], a new combination of loss functions (luminance (L_l) and feature loss) are used. Pixel-wise reconstruction and feature space loss are not new to inpainting [223, 226, 94]. The luminance guided L_l is defined based on the $\boldsymbol{\ell}_1$-loss so as to compute the loss using a range of constant pixel values in the RGB space. This preserves colour and luminance and does not over penalise large errors [236]. To adjust our perceptual loss, the L_l is used thus minimising any error >1. Also, the L_l allows better evaluation of the predictions to match the ground-truth. More specifically, the luminance loss (L_l) is based on $\boldsymbol{\ell}_1$ and expressed as follows:

$$L_l = ||K \odot (x_i - \hat{\mathbf{z}}_i)||_1^1 \tag{4.4}$$

where i is the pixel index with x_i and $\hat{\mathbf{z}}_i$ as pixel values of the ground-truth and the predictions, constraint by a constant K. Our feature loss L_f is a feature based $\boldsymbol{\ell}_2$-loss, rather than being computed directly on the image the loss is computed in a feature space. To achieve this, the pre-trained VGG-16 model trained on ImageNet [106] is adopted and used as a feature extractor in our loss function. To be more specific the output of block3-conv3 is used for this model to generate image feature. The $\boldsymbol{\ell}_2$ the base that computes the loss function, which is the same as the perceptual loss proposed in [94]. The advantage of using feature space is that a particular filter determines the extraction of feature maps, from low-level to high-level sophisticated features. To reconstruct quality images, the loss function is computed with feature maps determined by block3-conv3, resized to the same size as masks and generated images. The reason is that using another output for example block4-conv4 or block5-conv5 will result in poor quality, as the network starts to expand the view at these layers due to more filters used. Our feature loss is expressed as follows:

$$L_f = ||\phi_i[M_I] - \phi_i[I_R]||_2^2 \tag{4.5}$$

where M_I is the input image, I_R is the reconstructed image and N is dimensions obtained from ϕ feature maps with high-level representational abstractions extracted from the third block convolution layer. By combining L_l and L_f the following equation is obtained:

$$L_p = L_l + L_f \tag{4.6}$$

By using L_p the model learns to produce finer details in the predicted features and output without any blurry artefacts. Also, the Wasserstein loss (L_w) improves convergence in GANs by computing its the mean difference between two images. Finally the entire model trains end-to-end with back-propagation and uses the global Wasserstein-perceptual loss function (L_{wp}) defined in Equation 4.7, to optimise G_θ and D_θ to learn reasonable predictions. Our goal is to reconstruct an image I_R from M_I by training the generator G_θ to learn and preserve image details.

$$L_{wp} = L_w + L_p \tag{4.7}$$

4.4 Experiment

For this model, the Keras library with TensorFlow backend is used to implement and design the network. With the choice of the dataset, the experiment settings of state of the art [124] is considered and the CelebA-HQ dataset split into 27,000 images for training and 3,000 images for testing. Normalised floating-point representation is implemented during preprocessing on the image to set the intensity values of the pixels in the range [-1,1] and applied to the mask on the image to obtain our input, as shown in Figure 4.3. Pretrained weights from VGG-16 are initialized to compute our loss function. A learning rate of 10^{-4} in G_θ and 10^{-12} in D_θ is used and optimise the training process using the Adam optimiser [103]. The Quadro P6000 GPU machine is used to train these models. According to the hardware conditions, a batch-size of 5 in each epoch for input images with shape $512 \times 512 \times 3$ is considered so as to avoid memory problems. It takes 0.193 seconds to predict missing pixels of any size created by binary mask on an image and ten days to train 100 epochs.

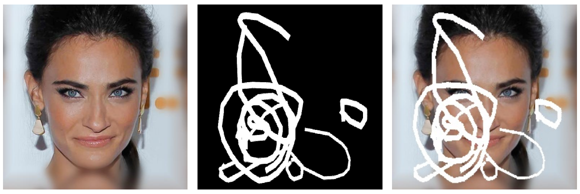


Figure 4.3: Process of input generation: a) CelebA-HQ image; b) Binary mask image [84]; and c) Corresponding masked image (input image).

4.5 Results

This section evaluates the inpainting methods qualitatively and quantitatively.

4.5.1 Qualitative Comparisons

Consider the importance of visual and semantic coherence; a qualitative comparison of our test dataset is conducted. First, is the WGAN implemention approach with L_f and L_w. It is observed that an induced pattern and pitiable colour on the images, as shown in Figure 4.4(d). Furthermore, dilated convolution, skip connections combined with end-to-end training using L_{wp} are introduced to handle the induced pattern and match the luminance of the original images. This model is compared with three popular methods:

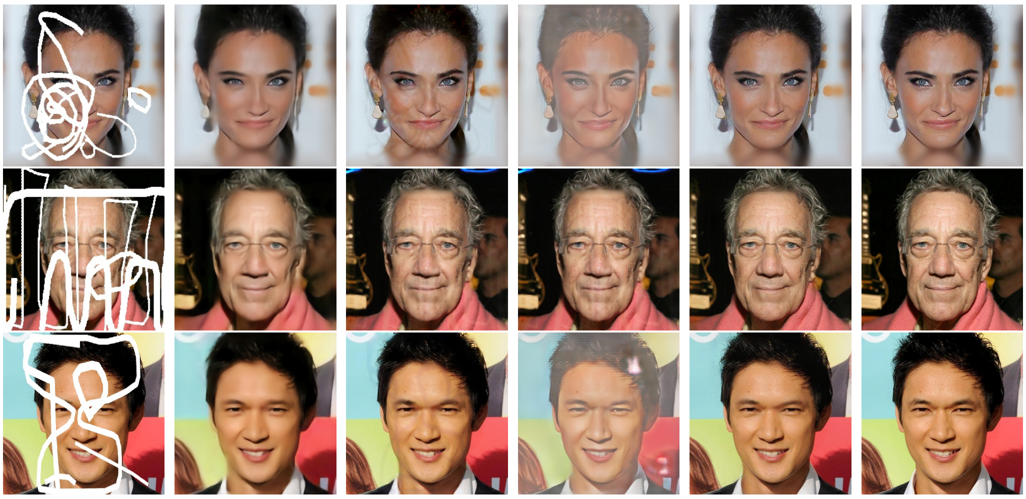

(a) Input (b) CE (c) PC (d) WGAN (e) SWGAN (f) GT

Figure 4.4: Qualitative comparison of our proposed **SWGAN** with the state-of-the-art methods on CelebA-HQ: (a) **Input masked-image**; (b) **CE** [152]; (c) **PC** [124]; (d) **WGAN**; (e) **SWGAN** (proposed method); and (f) Ground-truth image.

- **CE**: Context-Encoder method [152].
- **PC**: Image Inpainting for irregular holes using partial convolutions [124].
- **WGAN**: Wasserstein GAN method with perceptual loss.

The SWGAN model is compared to the state of the art on CelebA-HQ 512 × 512 test dataset and the results are shown in Figure 4.4. Based on visual inspection, Figure 4.4(b) illustrates blurry generated by the **CE** method [152]. On the other hand, **PC** [124] generates clear images but with residues of the binary mask left on the images as shown in Figure 5.4(c). WGAN induced pattern and low-contrast images, shown in Figure 4.4(d). Overall, our proposed SWGAN, as shown in Figure 4.4(e), produced the best visual results when compared to the ground-truth in Figure 4.4(f).

4.5.2 Quantitative Comparisons

To evaluate the performance quantitatively, some popular image quality metrics such as MAE, MSE, Peak Signal to Noise Ratio (PSNR), and SSIM are used. Table 4.1 compares our experiment results to the state of the art [152, 124] for image inpainting, with our S-WGAN in bold. For MSE and MAE, the lower the value, the better the image quality. MSE measures the average squared intensity difference of pixels while MAE measures the magnitude of error between the ground-truth image and the reconstructed image. Conversely, for PSNR and SSIM, the higher the value, the closer

Table 4.1: Quantitative comparison of various performance assessment metrics on 3,000 test images from the CelebA-HQ dataset. † Lower is better. ⊎ Higher is better.

Performance Assessment					
Method	Author	MSE †	MAE †	PSNR ⊎	SSIM ⊎
WGAN		3562.13	87.03	13.50	0.56
CE	Pathak et al. [152]	133.481	129.30	27.71	0.76
PC	Liu et al. [124]	124.62	105.94	28.82	0.90
S-WGAN	Proposed	**81.03**	**66.09**	**29.87**	**0.94**

the image quality to the ground-truth. Based on observation from Table 4.1, S-WGAN achieves lower MAE, MSE, higher PSNR and higher SSIM values in comparison with **CE** [152] and **PC** [124], which suggests that SWGAN provide more accurate predictions than the state-of-the-art inpainting algorithm.

4.6 Ablation Study

Experiments on the CelebA-HQ dataset are carried out to justify the SWGAN framework and verify the efficacy of L_p, and the results for the various adjustments of the SWGAN model are displayed. Firstly, investigations are conducted on the WGAN

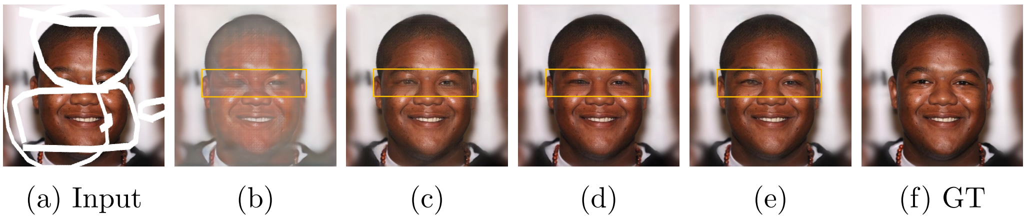
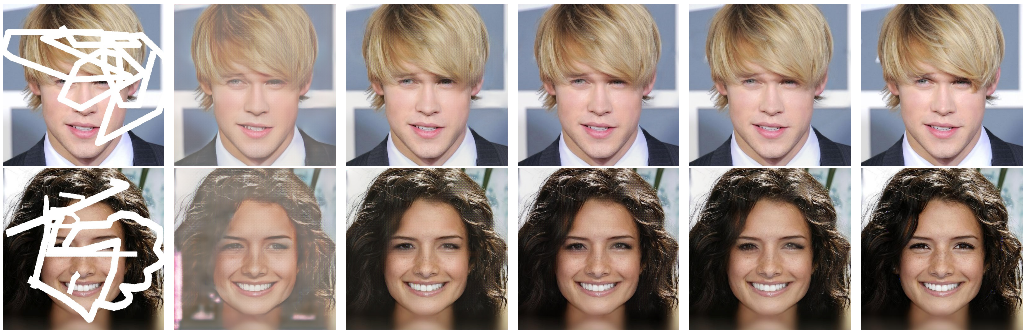

(a) Input (b) (c) (d) (e) (f) GT

Figure 4.5: Qualitative comparison of results using different architectures [94] on CelebA-HQ [98]. (a) Input masked image (b) Inpainted image by WGAN (c) Improved WGAN with skip connections (WGAN-S) (d) Improved WGAN with skip connection and dilated convolution (WGANSD) (e) Complete network with L_p (f) Ground-Truth image. The yellow box indicates the region where other models failed to inpaint successfully completely. This region in (e) shows the effectiveness of L_p on the inpainted image. (Zoom for best view)

and WGAN with skip connection (WGAN-S) using the L_f, and observed a slight improvement in texture and structure of the reconstructed masked regions of the images. Figure 4.5 (b) and (c) show changes influenced by skip connections. It is observed that visually and quantitatively, the WGAN-S performs better than WGAN model but not satisfactory as shown in the first part of Table 4.2. Secondly, an improved the WGAN-S model is implemented with dilated convolutions to each block, and additional convolution layers to obtain our WGANSD model. This model (WGANSD)



(a) Input (b) (c) (d) (e) (f) GT

Figure 4.6: Qualitative comparison of results using different architectures with the perceptual loss [94] on CelebA-HQ [98]. (a) Input masked image; (b) inpainted image by WGAN; (c) Improved WGAN with skip connection; (d) improved WGAN with skip connection and dilated convolution (e) Complete network with L_p; (f) The ground-Truth image. (Zoom to see differences between (d) and (e))

Table 4.2: Quantitative difference of results based on different architectures (WGAN), WGAN-S, WGANSD with L_f, and S-WGAN trained with L_p. † Lower is better. ⊎ Higher is better.

Performance Assessment					
Method	Author	MSE †	MAE †	PSNR ⊎	SSIM ⊎
WGAN		3562.13	87.03	13.50	0.56
WGAN-S		151.4	69.59	27.01	0.87
WGANSD		145.82	65.15	29.26	0.92
S-WGAN		**81.03**	**66.09**	**29.87**	**0.94**

is trained with the L_f and the S-WGAN model trained with our new combined loss function. It is observed that during training with the L_f the results are improved slightly, but not satisfactorily. To verify the differences of these models, a qualitative and quantitative evaluation is conducted. Visually, within the yellow rectangle on Figure 4.5 comparing columns (d) and (e), the S-WGAN result in column (e) improved with significantly enhanced local detail when compared with column (d) and the original on column (f). Also, in quantitative evaluation shown in Table 4.2, it is observed that S-WGAN trained end-to-end with L_{wp} predicts reasonable outputs with finer details. Also, more qualitatively results are shown in Figure 4.6 to demonstrate that S-WGAN generates images with contextualised features.

To validate our SWGANs' representational ability generalised to other masks other than the mask used for training this model, the various architectures of the proposed model are used to conduct experiments during the study. The Nvidia Mask [124] as the masking method used and the results are shown in Figure 4.7.

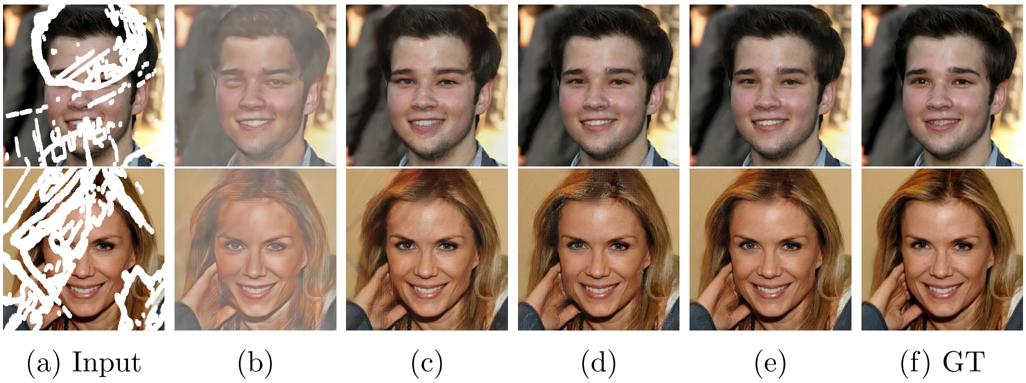


Figure 4.7: Qualitative evaluation of different architectures with perceptual loss [94] on CelebA-HQ [98] and Nvidia Mask. (a) Input masked image; (b) Inpainted image by WGAN; (c) Improved WGAN with skip connection; (d) Improved WGAN with skip connection and dilated convolution; (e) Complete network with L_p; (f) The ground-Truth image. (Zoom to see differences between (d) and (e)).

4.7 Discussion

Our proposed S-WGAN with dilated convolution and skip connections trained end-to-end with Wasserstein-perceptual loss function outperforms the state-of-the-art. Our model can learn the end-to-end mapping of input images from a large-scale dataset to predict missing pixels of the binary masks regions on the image. Our S-WGAN automatically learns and identifies missing pixels from the input and encodes them as feature representations, to be reconstructed in the decoder. Skip connections help to transfer image details forwardly and find local minimum by backward propagation.

Our experiments show the benefit of skip connection combined with Wasserstein-perceptual loss for image inpainting. The results of the proposed method are visually compared with state of the art [152, 124] in Figure 4.4. To verify the effectiveness of this network, more experiments are carried out with regular convolutions and used the L_f. It is observed that the generated images had checkerboard artefacts with pitiable visual similarity compared to the original image, as shown in Figure 4.4(d). Skip connections with dilated convolution and our new loss function were introduced, and obtained improved results that are semantically reasonable with contextualised features in all aspects.

Compared to existing methods, the generator of our S-WGAN learns specific structures in natural images by minimising L_p with an enhanced hallucinating ability powered by symmetric skip connections. Based on Figure 4.4, our SWGAN can handle irregularly shaped binary masks without any blurry artefacts and has shown edge-preserving and mask completion at border regions on the output images. Additionally, using the Wasserstein discriminator enables the overall network to perform better. This boosts the experimental performance of our network to achieve state-of-the-art results in inpainting task on high-resolution images.

One limitation is a consistent practice of other inpainting methods in the preprocessing step. Most preprocessing ignores the fact that the image has to be converted into normalised floating points representations and an inverse-normalisation on the output image, which contributes to the colour discrepancies on the output image, that leads to expensive post-processing. This has been resolved by using SWGAN with a new combination of the loss function that preserves colour and image detail.

4.8 Summary

This chapter proposed SWGAN, a new network to generate images that are semantically and visually plausible with contextualised features of face. This has been achieved by a novel network structure that can widen the receptive field in each block to capture more information and forward to the corresponding deconvolutional blocks. In addition, a newly combined loss function based on luminance and feature space is presented, together with Wasserstein loss. The network was able to generate high-resolution images from input covered with arbitrary binary masks shape and achieved a better performance when compared to the state-of-the-art methods. The proposed network has shown the effectiveness of skip connections with dilated convolutions as a capture and refining mechanism of contextual information combined with WGAN. However, in the real-world, it is unlikely that the missing region is of square holes. Therefore, the next chapter will discuss a novel algorithms specifically designed to inpaint irregular holes, such as coarse and fine wrinkles extracted from wrinkle detectors [222]. This chapter will also target irregular hole inpainting on natural scene images.

Chapter 5

RMNet: A Perceptual Adversarial Network for Facial Image Inpainting

In this chapter, a reverse mask mechanism and a reverse mask objective loss function are proposed to target only missing regions. This technique achieves contextualised features for missing regions by transferring non-corrupt image features to the end of convolution and reversing the mask during convolutions to target only missing regions. During backpropagation, the reverse mask loss assists the network by ensuring that the network layers are updated with gradients that target more of the missing parts, forcing the convolutions to make better predictions. As a result, the network constrains convolutions to predict only missing regions, reducing computing complexity while maintaining perceptual realism. The reverse mask loss is used as an approximation function to target reverse inpainting of masked regions, with back propagated errors and hints of conserved regions for better learning representation. The contents of this chapter are published in ***RMNet: A perceptual adversarial network for image inpainting*** *WACV 2021*

5.1 Introduction

This chapter considers specifically targeting the missing or damaged areas for inpainting while preserving the visible regions. The fundamental task of an inpainting algorithm is to fill in the missing regions with plausible information which are consistent with the boundary regions of the hole and the rest of the image. For most existing algorithms, it is a critical problem to solve, as transition between known and unknown regions are not naturally plausible to the naked eye. However, great progress is underway in image completion task since the proposed Partial Convolutions (**PC**) [124] and Gated Convolutions **GC** [227] set the benchmark to target irregular missing regions. These networks achieved great results, but to highlight some limitation still exist. With **PC**, each partial convolution layer has a mask which if re-normalised focuses on valid pixels and an automatic mask update for the next layer. The process in partial convolution layers takes in both the image and mask to produce features with a slightly filled mask. With more partial convolution layers, the mask region gets smaller and smaller, which can be harmful to the model performance. With **GC**, each convolution block produces two outputs (consider the first output to be A and and the second output B). Each output passes through different convolutional filters, with one through a gating block. In this process, output-A goes through an activation mechanism, while output-B goes through Sigmoid function with gating values between zeros and ones. RMNet forces the convolutions to subtract the non-corrupt areas through the reverse masking mechanism ensuring final predictions of the missing regions, with the help of the reverse mask loss. At the loss level, in the forward pass, the error between latent distributions of real and generated data are computed as squared difference as oppose to $\boldsymbol{\ell}_1$-base mask loss used in [124, 68] which calculates absolute difference. Using the $\boldsymbol{\ell}_2$-base reverse mask loss penalises more significant errors due to squared difference, making it suitable to handle masks of any size. Therefore, the error computed with the reverse mask loss forces the network via the backward pass to focus on the predictions of the missing regions yielding more plausible outcomes. Within the model are constrained layers with matrix operations namely spatial preserving operation and reverse masking operator. Unlike other networks where the mask is not concatenated at the output, our concatenated mask output allows the reverse mask loss to gain access into the network. This allows easy backpropagation access to individual layers enabling rich feature capture.

5.2 Connection to related work

Inpainting irregular or free-form mask plays an important part in image inpainting. Although it is not an easy task to achieve, it is still an ongoing research and there are room for improvement. A pioneer work to targeted hole regions for image inpainting is the **PC** proposed by [124]. A follow-up method **GC** [227] introduced gating mechanisms. The method in **GC** allows the network to have a learnable dynamic position and feature selection for each layer. The proposed method in [211] reused **PC** and introduced reversed and forward attention mechanism which allows the model to have a bidirectional attention learning on the feature maps. Instead of the hard-mask updating step, the authors modified the activation function for mask updating step. In addition to that, the authors introduced an asymmetric Gaussian shaped activation function for the attention map.

5.3 Proposed Framework

The RMNet is introduced in this section as a novel approach for solving image inpainting tasks. A Wasserstein Generative Adversarial Network (WGAN) is used as the foundation. A novel reverse masking operator that compels the model to target missing pixels is proposed using the encoding-decoding architecture of WGAN networks. This operator enables the network to restore the hidden part of an image while preserving the visible portion. In addition, a new loss function, reverse masking loss, is built around this reverse masking operator to minimise the error between the missing portions of the generated image and the ground truth.

5.3.1 Network Architecture

As mentioned previously in Section 5.3, RMNet is build using a GAN as base architecture. GANs have been previously used in image inpainting as they are able to generate missing pixels, unfortunately this often leads to the introduction of blurriness and/or artefact effects. The proposed models [124, 68, 227] attempted to solve this problem by employing partial convolution and gated convolutions. While these two approaches aim to target more efficiently missing pixels it is observed that they do not fully reduced the aberrations. Our aim through the reverse masking operator is to better target the missing region in the image while keeping visible pixels intact.

First, is to define some generic terminology that will be used through the rest of the chapter. The source image is defined as I_{gt}, the mask as M and the reversed masked $M_r = 1 - M$. The masked input image M_I is obtained as follows:

$$M_I = I_{gt} \odot M \tag{5.1}$$

where $\odot$ is the element wise multiplication operator.

Our network architecture is designed to have a generator (G_θ) and a Wasserstein Discriminator (D_θ). Our generator is designed with convolutional and deconvolutional (learnable up-sampling) layers. The convolutional layers encode features in latent

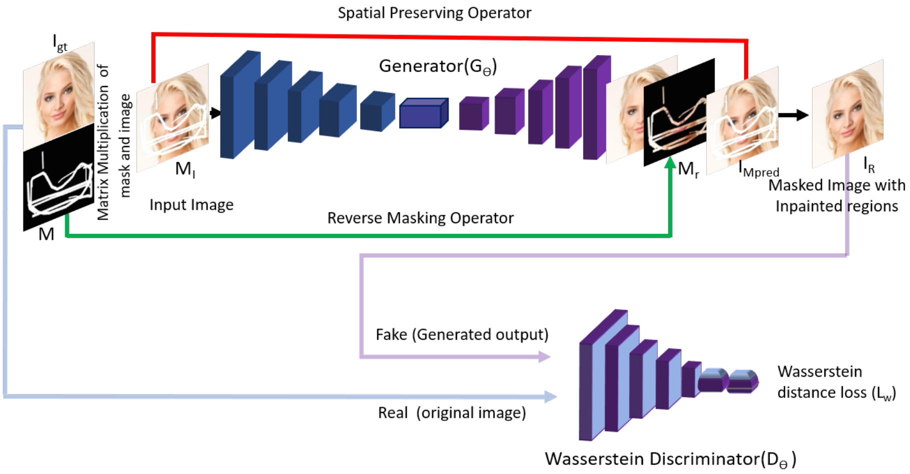

Spatial Preserving Operator
Generator(G_Θ)
I_{gt}
Matrix Multiplication of mask and image
M
M_I
Input Image
Reverse Masking Operator
M_r
I_{Mpred}
I_R
Masked Image with Inpainted regions
Fake (Generated output)
Real (original image)
Wasserstein distance loss (L_w)
Wasserstein Discriminator(D_Θ)

Figure 5.1: An overview of RMNet architecture at training showing the spatial preserving operation and reverse-masking mechanism.

space during convolution. These layers are blocks of convolution with filter size of 64 and the kernel size set to 5 × 5 with a dilation rate of 2 and Leaky-ReLU, α=0.2. Similar to Chapter 4 dilated convolutions are included to widen the receptive field to capture fine details and textural information. The convolutional feature maps obtained in each layer are the input to the next layer after rectification and pooling. Maxpooling is used to reduce variance and computational complexity by extracting important features like edges, and keep only the most present features. Include in the learnable up-sampling layers, reflection padding on a kernel size that is divisible by the stride (K-size=4, stride=2), and bilinear interpolation to resize the image, setting the up-sampling to a high-resolution, and through a Tanh function output layer. The goal of setting up the decoder in this way is to ensure that any checker-board artefacts [164] associated with the inpainted regions on the output image are cleaned and consistent with details outside the region. This technique is equivalent to sub-pixel convolution achieved in [179]. Specifically the WGAN adopted from [9] that uses the Earth-Mover distance, is used as part of our network to compare generated and real distributions of high-dimensional data. The generator will generate a reasonable reconstruction $G_\theta(\hat{z})$. Using the reversed masked operator M_r is obtained and combined with $G_\theta(\hat{z})$ to generate the predicted masked area of the image:

$$I_{Mpred} = G_\theta(\hat{z}) \odot M_r \tag{5.2}$$

The overall architecture is shown in Figure 5.1. By using this approach, our model predicts only regions with missing pixels which are consistent with surrounding pixels close to border regions of the original image. This results in high-quality reconstructed images that match the natural texture and structure of the original images that are visually plausible with contextualised features.

5.3.2 Loss Function

5.3.2.1 Generator Loss Function

The generator loss function L_G is implemented to evaluate two aspects of the predicted image: the quality of missing pixels area, and the whole image perceptual quality. Building L_G around these two metrics will ensure that the generator produces accurate missing pixels that they will blend nicely with the visible pixels. State-of-the-art methods [57, 94, 124, 126, 68] contribute to style transfer and image inpainting have used feature space instead of pixel space to optimize network. Using feature space encourages adversarial training to generate images with similar features, thus achieving more realistic results. Our new combination of loss function is computed based on feature space. This is achieved by utilizing pre-trained weights from the VGG19 model trained on ImageNet [106]. Features are extracted from block3-conv3 to compute the loss function using MSE [152] as our base. Instead of using pixel-wise representations, the extracted features are used to compute the squared difference applied to the input and output of the loss model as the perceptual loss (L_p), which is similar to [94], as in equation 5.3:

$$L_p = \frac{1}{\kappa}\sum_{i\in\phi} ||\phi_i[I_{gt}] - \phi_i[I_R]||_2^2 \tag{5.3}$$

where κ is the size of ϕ (output from block3-conv3 of VGG19), $\phi_i[I_{gt}]$ is the feature obtained by running the forward pass of VGG19 using I_{gt} as input and $\phi_i[I_R]$ is the feature obtained by running the forward pass on the output of the generator G_θ.

The reversed mask-loss (L_{rm}) is defined on the same bases as MSE, but targeting only valid features created by the mask region for reconstruction. The reversed mask loss compares the squared difference for corresponding pixels specific for regions created by the mask on the image and the reconstructed pixels of the masked-image. The reversed mask (M_r) and the original image (I_{gt}) are computed together via matrix operands to obtain L_{rm}, where

$$L_{rm} = \frac{1}{\kappa}\sum_{i\in\phi} ||\phi_i[I_{Mpred}] - \phi_i[I \odot M_r]||_2^2 \tag{5.4}$$

Finally by linearly combining L_p and L_{rm} the generator loss function is expressed as follows:

$$L_G = (1-\lambda)L_p + \lambda L_{rm}, \tag{5.5}$$

where $\lambda \in [0\ 1]$, to allow an optimal evaluation of features by minimising the error on the missing region to match predictions comparable to the ground-truth.

5.3.2.2 Discriminator Loss Function

Since the network is trained with Wasserstein distance loss function (L_w), it is expressed in this equation 5.6 as:

$$L_w = E_{I_{gt} \sim P_x}[D_\theta(I_{gt})] - E_{I_R \sim P_z}[D_\theta(I_R)] \tag{5.6}$$

Here the first term is the probability of real data distribution and the second term is the generated data distribution.

5.3.3 Reverse Mask

The advantages of this approach are discussed here using reverse mask operator compared to **PC** and **GC**, two approaches previously used for image inpainting. All three methods are summarized in Figure 5.2. The process in partial convolution layers takes in both the image and mask to produce features with a slightly filled mask. Each partial convolutional layer has a mask which if re-normalised focuses on valid pixels and an automatic mask update for the next layer. With more partial convolution layers, the mask region gets smaller and smaller, which can disappear in deeper layers and revert all mask values to ones. With gated convolutions, the convolutions automatically learn soft mask from data, extracting features according to mask regions. Each convolution block learns dynamic features for each channel at each spatial location and pass it through different filters. In this process, the output features goes through an activation mechanism (ReLU) while gating values (between zeros and ones) goes through Sigmoid function. The gating values show learnable features with semantic segmentation and highlighted mask regions as a sketch in separate channels to generate inpainting results. This network requires a substantial amount of CPU/GPU memory to run the gating scheme. Our proposed reverse mask forces the convolutions to subtract the non-corrupt areas through the reverse masking mechanism ensuring final predictions of the missing regions, with the help of the reverse mask loss, forcing the network via the backward pass to focus on the predictions of the missing regions yielding more plausible outcomes.

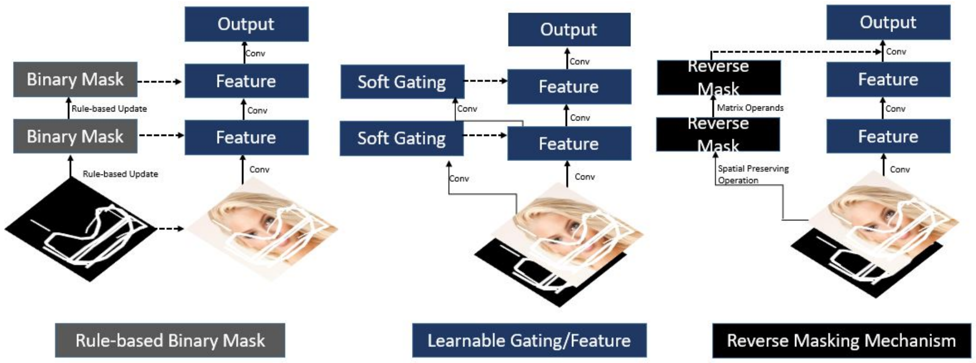


Figure 5.2: Illustration of partial convolution (left) and gated convolution (middle) and Reverse-masking (right). This illustration shows the differences between reverse masking compared to PC [124] and GC [227]

5.4 Experiment

Similar to previous chapter 4, the Keras library with TensorFlow backend is used to implement the proposed model of this chapter. The of datasets matches the state of the art [124, 126, 152, 126] with similar experimental settings. All images and masks are resized using OpenCV library interpolation function `INTER_AREA` to $256 \times 256 \times 3$ and $256 \times 256 \times 1$ respectively. For optimization, the Adam optimizer [103] with learning rate of 10^{-4}, $\beta = 0.9$ for G_θ and 10^{-12}, $\beta = 0.9$ for D_θ is used. The model is trained with a batch size of 5 on NVIDIA Quadro P6000 GPU machine, on Places2 and Paris Street View. The NVIDIA GeForce GTX 1080 Ti Dual GPU machine is used for training the CelebA-HQ dataset high-resolution images. It takes 0.193 seconds to predict missing pixels of any size created by binary mask on an image, and 7 days to train 100 epochs of 27,000 high-resolution images.

5.4.1 Training Datasets

- **CelebA-HQ** [99]: A dataset curated from CelebA [131] containing 30000 high quality images of 1024×1024, 512×512 and 128×128 resolutions based on previous Chapter 4.

- **Paris Street View** [45]: This dataset contains 14,900 training images and a test set of 100 images collected from Paris street views. The main focus of the dataset is the buildings of the city and very important in geo-location task.

- **Places2** [241]: A dataset containing over 1 million images from 365 scenery from places. It is suitable for model learning and understanding of diverse

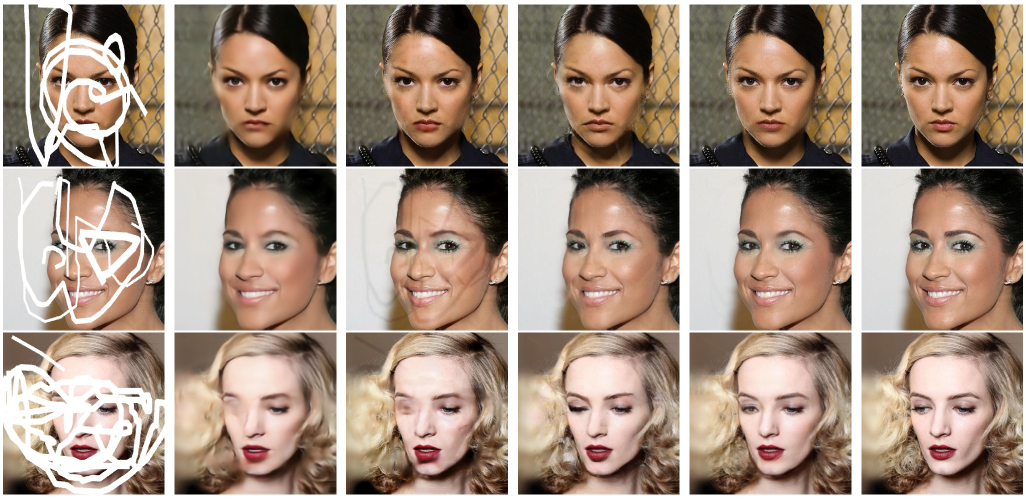

(a) Input (b) **CE** (c) **PConv** (d) **GC** (e) **RMNet** (f) GT

Figure 5.3: Visual comparison of the inpainted results by **CE**, **PConv**, **GC** and **RMNet** on CelebA-HQ [124] where Quick Draw dataset [84] is used as masking method using mask hole-to-image ratios [0.01,0.6].

complex natural scenery. The following scene categories were chosen: butte, canyon, field, synagogue, tundra, and valley (in that order) as per [216]. In each category, there are 5,000 training images, 900 test images, and 100 validation images. Our model is trained on the training set and evaluated on the validation set.

5.5 Results

To evaluate the performance of the proposed method, the RMNet is compared with three other methods on the same settings for image size, irregular holes and datasets. These experiments include

- **CE**: Context-Encoder [152]
- **PC**: Partial Convolutions [124]
- **GC**: Free-Form image inpainting with Gated Convolutions [227]
- **RMNet-0.1**: RMNet using L_{rm} when λ= **0.1**, our proposed method.
- **RMNet-0.4**: RMNet using L_{rm} when λ= **0.4**, our proposed method.

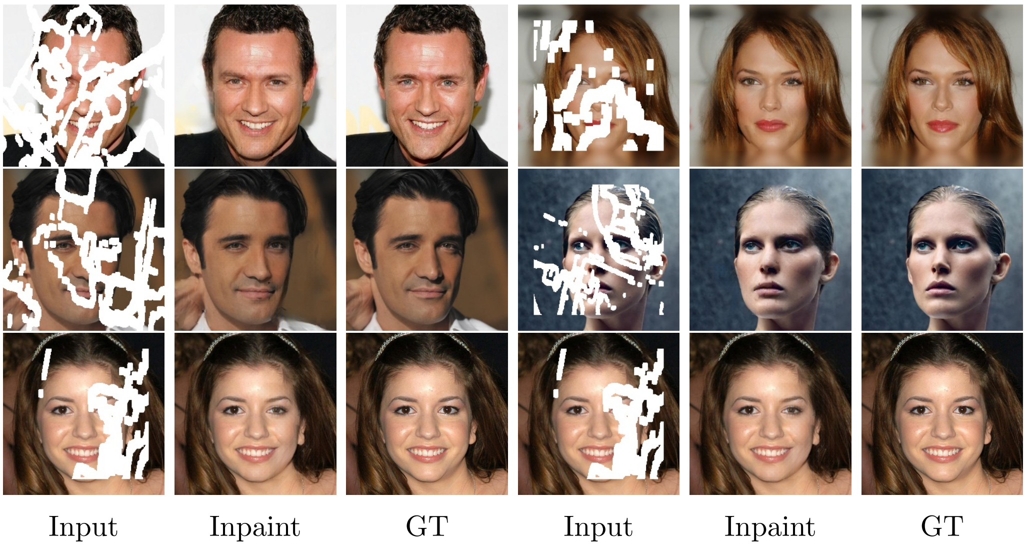


Figure 5.4: Results of image inpainting using RMNet-0.4 on CelebA-HQ Dataset [124] with Nvidia Mask dataset [124] used as masks, where images in column (a) are the masked-image generated using the Nvidia Mask dataset [124]; images in column (b) are the results of inpainting by our proposed method; and images in column (c) are the ground-truth.

5.5.1 Qualitative Comparison

Experiments are conducted based on similar implementations by [124, 152] and pre-trained model for the state of the art [227] and compared our results.

For Places2 dataset, 10,000 images are randomly selected for the training samples to match [152] and the same number of test samples are used to evaluate the model. The results are shown on Figure 5.6. For CelebA-HQ, the downloaded pre-trained models for the state of the art [227] are used on the testing set and compared the results obtained using the proposed model on the same test set and masking method. Based on visual comparison, the proposed model shows realistic and coherent output images. Observing from Figure 5.6, other models fail to yield images with structural and textural content as the images are either blurry or fail due to the image-to-hole ratio increase with arbitrary mask.

5.5.2 Quantitative Comparison

To statistically understand the inpainting performance, the quantitative performance of the proposed model is compared with the state of the art methods using four classic metrics: Frechet Inception Distance (FID) by Heusel et al. [75], Mean Absolute Error (MAE), Peak Signal to Noise Ratio (PSNR), and SSIM [207]. The FID measures the

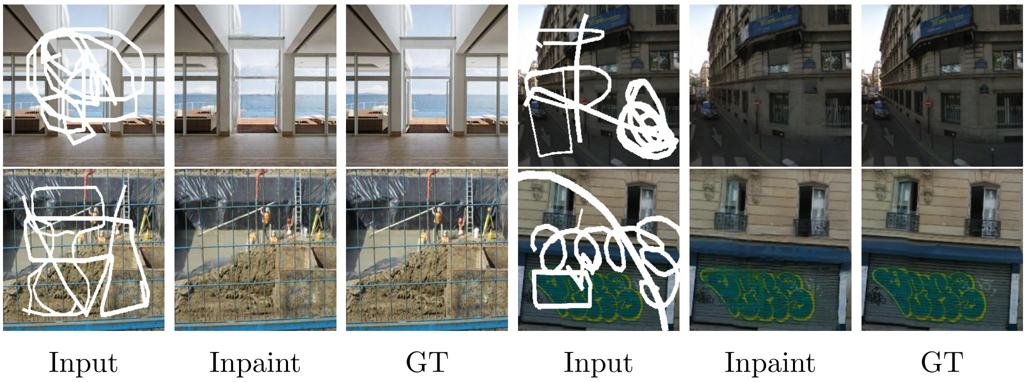


Figure 5.5: Results of image inpainting using RMNet-0.4 on Places2 [241] and Paris Street View [45], where images in column (a) are the masked-image generated using the Quick-Draw dataset [84]; images in column (b) are the results of inpainting by our proposed method; and images in column (c) are the ground-truth.

Table 5.1: Results from CelebA-HQ test dataset, where Quick Draw dataset by Iskakov et al. [84] is used as masking method with mask hole-to-image ratios range between [0.01,0.6]. † Lower is better. ⊎ Higher is better.

Performance Assessment					
Method	Author	FID †	MAE †	PSNR ⊎	SSIM ⊎
RMNet-0.1	Ours	26.95	33.40	38.46	0.88
CE	Pathak et al. [152]	29.96	123.54	32.61	0.69
PConv	Liu et al. [124]	15.86	98.01	33.03	0.81
GC	Yu et al. [227]	4.29	43.10	39.69	0.92
RMNet-0.4	Proposed	**3.09**	**31.91**	**40.40**	**0.94**

quality of reconstructed images by calculating the distance between feature vectors of ground-truth image and reconstructed images. The other metrics (MAE, PSNR, SSIM) evaluate at pixel and perceptual levels respectively. The results in Table 5.3 are evaluated based on masks with various test hole-to-image area ratios ranging from [0.01,0.6], on a test set of 3000 images from the CelebA-HQ.

Table 5.2: The inpainting results of RMNet-0.4 on Paris Street View and Places2, where Quick Draw dataset by Iskakov et al.[84] is used as masking method with mask hole-to-image ratios range between [0.01,0.6]. † Lower is better. ⊎ Higher is better.

Performance Assessment					
Dataset	Method	FID†	MAE †	PSNR ⊎	SSIM ⊎
Paris Street View	RMNet-0.4	17.64	33.81	39.55	0.91
Places2	RMNet-0.4	4.47	27.77	39.66	0.93

A lower FID score indicates that the reconstructed images are close to the ground-truth. A similar judgement quantifies the MAE, though it measures the magnitude in pixel error between the ground-truth and reconstructed images. For PSNR and SSIM, higher values indicate good quality images closer to the ground-truth image. Looking at the results in Table 5.1, our model achieves better performances than other state-of-the-art methods.

To generalise our model, further inference were conducted using the Nvidia Mask dataset [124] on the CelebA-HQ dataset. The results are shown in Figure 5.4. Note that these masks were not used during training of RMNet. It for testing only to demonstrate our model superiority and robustness across mask. Further experiments are conducted on Paris Street View and Places2 to generalize the model to inpaint natural scene imaegs. Masks of the same sizes [0.01,0.6] are randomly selected during testing. Results can be seen on Table 5.2 and Figure 5.5, which shows our model is able to generalize to various inpainting tasks and not just face inpainting.

5.5.3 Ablation Study

The effectiveness of reversed mask loss is investigated, and experiments conducted at different λ and compare its performance using hole-to-image area ratios between [0.01, 0.6]. The λ= **0, 0.1, 0.3, 0.4, 0.5** are used on reversed mask loss for different experiments with the same settings. The results are shown in Table 5.3. When λ=**0** the model has no access to the reversed mask to compute the loss function, it is observed that the mask residue is left on the image as shown on Figure 5.6. Based on the output images, although the spatial information of the image is preserved, the convolutional inpainted regions need assistance to minimise the loss between the mask and the reverse mask loss during prediction. For a start, a small value of λ=**0.1** is used. It is observed that the results get better but with obtain poor performance visually. That is, the pixels in the mask region did not blend properly with the surrounding pixels, leaving the image with inconsistencies in texture. Further experiments with λ= **0.3, 0.4, 0.5** and carried out and subjective evaluations using FID, MAE, PSNR and SSIM are recorded. With λ= **0.4**, the best balanced is stroke and generated images demonstrated the best results with no further improvement by increasing the value of λ. The mask as input to the CNN allows the network to learn the size of the corrupted region. The bigger the mask, the longer it takes to achieve perceptual similarity. This is because the region grows bigger and the network takes longer due to a smaller proportion of the loss covering the entire image to ensure the inpainted region is semantically consistent with the rest of the image.

Table 5.3: Results from Paris Street View and Places2 using Quick Draw dataset by Iskakov et al. [84] as masking method with mask hole-to-image ratios [0.01,0.6]. † Lower is better. ⊎ Higher is better.

Performance Assessment					
$\mathcal{L}_{rm}$	weight	FID†	MAE †	PSNR ⊎	SSIM ⊎
λ=0.1,	RMNet-0.1	26.95	33.40	38.46	0.88
λ=0.3,	RMNet-0.3	4.14	31.57	40.20	0.93
λ=0.4,	**RMNet-0.4**	**3.09**	**31.91**	**40.40**	**0.94**
λ=0.5,	RMNet-0.5	4.14	31.0	40.0	0.93

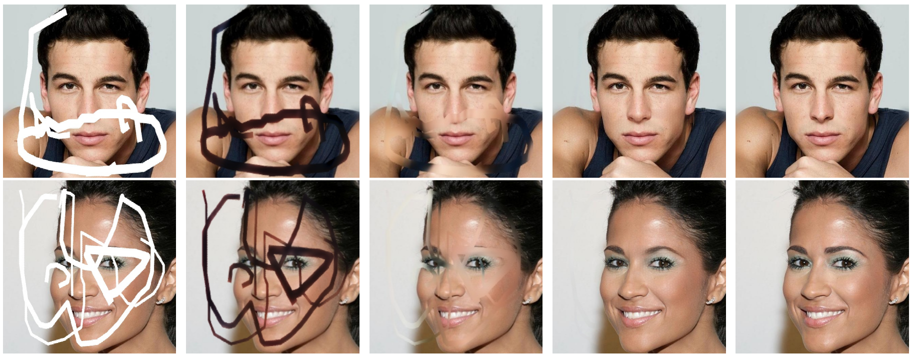

(a) Masked (b) $\lambda = 0$ (c) $\lambda = 0.1$ (d) $\lambda = 0.4$ (e) GT

Figure 5.6: Visual results on ablation study where (a) is the input masked image (b) results of the RMNet-base model without $\boldsymbol{\ell}_{rm}$. As loss on this model, $\boldsymbol{\ell}_2$ and binary-cross-entropy are used. (c) RMNet with $\boldsymbol{\ell}_{rm}$ loss at with weight application of 0.1 (d) RMNet with full $\boldsymbol{\ell}_{VP}(\phi)$ with λ=0.4 on Quick-Draw [84] as masking method.

5.6 Discussion

The ability to generalize good performance on machine learning algorithms based on end-to-end mapping of real data distribution to unseen data, is vital to the learning outcome by various models. In image inpainting, algorithms based on generative networks, predict missing regions following a real data distribution from a large dataset. Typical approaches predict these hidden regions by applying an encoding-decoding process to the image, where the missing regions are defined usually by a binary mask. The encoding process will produce a high dimensional feature representation of the image where the missing information has been recovered, while the decoding process will generate the original image, i.e. the input image without missing information.

Generally the learning procedure of the model parameters is performed by solving a loss function minimization. Often, the model parameters are learned using a forward-backward process. In the forward pass the loss function calculates an error between latent distribution of real and generated data. The loss is then back-propagated into the model to update the parameters weight (backward pass).

The ability of our model to identify the missing regions of the input image is essentially assisted by our reverse mask loss, that will force the network to primarily focus on the prediction of the missing regions. The reverse mask loss combined with the perceptual loss and the preserved spatial information within the network, will ensure accurate prediction of missing regions while keeping the general structure of the image and high resolution details.

A practice of existing inpainting GAN is the requirement to apply the mask on an image to obtain a composite image (masked-image), which in turn share the same pixel level. Additionally, when using irregular masks, these regions to be inpainted can be seen by the algorithm as a square bounding box that contains both visible and missing pixels, which can cause the GAN to generate images that can contain structural and textural artefacts.

To overcome this issue, the usual approach is modified to have two inputs to the model, during the training, the image and its associated binary mask. This allows us to have access to the mask, at the end of our encoding/decoding network, through the spatial preserving operation and gives us the ability to compute the following, that will be used during the loss computation:

- A reversed mask applied to the output image.
- Use a spatial preserving operation to get the original masked-image.

- Use matrix operands to add the reversed mask image back on the masked-image.

By using this method of internal masking and restoring, our network can inpaint only the required features while maintaining the original image structure and texture with high level of details. Our network shows better achievement, when compared to state-of-the-art methods, numerically and visually, where the output image are visually closer to the original image than other approaches.

Overall, this technique demonstrates that by combining global (perceptual) and particular (reverse mask) loss, superior results may be obtained, overcoming the restriction of having a model trained simply using global loss.

5.7 Summary

This chapter proposed a novel approach using reverse masking combined with Wasserstein GAN to perform image inpainting task on various binary mask shapes and sizes. Our model targets missing pixels and reconstructs an image with structural and textural consistency. This model also demonstrates plausible generated contents of missing regions of the generated image on high resolution images while preserving image details. Through the experimental results, it has been demonstrated that the model trained alongside reverse matrix operands of the mask is beneficial to image inpainting. In addition, the proposed model when compared with the state-of-the-art methods can obtain competitive results. Nonetheless, this model still faces challenges in generating subtle textural features of the facial image or generating images of high quality with fidelity preservation of features such as make-up and facial expression (see Appendix A. The next chapter describes a model with semantic capability reasoning of the later features.

Chapter 6

Foreground-Guided Facial Inpainting with Fidelity Preservation

In this chapter, a semantic segmentation mask is used within the network to enforce learning and ensure that the generated regions have meaningful contextual predictions. The network can efficiently infer important face features by using a semantic segmentation mask. To provide occlusion reasoning to the model, the objective learning function is computed with the segmentation mask of the skin region of the face, which updates weights in the generator with numerical hints in the form of gradients to make more plausible predictions. It is proved that the model can preserve a facial expression as well as additional important facial attributes by employing the segmentation mask as a loss. In addition, further studies are being undertaken on the facial regions to establish the model's superiority over current methodologies in this area.

6.1 Introduction

Facial image inpainting, with high-fidelity preservation, is a very challenging task. As already mention in Chapter 1, the highly developed visual system of humans will easily detect an incorrectly predicted region of the face. Thus a face with a predicted region that is not coherent with the rest of the image will be classed as invalid or determined as fake. However, this does not mean that previous inpainting algorithms have not performed well. The outstanding performance of the works in Chapter 4, 5 and the state of the art [152, 137, 227] are acknowledged; but there is more room for improvement to design models that can capture and preserve the subtle textures of key facial features without totally changing the identity of the subject. Furthermore, existing methods mentioned in Chapter 2 try to represent subjective quality by applying the SSIM metric, which evaluates the degree of structural similarity between the generated image and the original image.Another reliable approach used in literature is applying the PSNR metric which quantifies the reconstruction quality of the generated image to the ground truth image. Unfortunately, inpainting results are best determined by human perception, indicating that there is a strong evidence that visual quality of the blended inpainted regions with their surrounding is more important qualitative than the quantitative performance. Based on these facts, the design the proposed framework in this chapter that is capable of extracting and transferring facial features using convolutional neural network (CNN) layers. Specifically, it is design with a new loss function with semantic capability reasoning of facial expressions, natural and unnatural features (make-up). Experiments using the CelebA-HQ dataset demonstrate high-fidelity preservation of facial components compared to the state-of-the-art methods.

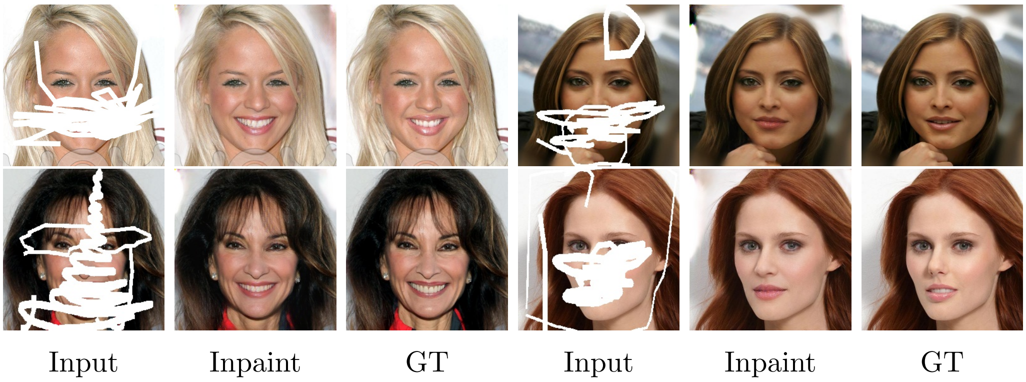


Figure 6.1: Inpainted images from the proposed model presented in this chapter showing semantic understanding with contextualised features.

6.2 Connection to Related Work

More recent approaches [238, 243, 125] are the use deep learning based methods to handle images with high complexity in terms of structure and texture. For example [238] extracts style from different images to guide the model to learn how to generate more plausible contextual information of missing regions. The approach [243] is designed to learn mapping of features extracted from an input masked image in latent space to the ground-truth in an end-to-end fashion. In this approach [125] the CNN model hallucinates high-quality textural and structural information that can fill in the missing contents by training a large scale dataset in a data-driven manner. The CNN model is known in literature to predict and understand an image structure without an explicit modelling of structures during the learning process [214]. Though it has shown high understanding of the image structure there are still difficulties in solving problems with masks of arbitrary sizes and producing a contextualised features of the inpainted regions. An example is seen in Figure 6.3 where the state-of-the-art [152, 90, 124] presented failures with arbitrary masks regions where the inpainted image is either overly smooth or left with mask residues. However, these failures can be caused by substantial image-to-mask hole variations or the complexity of the background. Another good example of deep learning based methods is proposed in [227] and popularly known as Gated convolutions (GC). This method generates images with less artefacts compared to the aforementioned methods. The reason other methods perform better than others possibly, due to the poor stability of convolutional features during learning and inference. Hence the aforementioned GC [227] may have have a better performance due to its ability to learn soft mask during convolutions as opposed to Partial Convolutions (PC) [124] which updates hard mask. The problem is because convolutional kernels are usually inconsistent when capturing information from irregular masked images due to the unpredictable nature of the inputs. However, there is still ongoing research on learning based methods [238, 229, 221, 11] with remarkable improvements proposed yearly as it is with our newly proposed solution to facial inpainting, which is presented in this chapter. Based on the study in Chapter 2, several limitations (which have already been mentioned in the named Chapter) are associated with training these methods. One of the reasons is that there are no accurate baseline measurement of techniques in this category. Thus, there is no standard parameters (learning rate, batch size, GPU) and design implementation for comparison. Most measurements have been based on quantitative values for SSIM and PSNR which do not really tie with the qualitative evaluation (visual comparison). However, quantitative valuations are still very important tools to measure these algorithms but the question is “Does the audience appreciate the visual quality or the quantitative valuations?” or “Are these methods using the same baseline parameters?” and “ Are the preservation of key facial features with fidelity preservation met?”. Another limitation is the improper utilisation of foreground regions within attention layers/units to better infer missing content on the image. A further limitation is that the hole-region on background is within the same pixel level with the rest of the image which makes it harder for learning based methods differentiate a small overlap. Moreover, it is not easy for convolutional layers to propagate features from one area of the feature map to another. This is because during convolutions, the convolutional layers find it difficult to connect all locations within a feature map [152]. This problem is ongoing and numerous solutions have been attempted. For example [152] tried to solve the problem by introducing fully-connected layers to directly connect all activation. Other models [124, 226, 227, 228, 238, 243, 89, 214, 129] have used different techniques with [124] introducing partial convolutions with hard mask updating and [227], gated convolutions with soft mask learning [227]. For this reasons, this chapter describes the proposed foreground guided image inpainting network specifically designed to capture and preserve key facial components. Our objective is to design and implement a network that has the capability to preserve the relevant features of the face with respect to various expressions and non-natural attributes as seen in Figure 6.1. To instantiate the design, the assumption is that the foreground pixels reflect the background ones, which are readily available for disentanglement in latent space and are masked within the input image by the binary mask regions. The authors in [214] used a foreground completion module to pass detected foreground contours of the masked (corrupted) image to perform an initial completion map of the foreground contours. The network uses the predicted contours as guide to perform the inpainting task. The key point to consider from our assumption is that the foreground segmentation mask serves a representation of the disentangled pixels of the background regions in latent space. Thus using the mask manifold [130] will enable the CNN layers to propagate features with respect to facial attributes (natural and non-natural), pose and shapes. Refer to Figure 6.1 showing inpainted images from our model with contextualised features.

Semantic scene understanding is an integral part of image inpainting because the hallucination of pixels to recover the damaged regions requires a semantic understanding of the global structure to the region to be inpainted. The semantic segmentation map of a face can well represent the foreground of the image, where the binary mask is applied to create the damaged region. During hand inpainting, the painter takes into consideration the background pixels and tries to semantically draw a silhouette structure outlining the boundaries before colouring and linking the colour end-nodes or strokes to complete the damaged pixels, thus ensuring consistency with the entire image. Naturally, it is intuitive to consider that an occluded face will normally have two eye spots, a nose and a mouth. Based on this assumption, one can conclude that occlusion reasoning can improve the ability of CNNs to better estimate or hallucinate missing pixels regions created by the binary mask.

The authors [172] proposed to use a modified version of PSPNet [237] to conduct semantic foreground inpainting task. The network is designed to take two inputs (masked image and mask) based on a two pipeline encoder-decoder network where the reconstructed images are semantics and depth for visible and occluded pixels. To ensure supervision, the authors designed the model to generated fake random foreground masks, which are applied on the input together with the real foreground masks during training. This method is only limited to inpainting with semantic mask and depth maps. Also, the two stage network makes it sub-optimal compared to a single pipeline in terms of efficiency. [136] introduced a max-pooling module and used semantic scene without foreground objects to conduct an inpainting task. The max-pooling module which designed to fit within the ResNet [73] encoder blocks takes an intermediate feature map, a foreground segmentation mask and binary mask as input to output an inpainted feature map. The module pools the foreground segmentation and binary mask simultaneously and compares their features with the previous mask to index updated pixels within the current iteration. The new patch is forwarded and merged with the input feature which is passed through more convolution blocks before feeding into the decoder. Note that extra max-pooling is operation is performed on the foreground mask before feeding the max-pooling module. One limitation of these modules is that sharing features between models can improve efficiency but degrade performance. [115] created the CelebAMask-HQ segmentation mask dataset as key intermediate representation of facial attributes and proposed a model that can flexibly manipulate these attributes with fidelity preservation. However, because GANs use a discriminator as the examiner, a direct supervision of an occluded region will not be possible if the ground-truth regions behind the binary mask are not available. Based on this assumption, this model is designed with a discriminator that will judge the occluded regions to ensure that the inpainted region is realistic and semantically consistent with contextualised features. Based on these findings, the model is designed to use foreground segmentation masks as loss model. This model differs from the other models [172, 214, 136] in that the binary masks are applied on the foreground masks, and passed through convolutions. The CelebAMask-HQ segmentation mask dataset [115] is used as foreground mask dataset. The following describes our proposed facial inpainting framework that uses foreground mask and a new loss function that uses the foreground mask to ensure image fidelity.

6.3 Architecture

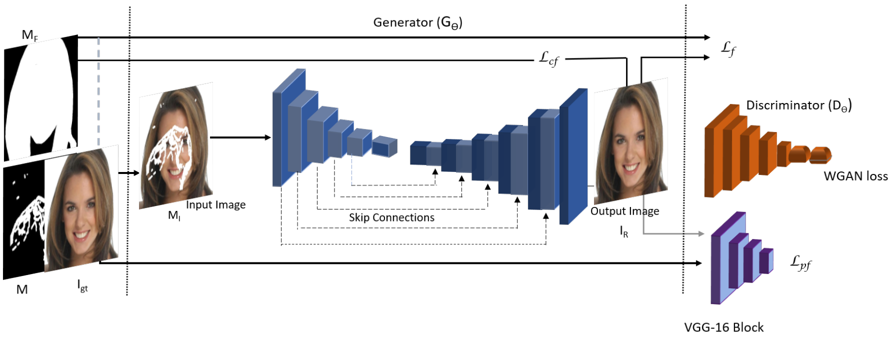


Figure 6.2: Our proposed foreground-guided image inpainting framework with symmetric chain of convolutional and deconvolutional features. The foreground segmentation mask and masked image are the inputs to the network and parameters of the loss functions.

The design of our proposed network has an encoder-decoder, as the generator (G_θ) and a discriminator (D_θ), to achieve realistic results. The encoder architecture is based on [88] with the exception of the foreground segmentation mask (henceforth, foreground mask) and masked image as input. During training, the foreground mask is kept in tacked and does not pass through convolutions. The foreground mask is used within the generator as an access mechanism for the loss to better average pixels during backward pass. The masked image is downsampled to allow the network to learn latent representations of facial feature maps with weak supervision from the foreground mask. A symmetric chain of convolutional and deconvolutional features are instroduced, where key components of feature maps without corruption extracted during convolutions are preserved, and added within the decoder. The encoder has 10 convolutional and 5 deconvolutional blocks each containing filters $[64, 128, 256, 512, 1024]$. Within the encoder, maxpooling operations are used to extract high level feature maps. For each block before the final layer, LeakyReLU activation and batch normalization are implemented at the encoder and decoder, where ReLU is used only within upsampling blocks to speed up the reconstruction process and LeakyReLU is more balanced and enables the network learn faster.

During this study, it was clear that for a much deeper understanding of the scene, it is important to design the network to synthesis high-level features over large spatial extents. Hence the reason why the features are symmetrically linked between the convolution and deconvolutional to enable the decoder to fill in large areas of scene with hints from surrounding. With this type of approach, the decoder will understand the context of the image and produce plausible predictions of the missing parts with contextualised features. However, this cannot be achieved without a loss function. The loss function is proposed to reduce the burden on the network and help to train better. The generator loss is introduced to minimises the error between the predictions and ground-truth taking into consideration the foreground mask as guide.

6.4 Loss Function

To train this model, a new loss function that takes into consideration of the foreground pixels to minimise the error between the generated and ground-truth images is implemented. The new loss function that uses face segmentation, is based on the L_2 because one of its properties allows an understanding of the overall structure of the missing regions in relation to context. However, despite the great ability to allow capture of structure in terms of context, a pixel wise version will lead to blurry artefacts. This is avoided by computing the loss with foreground mask combined with a context-foreground loss (L_{cf}) that uses the L_1 base to preserve its colour and luminance with a substantial evaluations of the predictions matching the ground-truth. For a better understanding, the loss functions used within the generator is expressed as follows:

$$L_{cf} = \frac{1}{N_{I_{gt}}} ||M_F \odot (M_I - I_R)||_1^1 \tag{6.1}$$

$$L_f = \frac{1}{N_{I_{gt}}} ||M_F \odot (I_{gt} - I_R)||_2^2 \tag{6.2}$$

where M_I is the input, I_R is the predicted output and $N_{I_{gt}}$ is the number of elements in I_{gt} in the shape height (H), width (W) and channel (C), i.e. $N_{I_{gt}} = H * W * C$ and $\odot$ is the element-wise multiplication of the foreground mask M_F with I_R and I_{gt}. These losses ensure preservation of luminance when computing the absolute difference between ground-truth image (I_{gt}) and the predicted image (I_R).

$$L_{p_f} = \frac{1}{N_{I_{gt}}} ||M_F \odot [\phi_i(M_I) - \phi_i I_R)]||_2^2 \tag{6.3}$$

where ϕ_i is the feature map of the $i^i th$ layer of pre-trained VGG16 model. The $\mathcal{L}_{p_F}$ uses the M_F and intermediate features from a fixed VGG16 model to compute the L_2 distance between ground-truth and predicted images.

$$L_p = \frac{1}{N_{I_{gt}}} ||(\phi_i[M_I] - \phi_i[I_R])||_2^2 \tag{6.4}$$

where ϕ is the VGG16 model function to extract high-level features to compute the loss function. The perceptual loss compares the distance between the generated and ground-truth images using pre-trained activation maps of the VGG-16 model. It is a feature level penalty that penalizes the results to ensure the results are perceptually similar to the ground-truth. The generator loss is defined as:

$$L_{total} = (1-\lambda)(l_f + L_{fp} + L_{cf}) + \lambda(L_p) \tag{6.5}$$

where λ are coefficients, the total loss minimizes the error between predictions and the ground-truth image. The while the discriminator loss measures how realistic the predictions from the generator are compared to the ground-truth.

For the discriminator, the Wasserstein GAN (WGAN) approach is adopted to measure the distance between predictions and the ground-truth.

$$\max_{D} V_{WGAN} = E_{x\sim p_r}[(D_\theta(I_{gt})] - E_{z\sim p_z}[D(G_\theta(I_R))] \tag{6.6}$$

Equation 6.6 refers to the WGAN loss based on distributions of I_{gt} (real) data and I_{pred} (generated) data.

6.5 Training and Experiments

6.5.1 Training Datasets

- **CelebA-HQ** [99] from Chapters 4 and 5.
- **CelebA-HQ-Mask** shown in Chapter 3.

6.5.2 Method Comparison

Our proposed method compares quantitatively and qualitatively with the state-of-the-art methods.

- **Context encoder-decoder framework (CE)** [152] introduced the channel-wise fully connected layer to solve the convolutional layer limitation associated with failures in direct connection of all locations within a specific feature map. The channel-wise fully connected layer is designed to directly link all activation; thus enabling propagation of information within the activation of a feature map.
- **Partial Convolution (PC)** [124] introduced convolutions with mask updating to alleviate the transfer of feature for irregular masks regions within convolutions.

- **Gated Convolution (GC)** [227] introduced gating mechanism that learns soft mask within convolutions to ease the transfer of features within convolutions. It is different from PC in that the irregular mask is learned whereas in former, hard mask is updated in each step.
- **Proposed** introduces semantic reasoning of features using a foreground mask within the network as a loss model. The key aspect here is that the foreground mask represents disentangled pixels of attribute features of the face. Thus semantic reasoning assists the convolutional layers to hallucinate pixels with fidelity preservation.

6.5.3 Implementation

This model is trained with the generator and discriminator loss defined in section 6.4 to solve the inpainting task posed by the missing regions created by the binary mask on the input image. The architecture is similar to the proposed in 4 but with different losses and applied loss weights (coefficients) to the generator loss. Our intention was to ensure that during training, the generator is punished more by increasing its loss weight of the foreground loss to learn structural and textural features to have an overall understanding of the semantic nature of the face region. Our proposed foreground loss further emphasizes the consistency of the predictions by feeding the generator via backward pass with a penalty on the background pixels using the foreground pixels. The Keras library with Tensorflow-backend is used for the implementation and training of the model end-to-end. The choice of dataset follows the experimental settings of the previous chapter and by the state-of-the-art method [124] to split our data into 27K train and 3K test images. The images are normalized as per 4; setting the pixel intensities of all input images in the range [-1,1]. Our model is trained for 100 epochs with a learning rate of 10^{-4} in $G(z)$ and 10^{-12} in $D(x)$ using the Adam optimizer [103]. Our hardware condition limited us to a batch-size of 5 because of the deep nature of the network. The NVIDIA P6000 GPU is used to conduct the full experiment from training to inference.

6.6 Results and Discussion

Previous research [152, 124, 91] used quantitative results to rank the performance of facial image inpainting. Here the performances are evaluated on predicted images, predicted face and hair regions only and the qualitative results.

Table 6.1: Quantitative comparison of various performance assessment metrics on 3,000 test images from the CelebA-HQ dataset. † Lower is better. ⊎ Higher is better.

Performance Assessment						
Method	Author	MSE †	MAE †	FID†	PSNR ⊎	SSIM ⊎
CE	Pathak et al. [152]	133.481	129.30	29.96	27.71	0.76
PC	Liu et al. [124]	124.62	105.94	15.86	28.82	0.90
GC	Yu et al. [227]	**102.42**	**43.10**	**4.29**	**39.96**	**0.92**
Proposed	Foreground-guided	194.86	57.38	9.63	34.35	**0.92**

6.6.1 Quantitative Results

In image inpainting in the wild, it is important to note that the visual and semantic understanding of the completed regions is of high importance to the audience. This is due to the fact that in real-life scenarios, users want to appreciate the visual quality of the blending between the inpainted regions and the original unmasked regions. However, in computer vision, the quantitative evaluation for these regions is to improve and eliminate weak baseline models. Based on previous state-of-the-art research, the Mean Square Error (MSE), Mean Absolute Error (MAE), Frenchet Inception Distance (FID), Peak Signal to Noise Ratio (PSNR) and Structure Similarity Index Measure (SSIM), are used to quantify the performance against the state of the art ([152, 124, 227]). Table 6.1 shows the quantitative evaluation for the inpainted images with one of ours in bold.

The high values obtained for MSE, MAE and FID show poor performance of the model whereas lower values for these metrics indicate better performance. For clarity, it is included on the table that † lower is better and ⊎ higher is better. PSNR and SSIM with higher values will indicate the prediction is closer to the ground-truth image, which will have a maximum score value of 1.

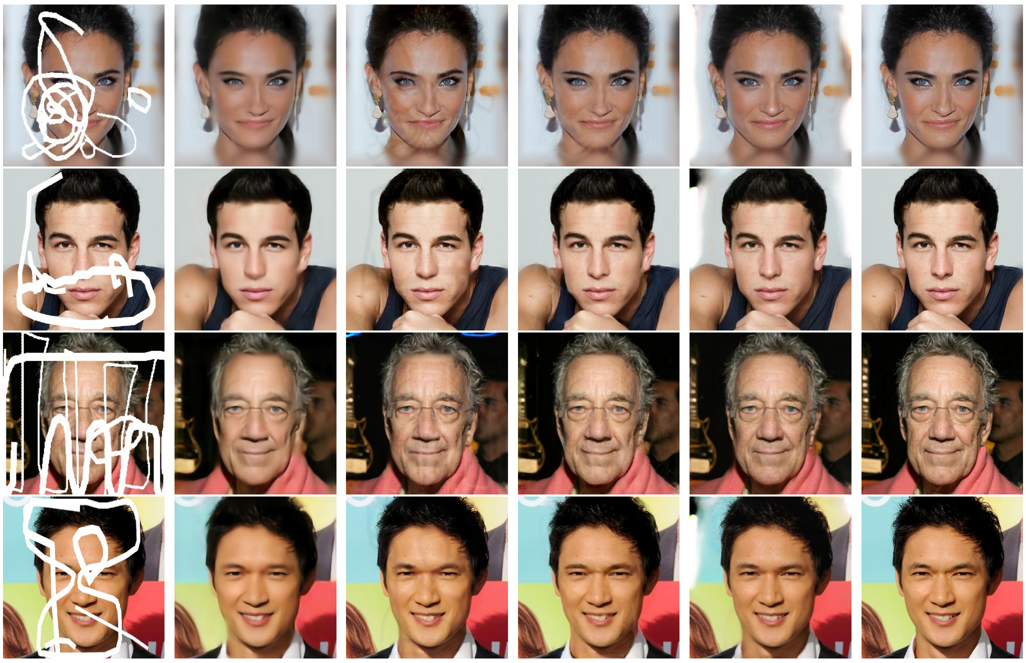
Our proposed method achieved the best SSIM score (tied with GC) and second performer in the majority of other metrics. This quantitative measures showed our method preserve the structure of the face. To further investigate, quantitative measures are performed on the foreground inpainted face and hair regions only, as shown on Table 6.2, with our proposed model outperformed the state-of-the-art models.

6.6.2 Qualitative Results

In this section, visual comparisons of the proposed model compared with the state-of-the-art are demonstrated. Without bias and based on code availability, the Pathak et al. [152] (**CE**), Liu et al. [124] (**PC**), Yu et al. [227] (**GC**) are used to measure against our model. From Figure 6.3, **CE** struggles with arbitrary hole-to-image mask

Table 6.2: Quantitative comparison of various performance assessment metrics on 3,000 test images from the CelebA-HQ dataset on Foreground inpainted regions. † Lower is better. ⊎ Higher is better.

Performance Assessment						
Method	Author	MSE †	MAE †	FID†	PSNR ⊎	SSIM ⊎
CE	Pathak et al. [152]	133.481	129.30	27.38	27.71	0.76
PC	Liu et al. [124]	102.72	4.35	7.99	29.24	0.87
GC	Yu et al. [227]	29.14	**1.47**	2.23	35.33	0.95
Proposed	Foreground-guided	**26.01**	2.58	**1.19**	**37.38**	**0.96**



(a) Input (b) CE (c) PC (d) GC (e) Proposed (f) GT

Figure 6.3: Qualitative comparison of our proposed model (e) with the state-of-the-art methods on CelebA-HQ, using the Quick-Draw binary mask dataset by Iskakov et al. [84] as masking method: (a) **Input masked-image**; (b) **CE** [152]; (c) **PC** [124]; (d) **GC**; (e) **Proposed**; and (f) Ground-truth image.

regions and the generated image is blurry, while **PC** and **GC** leave a bit of artefacts (best viewed when zoomed) on the generated image. Focusing on the face and hair regions, our model performs better than the state-of-the-art with no artefacts left on the inpainted regions. However, despite marginally comparable quantitative results on full inpainted images, our model completes and generates the facial image with no visible boundaries of the binary masks as seen on the generated images completed by the state of the art.

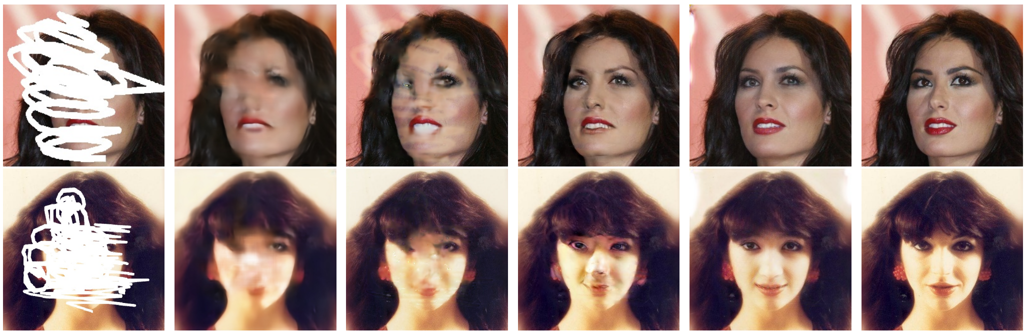

(a) Input (b) CE (c) PC (d) GC (e) Proposed (f) GT

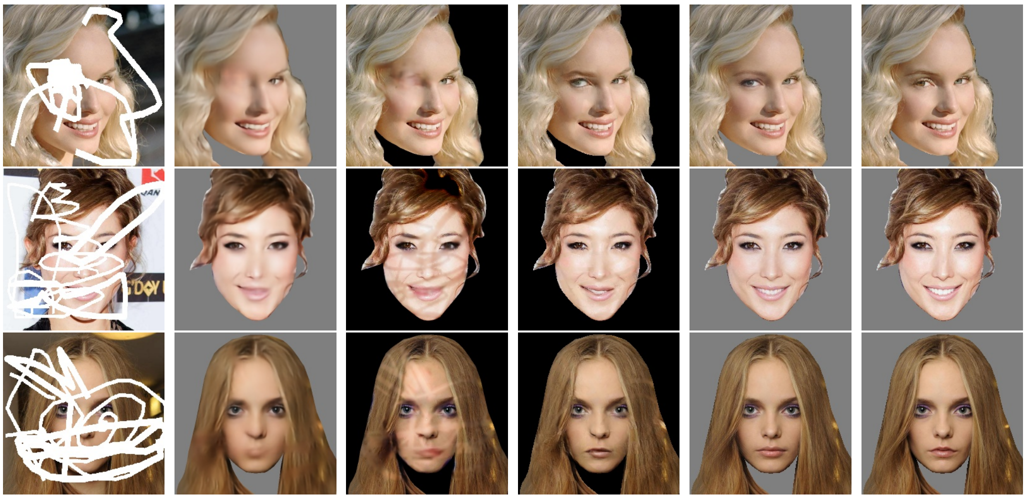
Figure 6.4: Qualitative comparison segmented Foreground Inpainted regions of our proposed method with the state-of-the-art methods on CelebA-HQ, using the Quick-Draw binary mask dataset by Iskakov et al. [84] as masking method: (a) **Input masked-image**; (b) **CE** [152]; (c) **PC** [124]; (d) **GC**; (e) **Ours**; and (f) Ground-truth image.

6.7 Semantic Inpainting with Fidelity Preservation

The qualitative results in Figure 6.3 showed the performance of our model has great visual quality when compared to the state of the art. To further show reasonable semantic understanding of predictions, our model can fill-in high-level textural and structural information as seen in Figure 6.4 where other methods have failed. As seen on the Figure 6.4, the lip region on the image inpainted by our model is fully recovered with and our model shows a full semantic understanding of the image by putting a broader smile as compared to the original input. Furthermore, on the same Figure 6.4, the earring, nose and eye regions are fully recovered with contextualised features with our model. This also show that our model has high semantic understanding of facial features when trained with the joint loss function. Thus semantic understanding of features in latent space significantly improves the visual quality of generated facial components, which is further supported by Figure 6.5. When focuses on the face and hair regions only, the first row of Figure 6.5 illustrates the ability of our method in predicting the missing eye region and the others show accurate prediction of mouth regions.

6.8 Summary

This chapter introduced a method to inpaint missing region(s) within an image using foreground guidance. The results obtained suggests the importance of foreground



(a) Input (b) CE (c) PC (d) GC (e) Proposed (f) GT

Figure 6.5: Qualitative comparison of our proposed model with the state-of-the-art methods on CelebA-HQ, using the Quick-Draw binary mask dataset by Iskakov et al. [84] as masking method: (a) **Input masked-image**; (b) **CE** [152]; (c) **PC** [124]; (d) **GC**; (e) **Proposed**; and (f) Ground-truth image.

guidance training for the prediction of challenging corrupted patches on an image. It has been shown that the proposed model can predict and reconstruct plausible and realistic features while preserving the realism of the faces. Our model can produce high-quality visual results that meet the goal of real-world in-the-wild scenarios. Further exploitation of foreground pixels is a promising foundation for future inpainting tasks. In this chapter, the significance of high-quality inpainting with fidelity preservation is demonstrated. This technique, however, is limited to faces because it is based on semantic reasoning of disentangled pixels represented in the foreground segmentation mask introduced within the network as a loss model. To generalise high-quality generated images, there must be a way to train the model to learn high-level features while not missing important subtle features required to generate the inpainted regions. In the following chapter, the network designed is considered with significant design implementations proposed specifically to highlight and extract high-quality features while transitioning easily to the decoder. The aim is to generalise this network to natural and building scene datasets.

Chapter 7

V-LinkNet: Learning Contextual Inpainting Across Latent Space of Generative Adversarial Network

In this chapter, the latent space of two encoders is used as a loss model in our proposed V-LinkNet, a GAN-based technique for learning contextual inpainting. A novel transition layer is introduced for feature transfer and explore the components of V-LinkNet and their importance to the network using experimental methods. To show performance discrepancies, different faces with the same mask and other masks with different faces are experimented with the proposed inpainting algorithm. This is because inpainting performance varies based on image conditions. Lighting, backdrop, and facial features can all affect the prediction of the missing region. A face with lots of illumination or posture change will underperform. In order to improve reproducibility and enhance inpainting research, a standardised protocol is proposed for identifying biases with different masks and images with varying background and textural alterations.

7.1 Introduction

Multi-column latent space learning [205, 210] have achieved the state-of-the-art results in many research areas [33, 27, 35]. GAN models, such as [205, 125, 220] have used multi-columns to encode and propagated features directly to the decoder or use a self-supervised Siamese style inference approach [210], where a style encoder is the supervisor of the generator. Existing works [124, 120, 126, 226, 227, 87] have shown that GAN coupled encoder-decoder models can achieve visually-reasonable content of the missing regions, which are semantically-consistent with the entire image.

The complexity of these methods may be a disadvantage because they are stacked differently depending on the network designer. Another possible explanation is that, due to the nature of the design, insufficient full resolution details (latent feature encodings) of the missing parts may be missed or failed to transfer to the decoder, or large missing parts require longer periods of training. Alternative approaches to image inpainting are two-stage models, where a coarse version is generated and then used as the input to a refinement network. Another issue is lack of adequate information during reconstruction (as already mentioned in Chapter 6), due to larger target pixel region. It is also tougher to construct high-dimensional distribution from natural scenes than from aligned faces with no visible aberrations. It is noted that a bottleneck [242, 219, 118] or a feature transfer mechanism like attention layers within deep layers of convolutions is often required when inpainting high-resolution images. Because high-resolution images have large feature maps, training takes longer if there isn't adequate GPU memory [152, 123, 229]. Additionally, when extracting features from high-resolution images, some features may be lost during the operation. However, more detailed information about low-level features, such as edges, is frequently captured within the first few layers of the convolution. As a result, failure to consider prior semantic distributions leads in unusual textures on the generated image.

So far solutions to some inpainting problems have been proposed in previous chapters, although not all limitations were solved. The first solution (Chapter 4) handled the task with high-resolution images, but it depletes computational resources. Furthermore, this model lacks the ability to focus solely on mask regions, which is why the proposed solution in Chapter 5 was introduced. The proposed solution in Chapter 5 achieved state-of-the-art results, but was limited in its ability to extract subtle textural features of the face and generate images that preserve natural and unnatural facial attributes, as well as expression. It was observed that the limitation of this model to inpaint images with large textural regions was due to the design's fewer layers when compared to Chapter 4. Nonetheless, I continued to search for a solution to this problem and proposed the model in Chapter 6, which is capable of extracting subtle textural features and generating images with fidelity preservation. Because it uses the segmentation mask from [115] to provide clues during backpropagation, this model can only handle faces. However, research opened up in this direction to design a model that can extract these features without requiring a semantic segmentation mask and that can generalise to natural or building scene images.

For reproducibility, the datasets providers for inpainting research split the images to training set and testing set. However, the pairing of the masks are mostly random and lack a standard protocol. For example, a tiny mask region and large mask region on the same image will have different results. Biases in quantitative and qualitative evaluations were identified for different masks and images with varying background and textural alterations. This is because image inpainting performance varies depending on image conditions. Lighting, backdrop, and facial features can all affect the prediction of missing region. Thus a face with lots of illumination or an angled posture facial image with complex background will underperform. Figure 7.1 illustrates the inpainting results of a mask on different faces. It is noted that the results are varied depending to the occluded regions. While the middle image of the second row from the left achieved SSIM of 0.93, the image next to the ground-truth image achieved poor results with SSIM of 0.89. Figure 7.2 shows another issue when inpainting is performed on faces occluded by different masks. With different masks, the inpainted results have significant discrepancies, hence the reason to propose a standard protocol for testing in this chapter.

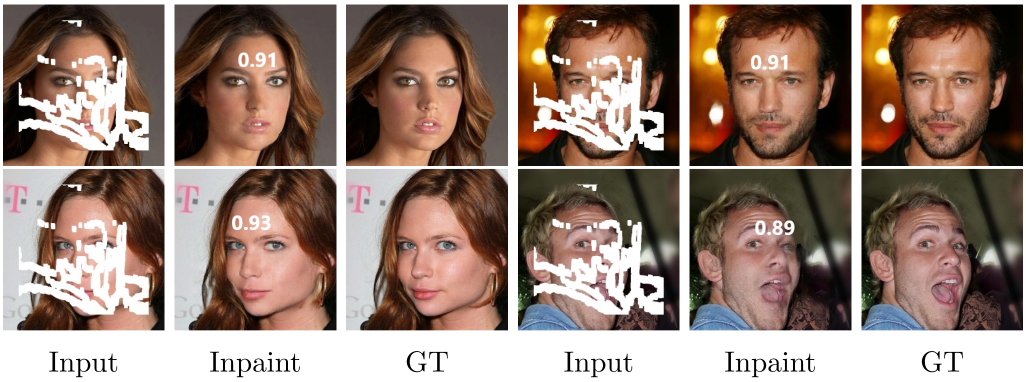


Figure 7.1: Illustration of inpainted results with one mask on different faces.Note the changes in SSIM value for different faces with the same mask.

In this chapter, a dual-encoder network approach is considered and introduces a new learning strategy for both models to communicate with each other. Here, the V-LinkNet is presented; a cross-latent space reverse mapping GAN for image inpainting. Below, the main contributions are highlighted:

- An end-to-end learning across latent space that uses feature information to encode fine details to complete the missing regions.

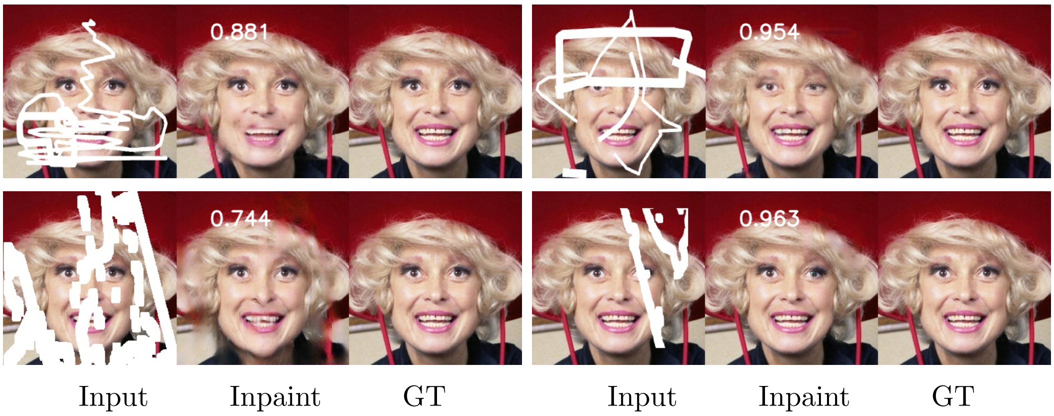


Figure 7.2: Inpainted images for the same image, different masks show different performance. See SSIM values on the inpainted images.

- A recursive residual transition layer is designed to capture high-level features in a similar manner as maxpooling units within convolutions with feature preservation and transfer technique employed as a ResNet-like unit within the block.
- A standard protocol by introducing testing sets with paired masks and images, which will be made available for the image inpainting community.

An ablation study is conducted to validate the results of the proposed solution and demonstrate that our results achieve better performance than the state of the art methods.

7.2 Connection to Related Work

7.2.1 Attention Transfer

The concept of contextual optimization of features to enhance textures within the target region for inpainting is a trending topic in image inpainting. To obtain best features, new techniques that use attention or residual learning continuously being improved. Some of these methods [226, 227, 200, 239, 211, 126, 228, 201, 238, 243, 126] have demonstrated the effectiveness of attention. Contextual attention [226, 227] has shown that a deep learning model to search for long range information to fill in missing regions. Despite the high performance already seen with these methods, there is still room to improve as swapping patches is challenging and can lead to inaccurate results.

7.2.2 Style Transfer

Given two image domains, a style transfer task can change the style of an image in one domain using an image of the other domain as reference[37]. One domain can be the content and other the style reference image. In the context of image inpainting, the content and style of the known region can be used to fill in the hole regions. Recently, style transfer has been used as a style-objective function to optimize CNN inpainting models [210]. Other style methods have been used, for example instead of using Gram matrix, the method in [116] have used MRF and [205] introduced implicit diversified MRF (ID-MRF) as an alternative for style transfer. To expand further, [205] used cosine similarity measure and pre-trained VGG19 network [183] to compare patches and compute the loss. With the cosine similarity measure, the objective function searches nearest neighbours for different generated feature patches.

7.3 Method

The V-LinkNet is a CNN with two encoders of different weights, a recursive residual transition layer and a decoder as the generator. Our network is a GAN that employs a global and local Wasserstein discriminator. A global and local WGAN discriminator [9] are used so that the model can generate realistic images. Consider inpainting as a masked image obtained by applying matrix operands on the mask and ground-truth images from the same distribution (training data). The inpainted (generated) image is the conditional distribution, the ground-truth is the marginal distribution, and the masked image is the joint distribution. Within the generator, one encoder emphasises on encoding contextual information and the other on preserving the image details of the unmasked regions. On one encoder, neurons are disconnected within the convolutions layers to allow the network to focus on missing regions, thus ensuring full contextual details are captured. Between the two encoders, a loss function is implemented to force the latent space to minimise the error of encoded features from both encoders. At the end of the two encoders, the outputs of both encoders are concatenated so as to obtain high level features encoded by both encoders. Just like morphological processing where two images are blended for erosion, features of both encoders are considered as two images, which can be blended to extract high-quality features, thus eliminating mask residues via erosion. This is achieved by designing a recursive residual transition layer to highlight and extract high-quality fine-grained features from the fused features of both encoders. Our recursive residual transition layer is a mini-ResNet block with residual maxpooling units and residual convolution

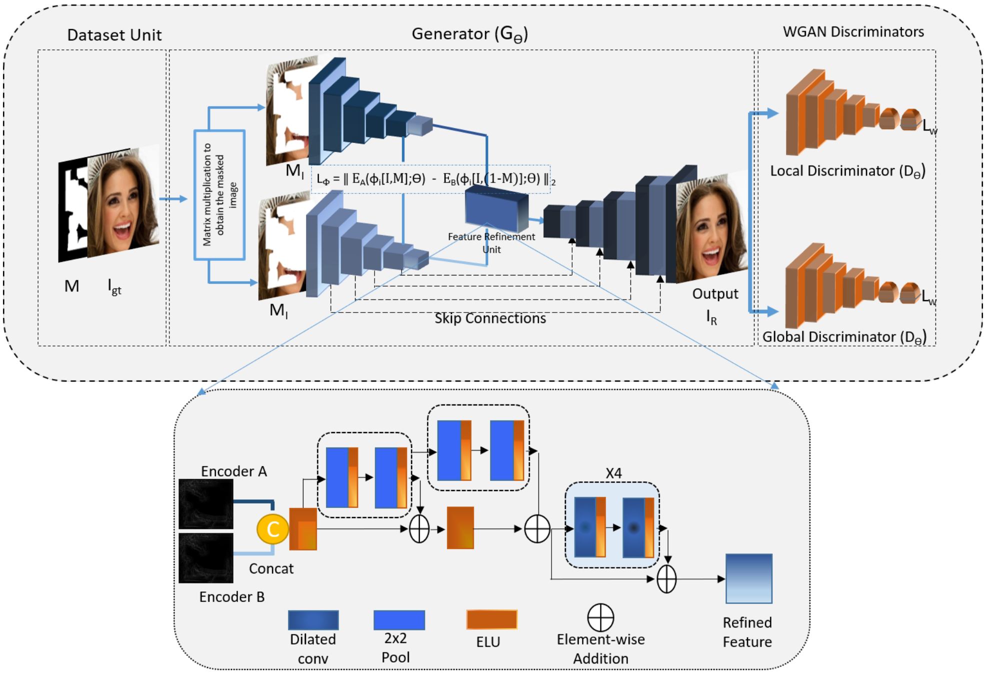


Figure 7.3: Overview of our proposed architecture during training. The proposed feature fusion refinement block passes refined features to learnable upsample layers within the decoder ($D_{\theta_E}(\cdot)$). The connected residual pooling refinement unit is further illustrated in 7.3.3. See section 7.5.1 for detailed implementation.

interlinked together to output fine-grained features with high-level semantic information for the decoder. The pooling unit within this layer emphasises on the contextual information from the fused features to capture more high-level background details of high-resolution feature maps [136]. The intuition is to ensure that the generated images satisfy semantic and visual consistency with preserved ground-truth realism.

The key components of this network are the loss function, context encoder and the recursive residual transition layer. The loss model is specifically designed to use reversed feature mapping in latent space (i.e performing mask reversal and use it as the input to the loss model). Details of the proposed method are shown in Figure 7.3, and the feature refinement unit illustrated in Figure 7.4. Lastly, the loss functions used during the training process are disucessed.

7.3.1 Problem Formulation

Previous works [152, 82, 124, 216] in computer vision have shown that inpainting is a learning problem that can be solved by encoding high-level features. The reconstructed output is geared towards having a close similarity to the input. Consider the inpainting task to have an input source $m_x = X \odot M$ and a target image X, where M is the binary mask. Te V-LinkNet is trained on the training set of X and use high-level features within both encoders to minimise the error. Midway between the paired encoders, the convolution blocks are modified from that of previous chapters4 6 by increasing the dilation rate from 8 and 16. Both encoders $E_{\theta_A}(\cdot)$ and $E_{\theta_B}(\cdot)$ learn high-level features to obtain output features $E_{\theta_A}(\phi)$ and $E_{\theta_B}(\phi)$ which are passed into a recursive residual transition layer. The recursive residual transition layer designed to fuse the features from both encoders to exploit feature information at different scales. The V-LinkNet learns through latent space loss and adversarial loss to reconstruct images with similar pixel values of the target image.

7.3.2 V-LinkNet Architecture

Our proposed V-LinkNet is a generative model consisting of a generator, a global and local discriminator as shown in Figure 7.3. The discriminators are included for adversarial training. Only the generator network is used during the testing phase.

The generator G_θ has dual encoder branches ($g_{\theta_A}(\cdot)$ and $g_{\theta_B}(\cdot)$) and a decoder ($f_{\theta_E}(\cdot)$). Within G_θ, encoder branch $g_{\theta_A}(\cdot)$ focuses on the capturing contextual information covered by the masked (unknown) regions. To ensure the reconstructed image is visually coherent with the structure and context of the ground-truth, $g_{\theta_B}(\cdot)$ is designed to capture encoding with main focus on perceptual and structural information. To ensure high-quality contextual features for missing regions, a loss is introduced between $f_{\theta_E}(\cdot)$ and $g_{\theta_A}(\cdot)$. During training, the loss between both encoders ensures ongoing communication in order to improve the models learning on contextual information. By employing this technique, the model coupled with other components can enhance visual consistency with contextualised features. The loss model is designed based on the MSE specifically to penalize large errors and provide fast learning. Both encoders $g_{\theta_A}(\cdot)$ and $g_{\theta_B}(\cdot)$ have eight convolution blocks, each with variations in spatial resolution and receptive fields at dilation rates of 2, 4, 8, 16. $g_{\theta_A}(\cdot)$ has dropout layers after each convolution block to reinforce learning. Block one to five, have batch normalization and Exponential Linear Unit (ELU) activation followed by maxpooling layer, while block six to eight has ELU and dropout. Within the decoder $f_{\theta_E}(\cdot)$, are learnable upsampling layers using bilinear interpolation each with a convolution block that includes batch normalization and ELU activation layers. The final convolution block has a Tanh activation layer with no batch normalization layer which is deliberate so as to accelerate training and stabilize learning. The output of the final layer $I_R = f_{\theta_E}(g_\theta)$ and the generator output as $G_\theta(I_R)$. The final output a generated image based on nonlinear weighted upsampling in latent space.

V-LinkNet consist of a training and inference (testing) phase, where the training sample are masks and ground-truth images. During training, the network learns with the main objective being to generate an image given the mask and the ground-truth. To minimise the error through back propagation, a suitably designed loss function is implemented, discussed in 7.4.2.1 that evaluates the training set to minimise the error to find high-quality matching features between the paired encoders. The corrupted image is projected onto the latent space of the generator through iterative backpropagation. The weights of both encoders are reused to compute an objective function that will specifically target valid regions. At each stage of the network training, the weights will assist with fast updating during learning to guide the model.

7.3.3 Recursive Residual Transition Layer

Residual learning has been well established in deep learning due to their ability to reduce training error in much deeper layers. A simple implementation of a residual block is a fast-forwarded activation layer within the neural network. By adding the activation layer of a previous layer, to a deeper layer within the network, a residual connection is achieved. In previous Chapters 4,5 and 6 it is observed that large portions of the background are not fully captured during feature abstractions within the decoder. Hence the reason why maxpooling is considered, an operation that highlights the most present feature of an image patch and calculates its maximum value. Because features encode spatial representation of visible patterns, it is more informative to consider the maximum presence of different features extracted from the image. Hence the reason why max pooling is considered instead of average pooling in this work. To capture much information a chained residual refinement unit is designed, aimed to capture large image background regions. The idea is to efficiently pool multiple window sizes, combine them using learnable weights and fast-forward to deeper layers to reduce the error during training. As a result, training gradients are obtained from the next connected layer within each layer, and these gradients are used to update the parameters in the current layer. This, in turn, influences the weights of the filters, causing the activation maps to increase or decrease, lowering the loss. The residual connection of the activation maps are combined with the output of the final pooling layer and the input of the residual layer to obtain the unit output feature map. This feature map is refined structural and textural information from both encoders that are propagated from known regions to facilitate image completion naturally during reconstruction at the decoder. Our proposed connected pooling operation combined with residual connections reduces the error near the boundary

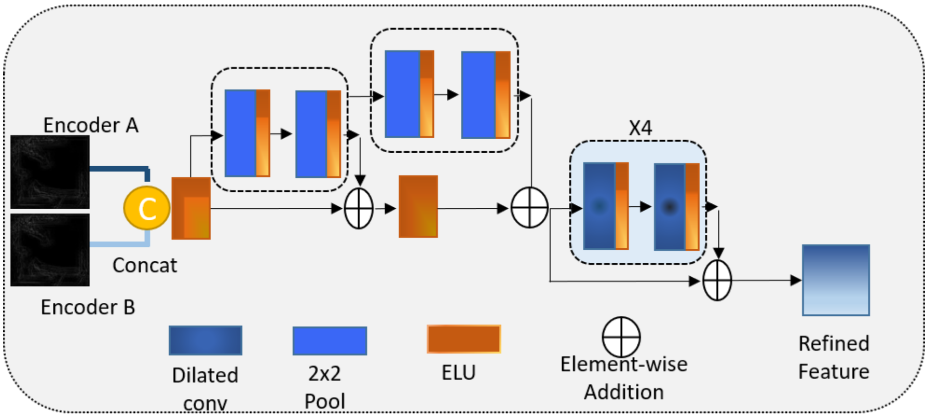


Figure 7.4: Illustration of connected residual pooling. This unit utilizes maxpooling with pool-size of 2×2 and ElU activation function as gating. The connected residual network uses dilated convolutions for refinement.

regions of the hole regions, as it fills it in with fine contextual information. This unit is designed to predict and delineate the binary mask regions as highlighted by the Sobel operator on the input features. The concatenated features when passed via pooling unit suppresses noise to project informative pixels. This is different from using the channel-wise attention which squeezes the spatial dimension of the feature map. The objective of this design is to extract meaningful pixels whilst suppressing uninformative ones before passing them to the decoder. First the concatenated features extracted from $E_{\theta_A}(\cdot)$ and $E_{\theta_B}(\cdot)$ are passed through ELU activation and then perform pooling operation followed by a 3×3 convolution and ELU unit as a gating layer. For more refined details a dilation rate of 16 is used within the convolution layer of this unit.

$$F_\phi(X_i) = \sigma([M_{pool}]F_{3\times3}) \tag{7.1}$$

where σ is the ELU activation function, $F_{3\times3}$ is the convolution layer. The final feature map is given by:

$$X_\phi = A_\phi(F_\phi(X_i) \oplus X_i) \tag{7.2}$$

where $\oplus$ is element-wise addition and lastly a dilated convolution layer to refine to transfer the feature to the decoder.

7.4 Loss Function

7.4.1 Feature Losses

More recent approaches [227, 136, 224, 229] use pre-trained VGG16 or VGG19 [183] to evaluate or enhance the perceptual quality of image inpainting results. These models [205, 227, 238, 224] have perceptual and style losses and these losses are still undisputed when it comes to evaluating or improving the overall performance of the generator. Inspired by perceptual losses in feature space, a novel feature loss in latent space is proposed. Features are low-dimensional latent state representations captured in latent space. By reusing deeper-level features from both encoders in latent space, an objective learning loss model is designed to capture rich features of the reversed regions covered by the mask. In addition, it is desirable for the inpainted regions to be as close to the counterpart regions of the ground-truth. Thus a head-start with faster update of parameters and weights to the generator is important in this task. The reasons for this is that both encoders will learn from each other. Another reason is that loss functions can become difficult in latent representations, thus reusing latent representations for cost functions give the network easy access to compute the gradients for better head-start. One encoder will take the compliment of the reversed masked region and the other will take in the masked region. The textural feature of the encoders will have understanding from the reversed input while the masked region will preserve the regions under the mask. This allows the network to learn by correcting each other from a reasonable prediction. Moreover, there is no guarantee that continuous training will accurately capture more refined features in space for this specific task. Thus using all layers to find a better gradient computation increases computational complexity hence why only one deeper identical layers of both encoders are used for this experiment.

7.4.2 Latent space feature-aware gradient loss

Utilizing image gradients is very common practice of various image processing algorithms [234, 80]. In image inpainting, known and unknown regions are representations of the masked image, thus applying gradient algorithm to detect occlusion boundaries can prove useful. Image gradients highlight directional change in images and can be used in edge detection algorithms [213]. [17, 40, 197] by making use of edge information. Diffusion-based inpainting, uses fluid dynamics and partial differential equations to propagate information along the edges from known to unknown regions. Because edges are continuous, information travelling along isophote (line joining points with the same pixel level intensity) match gradient vectors at the boundary between the missing pixels and known pixels. However, the use of edge information vary based on hyperparameter and the choice of edge detector. The Sobel operator is not new to image inpainting [170]. The Sobel algorithm [96] is capable of extracting occlusion boundaries because images contain noise which can generate a sudden transition of pixel values [234]. During edge detection with the Sobel operator, noise can be suppressed without removing edges, edges are enhanced using a high pass filter and elimination of spurious edges which are noise related (edge localisation). In [170] the Sobel operator is utilized to obtain gradient information of generated and ground-truth images to compute a loss function. The loss function measures the quality of the generated output based on edge information. Another utilization of gradient information is proposed in [232] where the gradient map of the mask and masked image are utilized within the network to obtain gradient features, which are later fused with image features to obtain the final image. To determine the direction of filling priority, the model is designed to target the image gradients of feature maps and use this information to construct a loss model.

Based on this, a variation of our proposed latent space loss is computed, namely feature-wise latent space gradient loss with depth-wise convolutions, using the Sobel operator. The goal of using feature-wise gradient is to investigate filling-in large missing regions that require depth penetration while minimising computational complexity. Feature gradients of the third convolutional layer of both encoders are obtained to compute the loss model. To re-enforce on outer edges and fidelity of the generated image, image gradients of the generated and ground-truth images are used to assist in the final reconstruction. Note that the edge map based on the gradients are computed in y and x directions.

7.4.2.1 Generator Loss

The generator loss evaluates the missing pixel region and the perceptual quality of the image. To maximise contextual and feature-wise learning, high-level features are extracted from deeper layers of $E_{\theta_A}(\cdot)$ and corresponding features of $E_{\theta_B}(\cdot)$. To help the model learn, the image, mask and reversed mask are included as part of the loss function so that it can actually penalize the error during encoding and reconstruction for better inference.

$$L_\phi = ||(E_{\theta_A}\phi[M, I_{gt}] - E_{\theta_B}\phi[(1-M), I_{gt}])||_2^2 \tag{7.3}$$

$$G_x = I_{gt} * X_{edge}(i,j), G_y = I_{gt} * Y_{edge}(i,j) \tag{7.4}$$

where G_x and G_y are gradients computed by depth-wise convolution using the x and y components of the edge operator on an image I_{gt}.

$$\nabla I_{gt} = \sqrt{G_x^2 + G_y^2} \tag{7.5}$$

$$L_{edge} = ||\nabla I_{gt} - [G_\theta(\nabla I_R)]||_2^2 \tag{7.6}$$

$$\nabla \phi = ||\nabla E_{\theta_A} \phi[M, I_{gt}] - E_{\theta_B} \phi[(1-M), I_{gt}]||_2^2 \tag{7.7}$$

$$L_{edgeLoss} = \lambda \cdot L_\phi + (1-\lambda) \cdot L_{edge} \tag{7.8}$$

Where $\lambda = 0.5$ as coefficient to obtain $L_{edgeLoss}$. The pixel space L1-norm, based on a range of pixel values with the input image and output image is used here.

$$L_{pix} = ||K \odot (I_{gt}(i,j) - I_R(i,j))||_1^1 \tag{7.9}$$

$$L_{vgg} = ||\phi[I_{gt}] - \phi[G_\theta(I_R)]||_2^2 \tag{7.10}$$

Further, the reversed mask loss (L_{rm}) from 5 is used to compute a contextual loss. Here the aim is to keep the known pixel locations of the input image by penalizing the predictions thus creating similar pixels based on the reversed mask and masked regions.

$$I_c = M \odot I_{gt} + (1-M) \odot G(z) \tag{7.11}$$

$$L_{rm} = ||\phi(1 - M \odot I_{gt}) - \phi(I_C||_1^1 \tag{7.12}$$

$$L_c = ||\phi(1 - M \odot I_{gt}) - \phi(I_R \odot M||_2^2 \tag{7.13}$$

The total loss (L_T) is a weighted sum of all the losses with highest weight applied to L_ϕ.

$$L_T = \alpha_1 \cdot L_{vgg} + \alpha_2 \cdot L_{rm} + \alpha_3 \cdot L_{pix} \tag{7.14}$$

where $\alpha_1 = 0.5, \alpha_2 = 0.3, \alpha_3 = 0.1$ are coefficients of the weights applied to the loss.

7.4.2.2 Discriminator Loss

For adversarial loss, he Wasserstein distance loss similar to [205, 87] used in both discriminators.

$$L_w = E_{I \sim P_x}[D_\theta(G_\theta(I_{gt}))] - E_{I_R \sim P_z}[D_\theta(G_\theta(I_R))] \tag{7.15}$$

where real-data distribution is represented in the first term and generated-data distribution is the second term. Finally, the overall objective loss function of the model is defined in Equation 7.16.

$$L_F = L_T + L_w \tag{7.16}$$

7.5 Experiments

In this section, a standard protocol for testing is proposed.

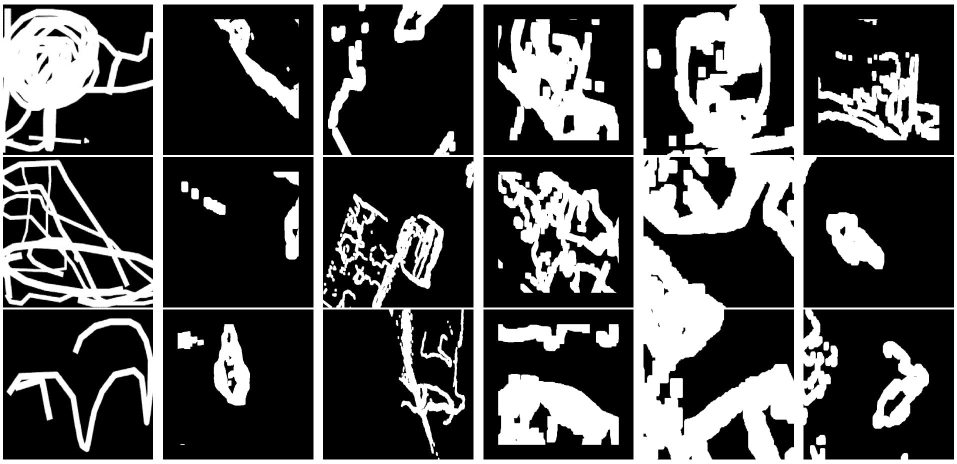
7.5.1 Implementation

Implementation of the V-LinkNet is done using the Keras library with a Tensorflow backend, and training of the model is done on a P6000 GPU computer. All images are scaled to 256 × 256 × 3 and 512 × 512 × 3 and align them with their appropriate masks. The network is pretrained using a novel loss function that backpropagates gradients using features from both encoders. The RSMProp optimizer, with a 0.0005 learning rate is used for pretraining the network. The generator and discriminator networks are updated following pretraining and utilised the Adam optimizer [103] with a learning rate of 0.00001 and a beta of 0.5. The network is trained with a batch size of 5 and for 100 epochs, which takes around three to five days depending on the amount of the training data. After obtaining a well-trained model, the reverse mask loss and a decreased learning rate (e.g., 1e5) is used to fine-tune it while retaining the original network topology. The input is updated throughout completion using a contextual loss and a perceptual loss with coefficients of 0.1 and 0.9, respectively. Stochastic clipping is employed during back-propagation. A modest value is considered to ensure that contextual loss is prioritised during test-time optimization, and that the inpainted part of the generated image most closely resembles the input background context of the entire image. The generator and discriminator are fixed during back-propagation. Note that due to the limited computational resources, the network is not pretrain for 512 × 512 images. CelebA-HQ, Paris Street View, and Places2 datasets were utilised, each having a comparable training and testing set suited to the state of the art. A fully trained model can predict missing pixels for image-to-mask ratios ranging from [0.1 to 0.8] during testing. The inference time can be between 0.192 seconds to 63 seconds depending the on mask size. During inference, the network design with batch normalisation layers disabled.

7.5.2 Datasets

7.5.2.1 Standard Protocol Testing Dataset

This section introduces the standard protocol test set for face inpainting and the mask test set (Figure 7.5 generalised to Paris Street View [45, 152] and Places2 [241] test sets. The proposed protocol share the filenames of the paired image and mask test set with comma-separated value (CSV) file. This test protocol set is labeled according to CelebA-HQ [99] that contains 3000 high-resolution face images from CelebA [131]. The masks are labeled following the test set of QuickDraw Mask [84] and Nvidia Mask [124] test set specific to each image. These images are paired and named on a CSV file. The proposed standardised test protocol is compiled with facial images, mostly profile faces with pose variations and various textural backdrops. The mask difficulty and image difficulty were investigated to support the notion that the complexity of the inpainting task is based on pose, backdrop, and mask hole variation: Figure 7.2 shows the various mask paired to specific image and the performance evaluation. For Places2 and Paris Street View datasets, the evaluations are based on the mask difficulty on the standard test set of these datasets. Note that our evaluation masks are set to 3000 mask images.



(a) **M1** (b) **M2** (c) **M3** (d) **M4** (e) **M5** (f) **M6**

Figure 7.5: Examples of our standardized protocol. (a) MaskDataset1 [0.001,0.6] **M1** [84] (b)MaskDataset2 [0.001,0.1] **M2** (c)MaskDataset3 [0.1,0.3] **M3** (d) MaskDataset4 [0.3,0.4] **M4** (e) MaskDataset5 [0.5,0.6] **M5** (f)MaskDataset6 [0.1,0.4] **M6**. Note that **M2** to **M6** are from the Nvidia Mask dataset [124]

7.5.2.2 Training Datasets

The training datasets are **CelebA-HQ** [99], **Paris Street View** [45] and **Places2** [241] used in the previous Chapters 4,5 and 6. Each training image for both Paris Street View and Places is resized to size 256 × 256 pixels which is then used as an input to our model.

7.6 Results

7.6.1 Baseline model Comparison

This section presents a quantitative and qualitative evaluation of the proposed V-LinkNet in comparison to state-of-the-art methods.

- **Context encoder-decoder framework (CE)** [152] introduced the channel-wise fully connected layer to solve the convolutional layer limitation associated with failures in direct connection of all locations within a specific feature map. The channel-wise fully connected layer is designed to directly link all activation; thus enabling propagation of information within the activation of a feature map.

- **Partial Convolution (PC)** [124] proposed partial convolutions with mask updating to enforce learning in irregular hole regions during convolutions and ease feature transfer to subsequent layers, allowing convolution layers to target more of the missing regions as a result.

- **Gated Convolution (GC)** The authors [227] proposed a gating mechanism that learns soft masks within convolutions to make the transfer of features within convolutions more convenient. This method differs from **PC** [124] in that the irregular mask is learnt rather than being updated in each step, whereas the former does not have this feature.

- **RMNet (RM)** Chapter 5 introduced reverse mask mechanism within the network. The reverse mask forces the convolutions to subtract visible regions through the reverse mask mechanism, thus ensuring the output prediction is on the missing regions only.

- **V-LinkNet** Pre-trained model with latent space loss and introduce recursive residual transition layer combined with perceptual losses and reverse mask loss. The recursive residual transition layer forces the model to focus on learning between disentangled pixels of valid and non-valid regions as it transfers highlighted high-quality features to the decoder.

7.6.2 Qualitative Results

The figures in this section depicts the visual results of the V-LinkNet method. Figure 7.8 depicts a visual comparison of the V-LinkNet method to the state of the art. Without bias and based on code availability, the Pathak et al. [152] (**CE**), Liu et

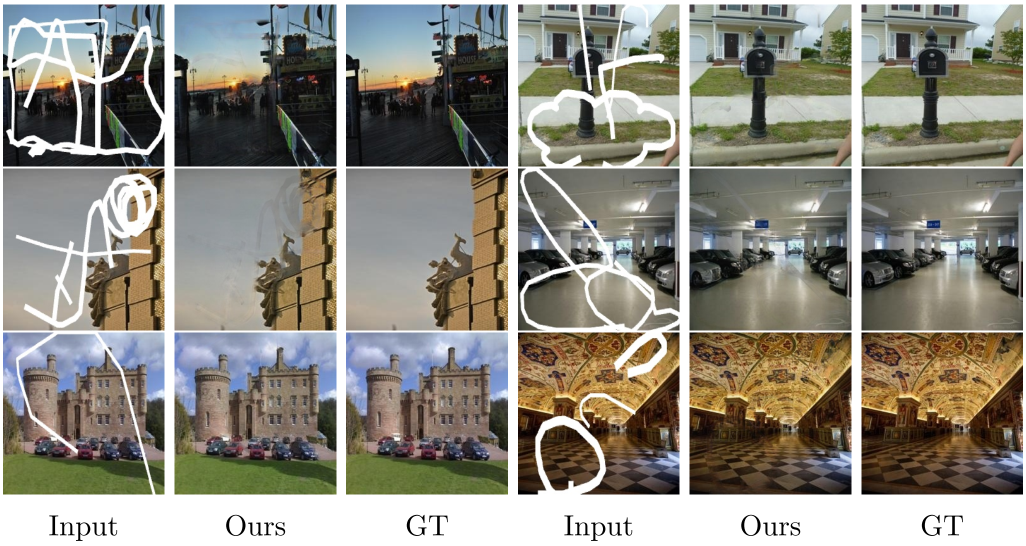


Figure 7.6: Results showing inpainted images using V-LinkNet on Places2 Dataset with MaskDataset1 of our standardized test set ranging from [0.1-0.6], where images in column (a) are the masked-image generated using the Quick-Draw Mask dataset [84]; images in column (b) are the results of inpainting by our proposed method; and images in column (c) are the ground-truth.

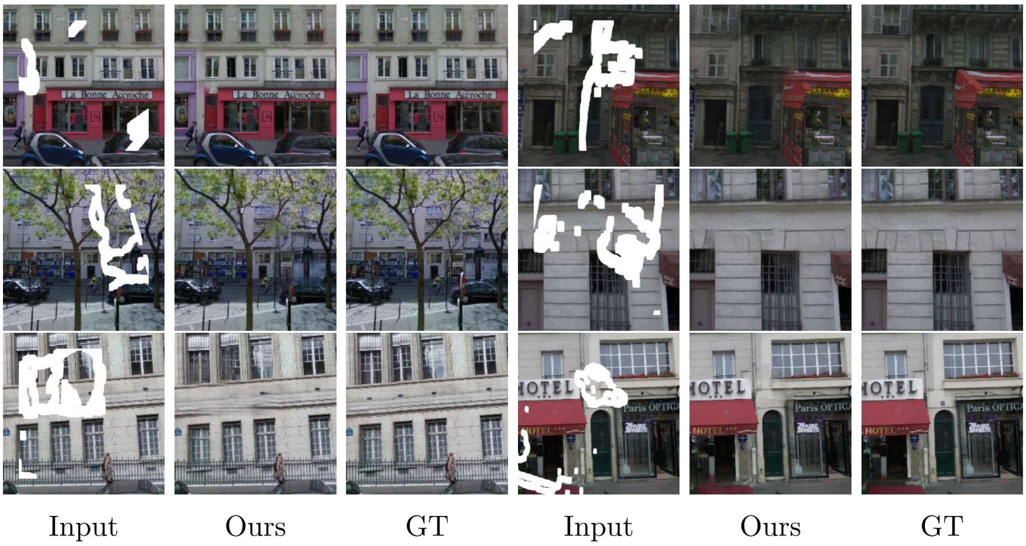


Figure 7.7: Visual results showing inpainted images using V-LinkNet on Paris Street View Dataset with MaskDataset2 of our standardized test set, where images in column (a) are the masked-image; images in column (b) are the results of inpainting by our proposed method; and images in column (c) are the ground-truth.

al. [124] (**PC**), Yu et al. [227] (**GC**) to measure against our model as per previous chapters 4, 5 and 6. The visual comparison of Places2 and Paris Street View datasets best represent how our model can generalise to natural scene images. The generated images are shown on Figure 7.6 for Places2 [240] while Figure 7.7 shows the inpainted images generated from Paris Street View dataset [125]. For comparison with the state-of-the-art and a variation of our model, the generated images are shown on Figure 7.8. Based on this Figure 7.8, **CE** struggles with arbitrary hole-to-image mask regions and the generated image is blurry, while **PC** and **GC** leave a bit of artefacts (best viewed when zoomed) on the generated image. Focusing on the face and hair regions, our model performs better than the state-of-the-art with no artefacts left on the inpainted regions. However, despite marginally comparable quantitative results on full inpainted images, our model completes and generates the facial image with no visible boundaries of the binary masks as seen on the generated images completed by the state of the art.

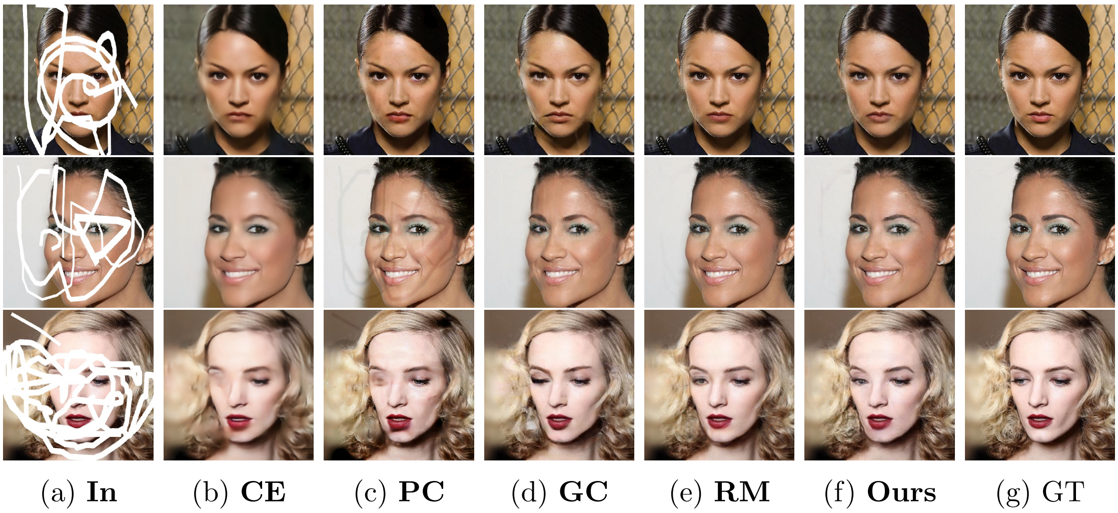

(a) **In** (b) **CE** (c) **PC** (d) **GC** (e) **RM** (f) **Ours** (g) GT

Figure 7.8: Visual comparison of the inpainted results by our models **Ours**, **CE**, **PC**, **GC** and **RM** on CelebA-HQ [124] where MaskDataset1 is used as masking method with mask hole-to-image ratios [0.01,0.6].

7.6.3 Quantitative Results

It is important to note that the visual and semantic understanding of the completed regions is critical to the audience when inpainting in the wild. This is because the visual quality of the blending between the inpainted regions and the original unmasked regions should be unnoticeable in real-world scenarios. For quantitative evaluation to track model performance and based on previous state-of-the-art research, the MAE, FID, PSNR, and SSIM are used to quantify performance against the state of the art ([152, 124, 227]). The high values obtained for MAE and FID show poor performance of the model whereas lower values for these metrics indicate better performance. For

Table 7.1: Quantitative comparison of various performance assessment metrics on 3,000 set 1 test images from the standardized protocol dataset. † Lower is better. ⊎ Higher is better.

Performance Assessment					
Method	Author	MAE †	FID†	PSNR ⊎	SSIM ⊎
CE	Pathak et al. [152]	129.96	29.96	32.61	0.69
PC	Liu et al. [124]	98.01	15.86	33.03	0.81
GC	Yu et al. [227]	43.10	4.29	39.96	0.92
RM	Chapter 5	**31.91**	3.09	**40.40**	0.94
V-LinkNet	V-LinkNet	37.97	**2.76**	39.75	**0.96**

clarity, the † stands for lower is better and ⊎ higher is better. PSNR and SSIM with higher values will indicate the prediction is closer to the ground-truth image, which will have a maximum score value of 1. Table 7.1 shows the quantitative evaluation for the inpainted images with two of ours in bold. Our proposed method achieved the best FID and SSIM.

To generalise this model, quantitative measures are evaluated for Places2 and Paris Street View datasets. The results are compared with that of 5 and the findings presented on Table 7.2. On the Paris Street View dataset, the proposed model outperformed that of Chapter 5, but failed on the Places2 dataset. The best results are highlighted in bold.

Table 7.2: The inpainting results of V-LinkNet on Paris Street View and Places2, where our standard protocol MaskDataset1 [84] is used as masking method with mask hole-to-image ratios range between [0.01,0.6]. The results are compared against the state-of-the-art [87]. † Lower is better. ⊎ Higher is better.

Performance Assessment					
Dataset	Method	MAE †	FID†	PSNR ⊎	SSIM ⊎
Paris Street View	RMNet 5	33.81	17.64	39.55	0.91
Paris Street View	V-LinkNet	**26.60**	**14.94**	**40.9**	**0.95**
Places2	RMNet 5	**27.77**	**4.47**	**39.66**	**0.93**
Places2	V-LinkNet	107.68	38.34	34.45	0.91

7.7 Ablation Study

To understand the proposed method, an investigation is carried out to demonstrate the effectiveness of each component contributing to the inpainting task. The visual results are shown on Figure 7.9 and the quantitative evaluations on Table 7.3. Another

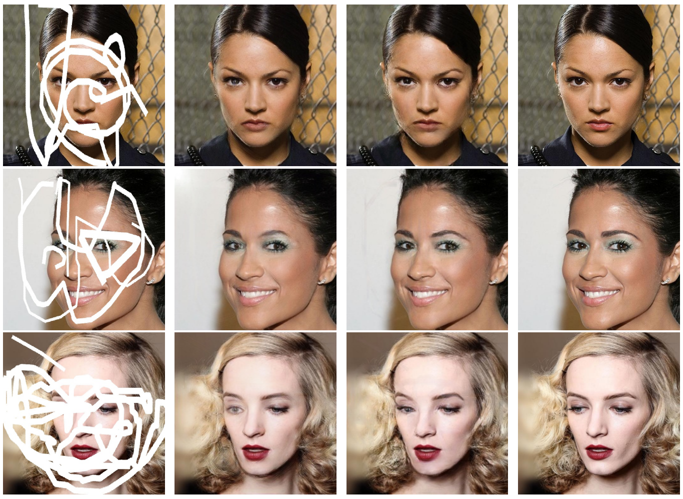
Table 7.3: Quantitative comparison of the performance assessment metrics on 3,000 images from the standard protocol test dataset. The masking method applied is based on MaskDataset1. † Lower is better. ⊎ Higher is better.

Performance Assessment					
Method	Author	MAE †	FID†	PSNR ⊎	SSIM ⊎
VN1	(V-LinkNet)	37.81	3.91	35.54	0.92
V-LinkNet	(Ours)	**37.97**	**2.76**	**39.75**	**0.96**

evaluation is carried out to assess the algorithm's performance across all datasets of the proposed standardised protocol.

7.7.1 Recursive Residual Transition Layer

This section examines whether residual features from our recursive residual pooling unit has a positive effect on our model. The results in Table 7.3 demonstrate that residual refinement has a positive impact on the overall performance of our model. According to our findings, this improvement is attributable to the elimination of low-level information as a result of the pooling units being interconnected residually, which allows direct backpropagation of high-level information throughout the learning process.



(a) **In** (b) **VN1** (c) **VN2** (d) GT

Figure 7.9: For ablation study, the inpainted results are compared by variations of the models **VN1**, **VN2**, on CelebA-HQ [124] where MaskDataset1 from our standardized set is used as masking method with mask hole-to-image ratios [0.01,0.6].

7.7.2 Latent space feature loss combined with edge-based gradient loss

A slight modification is performed on the recursive residual transition layer by removing the pooling unit. The modified layer is a residual block with the concept of attention in our inpainting task. This modification includes 1 × 1 convolutions on $g_{\theta_A}(\phi)$ and $g_{\theta_B}(\phi)$ output and concatenate the projected features maps. For dynamic feature selection, a softmax function is utilised on the concatenated feature map. Applying softmax after 1 × 1 convolutions on each encoder output enables precise feature values, thus preserving local and detailed information. During the experiment, the model uses L_{vgg}, $L_{edgeLoss}$ combined with L_1 pixel-wise reconstruction loss. It is observed that using the L_ϕ loss combined with $L_{edgeLoss}$ gets rid of checker-board artefacts on the generated image. Another observation is that the Sobel operator used to compute $L_{edgeLoss}$ helps with noise reduction and enhances the image quality of the generated output. However, the quantitative evaluation of the model with this loss is not great compared to the full model without the Sobel loss.

7.7.3 Quantitative evaluation of the standardized protocol test sets for celebA-HQ

This protocol is designed to evaluate the performance on a set of mask and images. The mask ratios in the Masksets range from [0.01,0.6]. The different MaskDataset and ratios are: MaskDataset1 [0.1,0.6], MaskDataset2 [0.01,0.1], MaskDataset3 [0.1,0.3], MaskDataset4 [0.3,0.4], MaskDataset5 [0.5,0.6] and MaskDataset6 [0.1,0.4].

Table 7.4: Summary of quantitative results of the proposed standardized test set on CelebA-HQ [98] and Paris Street View [45] datasets. The performance evaluation vary from maskset to imageset and are approximated to 2 decimal places. The results included are for distortions (image-to-mask ratio) between 10%-20% on image sizes 256 × 256. † Lower is better. ⊎ Higher is better.

Performance Assessment					
Dataset/Mask Ratio	Mask Type	MAE †	FID†	PSNR ⊎	SSIM ⊎
MaskDataset1 [0.01,0.6]	Irregular	37.97	2.76	39.75	0.96
MaskDataset2 [0.01,0.1]	Irregular	21.35	3.36	39.04	0.94
MaskDataset3 [0.1,0.3]	Irregular	33.64	5.23	36.53	0.91
MaskDataset4 [0.3,0.4]	Irregular	64.15	12.06	33.72	0.89
MaskDataset5 [0.5,0.6]	Irregular	107.33	15.82	31.90	0.74
MaskDataset6 [0.1,0.4]	Irregular	25.75	4.19	37.7	0.93

The MaskDataset6 are a selection of irregular masks that are used as masking method for more than one image (i.e one mask is applied on different images and evaluation carried out to show variations in inpainting performance). Each mask is evaluated on more than one image and the performance is different across the dataset. The overall results are shown in Table 7.4.

This study is conducted to identify biases for different masks on different images and propose a standard protocol that will propel research in inpainting. The mask-to-area ratio was determined using OpenCV toolbox. Based on this study, it is observed that the performance of an algorithm will very much depend on the mask type used and the image. There are some conditions on a facial image that can influence the performance such as the pose, the lighting, features and background. In the case of CelebA-HQ dataset, it is observed that if the mask is on the skin region, the performance evaluation has better scores compared to when the mask is applied on a difficult background with variations in lighting conditions. Compare the different quantitative performances of the standardised testing protocol on Table 7.4, where the MaskDataset2 (mostly tiny mask regions and a few large mask regions) applied to facial images shows an SSIM performance of 0.94, which is higher than the more difficult binary masks (MaskDataset5) applied to the same set of images with an SSIM performance of 0.74. Furthermore, the mask applied to a face posed at an angle will influence the results either positively or negatively. This is supported by the SSIM perfomance value of 0.93 obtained for the same mask on various facial images of the standard tests set during the experiment as shown on Table 7.4. It is based on this finding that the standardized testing protocol is proposed for research to progress in this direction. The proposed method's shortcomings are that it needs a larger GPU for training, using more computing resources, and that, unlike other approaches in the literature, it struggles to generate plausible textures for large missing areas.

7.8 Propagating High Level Features

For each spatial location, each convolutional layer expects a certainty. V-LinkNet handles high-level feature propagation as a learned operation within a residual unit designed with maxpooling units and a residual convolution unit to create the full layer. The proposed solution is simple and efficient, and it acts as a bridge to the decoder. After each convolution, ELU is used as the activation function, followed by maxpooling and batch normalization. To avoid exploding gradients, l2 regularizers at 1E-5 are used within the convolution blocks. The V-LinkNet model can propagate high-level features to the decoder using this unit. To validate its efficacy, an ablation study is conducted with various model components. The unmodified recursive residual transition layer combined with the losses excluding the Sobel gradient loss is found to be the best model combination. It is observed that combined recursive residual unit, which is linked to residual pooling and residual convolution, enables direct backpropagation within the bottleneck's deeper layers. This forces the selection of high-level information during decoder layer propagation, resulting in high-quality reconstruction of inpainted regions. Certainty, the propagation of high-level features has been completely learned and transferred to the decoder. Furthermore, the feature-wise loss model shared by both encoders aids the model during early learning, resulting in a better learning strategy shared by both encoders. The losses and Wasserstein discriminators improve the semantic consistency of our model, which ensures fine contextual information.

7.9 Summary

This chapter introduces a novel inpainting technique that uses two encoders to learn from one another in order to improve on previous inpainting methods. Furthermore, the dual-encoder approach takes advantage of semantic coherency across textural features in latent space. In addition, the loss model between encoders that uses features by both encoders provide clues from reversed regions to encourage high level feature abstractions. A newly designed recursive residual transition layer that fuses features of both encoders to enhance the projected textures and also serve as a feature propagation module is also introduced in this chapter. The proposed method is capable of producing high-quality semantic structural and textural features that match the overall image. Finally, a standardised protocol for testing is proposed in order to improve research and reduce the propagation of weak baselines in deep learning inpainting models.

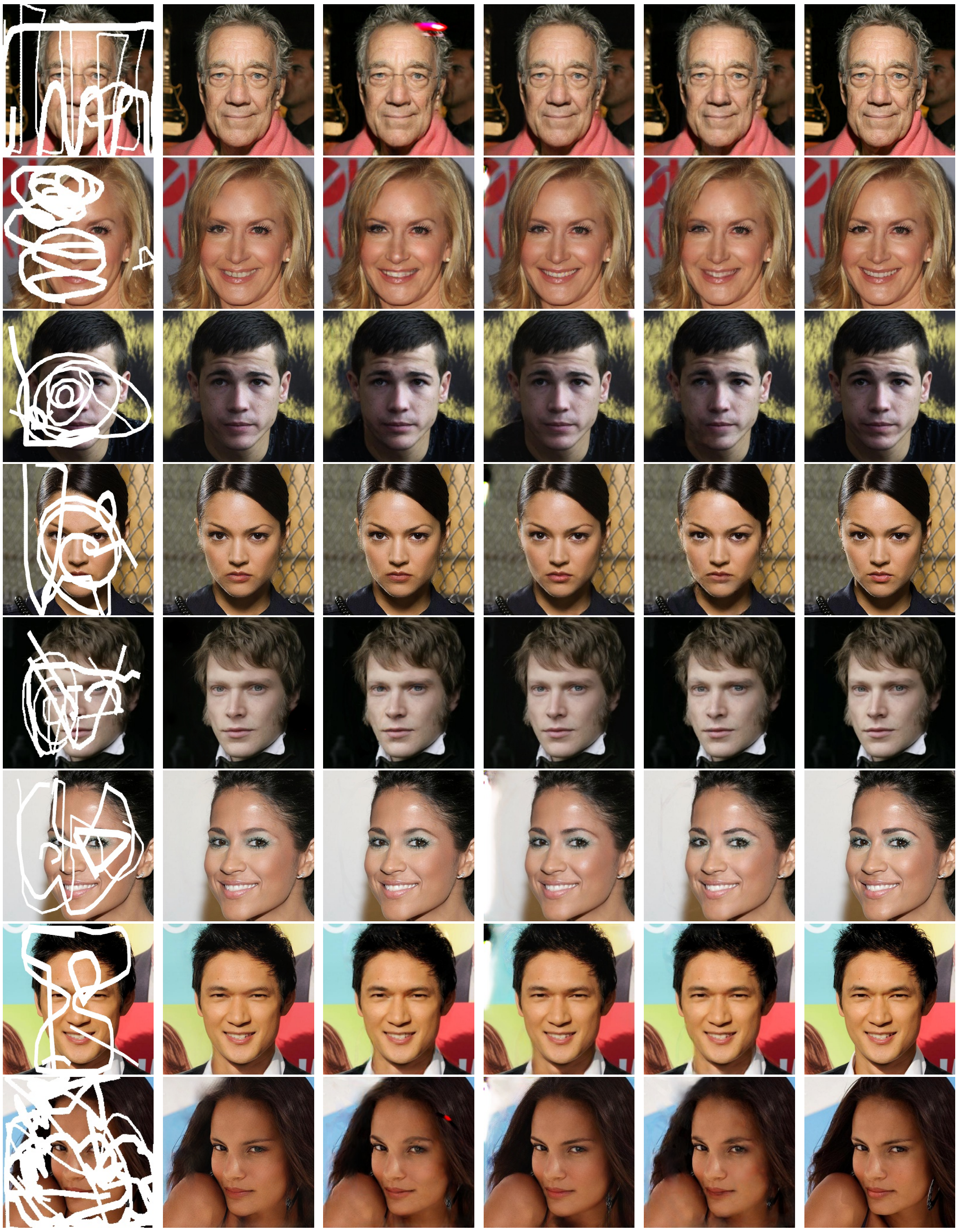
Chapter 8

Conclusion

This chapter summarises the research findings and the outcomes of the thesis. It provides new insights for future work and concluding remarks on the direction of facial inpainting.

8.1 Introduction

This thesis proposed new methods to image inpainting based on the fundamental components of GANs. The thesis aims to design novel facial inpainting algorithms by capturing contextualised features to fill in missing regions with irregular holes or free-form mask. This thesis has presented **SWGAN** in Chapter 4, **RMNet** in Chapter 5, **FGAN** in Chapter 6 and **V-LinkNet** in Chapter 7 as proposed solutions to facial inpainting. Two of these models have been extended to natural scene images to generalise the proposed solutions and to show that their performance are not only limited to facial inpainting. The proposed methods are considered as effective solutions for predictions of corrupted facial images at this point in time. Furthermore, when the proposed approaches are merged, these can successfully generate plausible results with both square and irregular masks. This is achieved by the dual-encoder network proposed in Chapter 6. The contributions of this thesis are summarised in the following parts, and the limitations are highlighted in order to provide possibility for advancement in this field of research.



(a) Input (b) **CH 4** (c) **CH 5** (d) **CH 6** (e) **CH 7** (f) GT

Figure 8.1: Visual summary of the performances by all the models proposed in this thesis based on the standardised testing protocol: (a) **Input masked-image**; (b) Chapter 4 (**CH4**); (c) Chapter 5 (**CH5**); (d) Chapter 6 (**CH6**); (e)Chapter 7 (**CH7**); and (f) Ground-truth image. (Zoom to see changes.)

8.2 Comparative Performance for all Methods

Quantitative and Qualitative evaluation is good practice in computer vision. This helps to improve on existing methods, thus propelling research in the right trajectory. Because this is important, a comparison of all the methods proposed in this thesis will help the reader to understand that the visual differences and quantitative differences are influenced by the different components of the proposed algorithms. Figure 8.1 shows the visual comparison on the various methods. The evaluations are conducted using the standardised testing protocol and a summary is shown on Table 8.1 for quantitative measures across all algorithms proposed in this thesis. Many newly proposed methods [238, 229, 11, 221] including the contributions in this thesis show how important a contributor is to a network. A slight change can improve or mar the results. The proposed method in Chapter 4 performed well to capture contextual information but left some checker-board artefacts in some images with difficult masks. This is because, it focused on the global image and no module was included in the model to capture allow it focus on the hole regions. In addition, computational resources was a problem as it faced memory issues. This limitation was taken into consideration and a new design introduced in Chapter 5. The model in Chapter 5 significant changes with reduced layers and a reverse mask mechanism and spatial preserving operator introduced. Nonetheless, this model in Chapter 5 still failed to capture key subtle features and faced some difficulty with difficult mask unlike the state of the art models [152, 124, 227]. The issue here is that the limitations are different as can been seen on the mask residue left on the third image of the first row on Figure 8.1 when compared to the state-of-the-art [152, 137] presented in Chapter 4. The model in Chapter 6 performed well in preserving the key subtle features and natural attributes of the face but failed to preserve background textures. The method in Chapter 7 performed well overall compared to all other methods proposed in this

Table 8.1: Quantitative comparison of various performance assessment metrics on 3,000 set 1 test images from the standardized protocol dataset. † Lower is better. ⊎ Higher is better.

Performance Assessment					
Method	Chapter	MAE †	FID†	PSNR ⊎	SSIM ⊎
SWGAN	Chapter 4	66.09	4.14	29.87	0.94
RMNet	Chapter 5	**31.91**	3.09	**40.40**	0.94
FGAN	Chapter 6	57.38	9.63	34.35	0.92
V-LinkNet	Chapter 7	37.97	**2.76**	39.75	**0.96**

thesis, thus robust to feature extraction, propagation and information dissemination. However, unlike the state of the art methods, it is limited in generated plausible features for images with large missing regions. Another observation is that the image has so much details and lighting effect on the forehead of the facial region can influence the results for facial inpainting as shown in Chapter 7. This is reinforced by a detailed examination of the images generated by the proposed model in Chapter 6, which failed in cases when the backdrop is too bright, while displaying high-quality inpainting inside the facial region. This demonstrates the need of considering all aspects of inpainting (feature extraction, feature propagation, feature dissemination, and feature regularisation) while developing and modelling. As a result, depending on the algorithm, the limitation would vary depending on the design and training duration. It is necessary to analyse the balance of computing resources. Because of computational resource constraints, the algorithms proposed in this thesis all have decreased layers at some point owing to memory constraints.

8.3 Research Findings

This section outlines the research findings of this thesis with respect to the objectives described in Chapter 1, section 1.6. Table 8.2 is a summary of the objectives of this work and their corresponding outcome. The first objective explored the literature on image inpainting with focus on irregular hole inpainting on facial images. The objective is achieved with the proposed guidelines to the research community in image inpainting in the form of a literature review paper published here [PUB1]. So far, the research conducted in Chapter 2 has served as guidance to design the algorithms that have met the objectives of this thesis, as evidenced by the publishing of the novel proposed methods that were developed from the research. These publications ([PUB2], [PUB3], [PUB4] and Chapter 7 form the contribution chapters to this thesis.

Section 4.3 is a facial inpainting method that uses a new combination loss function that constraints colour saturation within feature loss. The model can inpaint high resolution images thanks to implementation of dilated convolutions combined with skip connections to facilitate the process. This work lays the foundation to the contribution in Chapter 6.

In Section 5.3, a novel method is proposed. This method introduces reverse masking to image inpainting. The reverse masking methods combined with a newly proposed reverse mask loss assists the model during training. This work lays the ground work to the contribution in Chapter 6. The second objective is to use the proposed

Table 8.2: Objectives and outcomes of this thesis.

No	Objective(s)	Outcome(s)
1	To conduct an informed study on the state-of-the-art algorithms of image inpainting, nested with comparative study of these techniques on facial images. Identify the gap in facial inpainting and suggest future direction based on research findings.	Identified the gaps in literature and publish a research paper ([PUB1]) that highlights these gaps with future works to halt the proliferation of irresolute baseline models that lack contextualised features.
2	To develop new deep learning architectures for image inpainting based on Generative adversarial networks (GANs)	Three novel methods ([PUB2], [PUB3], [PUB4] are proposed and published. Furthermore, a new technique for image inpainting is proposed in Chapter 7.
3	To propose new approaches to facial inpainting with the capability of occlusion reasoning that can preserve fidelity of attributes even with large hole-to-image ratios.	A new technique with semantic segmentation masks is proposed in [PUB4]
4	To identify potential biases in performance evaluation and conduct empirical studies to compare the performance of the inpainting algorithms.	Introduced a standardised protocol for facial inpainting with paired irregular masks sizes with different variations. Chapter 7

methods in the contribution chapters to design an application for facial inpainting. A prototype (see Appendix B) to demonstrate this objective has been developed using Pythons QtCore module. The module includes a platform of independent implementations for animations, state machines, etc that can be used to design user applications. The objective is to showcase our work with an interactive tool using image inpainting where all the weights of our pretrained models can be explored to compare the different models in real time.

The third objective is to focus on the facial features on the image to preserve fidelity of attributes. This is achieved by using the segmented region of the facial skin within the model. It was observed that, this method extracts key subtle features to generate key attributes of the facial image with close similarity to the ground-truth regardless of the missing region (see Chapter 6). To show that this method performed well, the foreground (face only) regions of the inpainted image are segmented and evaluated using the benchmark metrics and evaluation techniques from baseline models. The evaluation as shown in Chapter 6 demonstrates that the quantitative measures of the proposed method outperforms the state of the art.

The fourth objective is to propose a standardised protocol to identity potential biases in the evaluation of generated images with respect to the masking method used. This objective is detailed in Chapter 7. This is achieve by curating from Nvidia irregular mask dataset which have masks of various sizes ranging from [0.01,0.8] and matching these against particular images from the CelebA-HQ test set based on the state of the art [137, 205, 120, 226, 227]. This is of use in that, it will strengthen future methods to evaluate the same facial features based on the same corrupted regions, thus improving on inpainting. This has set a standard to facial inpainting because the face is complex and if inpainted poorly can be visibly identified with the human eye.

This work answered the research questions in Chapter 1. The first question is addressed by proposing the method in Chapter 4, which introduced network component (dilated convolutions and skip connections) to capture more contextual information to handle high-resolution images. Finally the results were evaluated and compared with the state-of-the-art methods and used in further research to answer the second and third research questions. Thus in further proposed methods (Chapters 6 and Chapter 7) dilated convolutions and skip connections are utilized. However, both of these networks have this in common but differ in other components, which address facial fidelity and irregular hole inpainting with large textural regions. In Chapter 5, dilated convolutions and skip connection is not utilized but a novel technique to target irregular holes whilst preserving the valid regions is proposed. Furthermore a comparison of all the methods is outlined in the introductory part of this chapter to showcase how the performance improved with research.

8.4 Future Work

The future of image inpainting holds great promise for the development of new learning-based algorithms capable of high-quality and high-level feature extraction for image understanding. The following are a few of the possible paths to take in order to advance this field.

- Inpainting irregularly damaged regions on an image is still a challenge. One of the most common applications for image completion is to delete the parts of a picture that the user does not wish to be shown. These photos captured with a smart phone or camera are often of extremely good quality (high-resolution) and shot in very complex environments. It will be more difficult for the general public to create rectangular-shaped masks that directly match the content they wish to remove from such images than it will be for them to make irregular-shaped masks that follow the curves of the region to be eliminated. As a result, future research will focus on developing additional models that can solve this challenging task while at the same time requiring less computational resources. An effort to overcome this issue is illustrated in Chapter 4 using SWGAN, which proved effective in outperforming the state of the art for high-resolution images.

However, it was shown that this model requires a longer training time and is limited in its capacity to solve the problem of irregular masks.

- The necessity of a lightweight solution to image inpainting is the future. To address the complexity in irregular missing pixel regions, the RMNet was proposed to target missing pixels only whilst preserving the visible ones. Furthermore, to tackle the inpainting problem in real world scenarios, a robust architecture of extremely deep neural networks that can function on mobile devices needs to be developed. Despite the fact that the RMNet was utilised to tackle this issue, it may be too heavy to operate on mobile devices, thus there is still potential for improvement. The structural adjustments will serve as a roadmap for this kind of improvement on the RMNet. That is a more reduced layers of the RMNet trained over a decreased number of epochs with a lower amount of computing resources. One other suggestion is to explore standard convolutions and make changes that could be robust to distinguishably extract features of valid pixels and hole regions. Also feature propagation or feature information dissemination to subsequent layers should be considered when designing such models with less parameter accretion.

- Generating images with contextualised features is still a challenge. An attempt to solve this problem is proposed in Chapter 6, where a foreground mask is introduced into the network to serve as a representation of disentangled pixels of attribute features of the face, thus invoking semantic reasoning to assist the convolutional layers to hallucinate pixels with fidelity preservation. However, this network faced some drawbacks; where it performed well in performing inpainting on facial features but poor performance is associated with background regions. Another future development is to target more on preserving the background regions. Thus a design to handle this will be to preserve the background during inpainting and guide the network to target subtle facial features as well as generating features with fidelity preservation.

- A further attempt was to ditch the foreground mask and propose a model that will highlight high-level facial features. The preceding algorithms laid the groundwork for the design of the V-LinkNet architecture in Chapter 7. Although this model can handle all mask types and high-resolution images, it needed a large amount of processing power, necessitating the design and implementation of a simpler and robust light-weight architecture. Another potential future approach is the collection of extremely big and diverse training datasets. A dataset of this nature should include a fairly even distribution of people from all three groups of human beings.

- Loss functions are still a major issue. This area for image inpainting is a future development that research should focus on the design of the loss function, which would be beneficial. The loss functions of supervised models are primarily concerned with comparing the outputs to the desired pictures. Using an average with different weights on the masked interior contents and the boundaries of the masked region, for example, may be used to enhance the loss functions of the supervised models, according to one idea. The blurriness problem, which is linked with masks at border regions, might be solved by slightly overlapping the masked and unmasked areas and giving the overlapping boundaries higher weights in the loss. The loss functions of unsupervised models will be more difficult since these models will need to take into account different networks or latent distributions, which will make them more complicated. For example, the loss function of the GAN model necessitates the modification of the hyperparameter alpha in order to achieve better results. Another loss proposal is that when designing feature propagation within convolution layers, a suitable loss to compute the error within these layers should be considered; otherwise, it may take longer to achieve the desired results or there will still be failures in contextual information with large hole regions. When it comes to cost-effectiveness, combining the conventional approach with the deep learning method is another suggestion for the research community in image inpainting. Traditional techniques may fill in the gaps in the information depending on the surrounding context. Deep neural networks might be used to first rebuild some simple background information from the pictures or the borders of missing sections, and then forecast the most significant or most complicated component of the image using that information.

- Facial wrinkle inpainting. Future research, will focus on bounding box localisation targeting wrinkles in order to explore localised wrinkle segmentation. A newly proposed dataset that will facilitate this development will be needed for facial wrinkle inpainting. In addition, an expanded study on GAN dissection to explore neurons responsible for wrinkles, colourisation and other facial attributes may be useful. It will be great for novel techniques on image normalisation and other image preprocessing and post processing techniques to be proposed to aid visual computing.

- Empirical evaluation techniques that match the HVS are still under investigation. Experimentation on how individuals perceive image symmetry or cluttered images on generated faces vs how the same information is viewed on natural scene images is advised. This will provide hints on how convincing a generated image should seem based on subjective assessments before using quantitative measures.

Finally, it would be nice to see that researchers have embraced the proposed standardized protocol in Chapter 7 as a benchmark for all models going forward. Thus enabling perform evaluations with different evaluation metrics under the same condition (mask-to-image ratios). This will help to improve models in terms of optimization, hyperparameters and computational cost.

The research community has to work on a realism preservation evaluation metric that will assess if a generated image is preserved based on the human perception. At the moment, SSIM is the best measure so far, but a new technique will be much better.

8.5 Concluding Remarks

In conclusion, image inpainting derives from human visual perception to continue end nodes of missing pixels on damaged images as a traditional painting restoration method. Image inpainting has always been a difficult task, even among artists, due to differences in visual perception and understanding. To address these issues, digital image inpainting (Traditional and Deep Learning methods) were introduced based on previous research. Computational evaluation methods for validating inpainted images and quantifying contextualised features of inpainted regions in comparison to their original counterparts have been proposed in literature and have been explored in this thesis to evaluate the proposed algorithms for facial image inpainting. This thesis presents a collection of contributions that provide additional solutions to facial image inpainting. Techniques capable of preserving subtle textural features, facial expressions, and natural and unnatural facial features on high resolution images are included. One of the other two methods can force the network to focus on missing regions and generate images with contextualised features that are consistent with the rest of the image. The other is capable of highlighting and extracting high-level features while allowing for a smooth transition to the decoding layers. Furthermore, a standardised protocol as a testing dataset to identify disparities of generated images with different masks and the same mask with different facial images has been proposed and made available for reproducibility to allow research in facial image inpainting to progress in the direction of improving newly proposed techniques.

Appendix A

Network Configuration

In this section, the RMNet model 5 configuration is reported in detail. On Table A.1, shows the configuration of the Generator (G_θ). where M_r is the reversed mask and M_I is the masked image. On Table A.2 , the configuration of RMNet discriminator

Table A.1: RMNet Generator Architecture

Layer	Size/Dilation Rate	Output Size
Maskinput	None	$256 \times 256 \times 1$
Imageinput	None	$256 \times 256 \times 3$
lambda1 (M_r)	None	$256 \times 256 \times 1$
multiply1 (M_I)	None	$256 \times 256 \times 3$
conv2d5	5×5 /2	$256 \times 256 \times 64$
conv2d6	5×5 /2	$128 \times 128 \times 64$
conv2d7	5×5 /2	$64 \times 64 \times 128$
conv2d8	5×5 /2	$32 \times 32 \times 256$
conv2d9	5×5 /2	$16 \times 16 \times 512$
dropout1 (Dropout)	None	$16 \times 16 \times 512$
upsampling2d1	None	$32 \times 32 \times 512$
conv2dtranspose1	4×4 /2	$64 \times 64 \times 512$
upsampling2d2	None	$64 \times 64 \times 512$
conv2dtranspose2	4×4 /2	$64 \times 64 \times 256$
upsampling2d3	None	$128 \times 128 \times 256$
conv2dtranspose3	4×4 /2	$128 \times 128 \times 128$
upsampling2d4	None	$256 \times 256 \times 128$
conv2dtranspose4	4×4 /2	$256 \times 256 \times 64$
conv2dtranspose5	4×4 /2	$256 \times 256 \times 3$
activation5	None	$256 \times 256 \times 3$
multiply2	None	$256 \times 256 \times 3$
add1	None	$256 \times 256 \times 3$
concatenate1	None	$256 \times 256 \times 4$

(D_θ)is shown. All the sizes are included but not of the Flatten, Dense1 to Dense3 layer parameters as this will depend on the size of the input image. However, all sizes displayed on both tables are based on a 256 × 256 × 3 image.

Table A.2: RMNet Discriminator Architecture

Layer	Size/Stride	Output Size
conv2d1	3×3 /2	$128 \times 128 \times 64$
conv2d2	3×3 /2	$64 \times 64 \times 128$
conv2d3	3×3 /2	$32 \times 32 \times 256$
conv2d4	3×3 /2	$16 \times 16 \times 256$
flatten1	None	65536
dense1	None	512
dense2	None	256
dense3	None	1

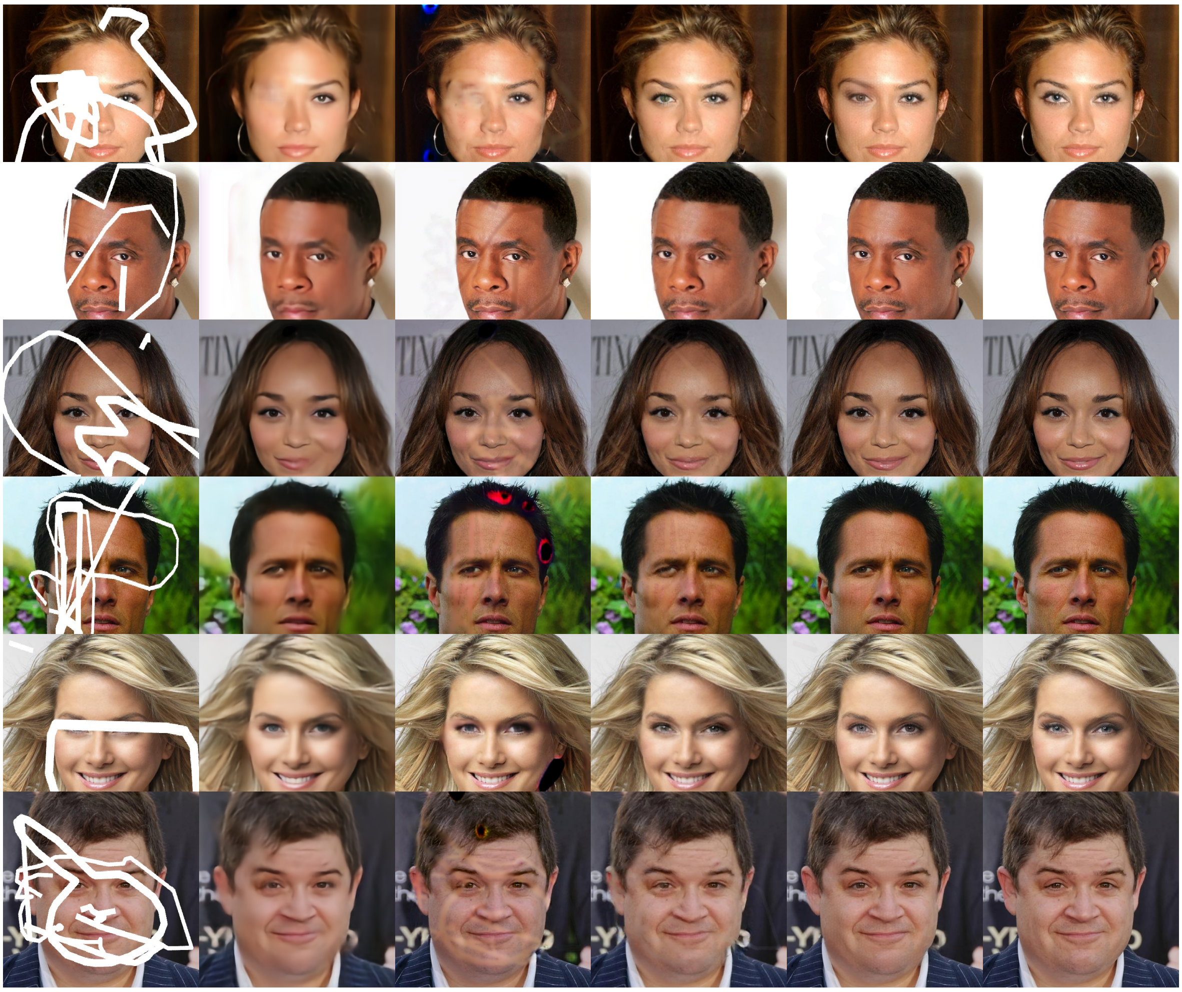
A.1 Qualitative Results

In this section, more qualitative results of RMNet on CelebA-HQ [124], Paris street view [45] and Places2 [241] datasets are shown. Figures A.1, A.2 and A.3 show results of RMNet compared with the state of the art methods [152] [124] [227]. On Figure A.4 more results of the model extended on Nvidia mask dataset are shown. Figures A.5 and A.6 show results on Paris street view [45] and Places2 [241]. In general, our model yield better results than the state-of-the-art when compared visually.

- **CE**: Context-Encoder by Pathak et al. [152]
- **PConv**: Partial Convolutions by Liu et al. [124]
- **GC**: Free-Form image inpainting with gated convolutions by Yu et al [227].
- **RMNet**: RMNet using $\boldsymbol{\ell}_{rm}$ when λ= **0.4**.

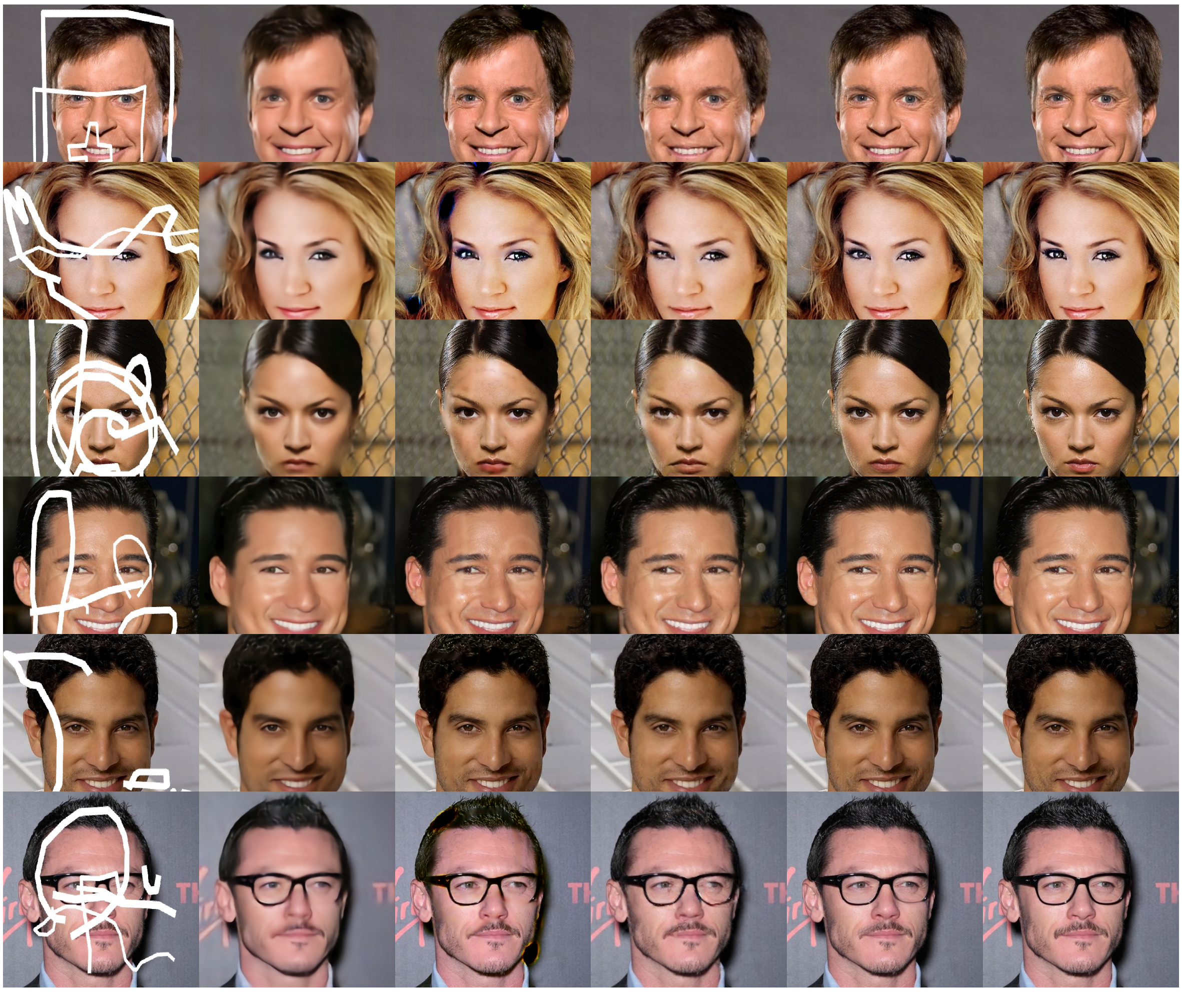
where $\boldsymbol{\ell}_{rm}$ is the reverse mask loss and λ is the weight applied on $\boldsymbol{\ell}_{rm}$.

On Figure A.3, some examples of failed cases by the proposed model compared to the state of the art are shown. However, our model was able to complete reasonable structure with some artefacts compared to the state of the art.



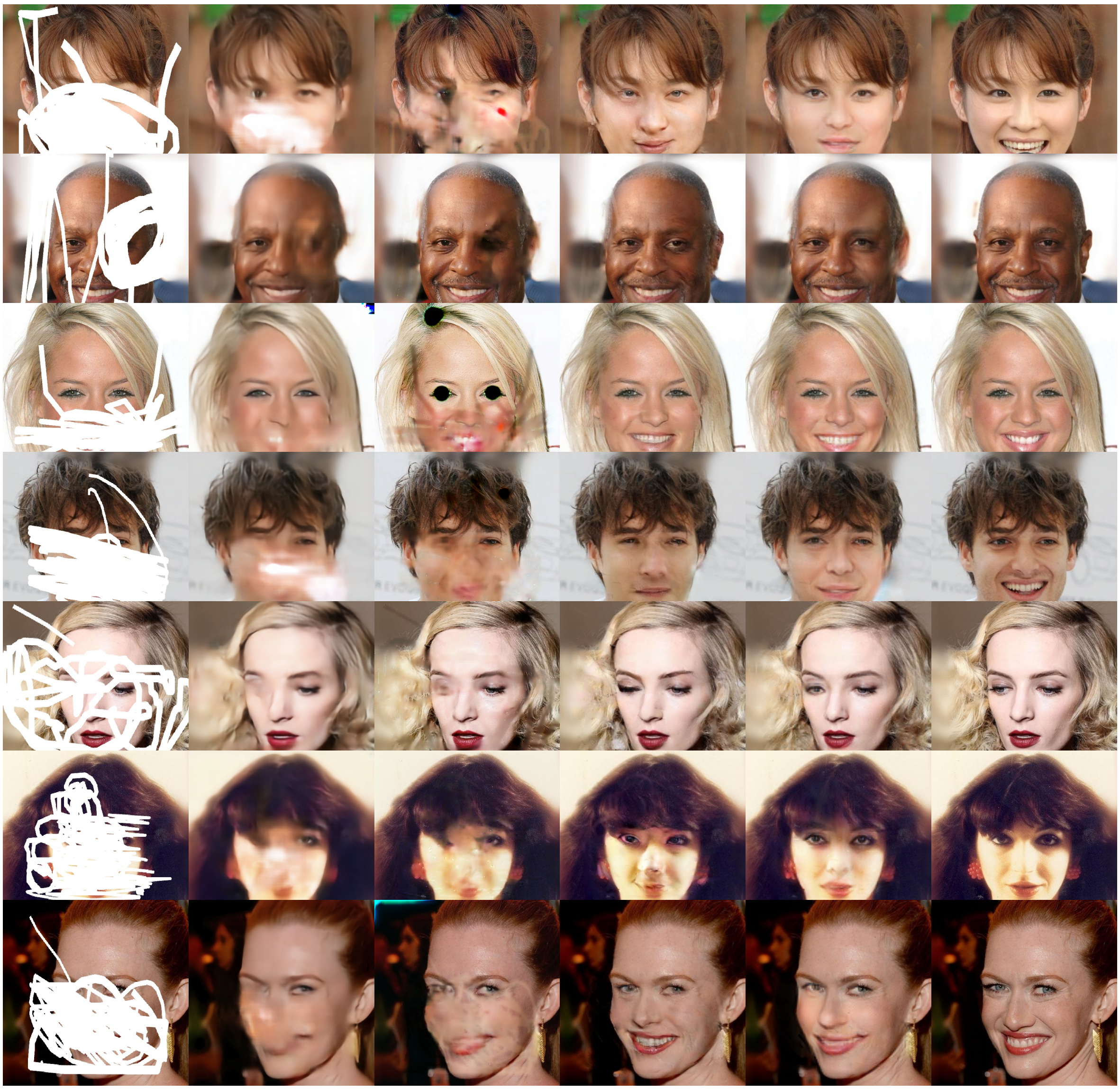
(a) **Masked** (b) **CE** (c) **PConv** (d) **GC** (e) **RMNet** (f) **GT**

Figure A.1: Examples of predictions by **CE**, **PConv** and **RMNet** on Quick Draw dataset[84] as masking method on CelebA-HQ [124].



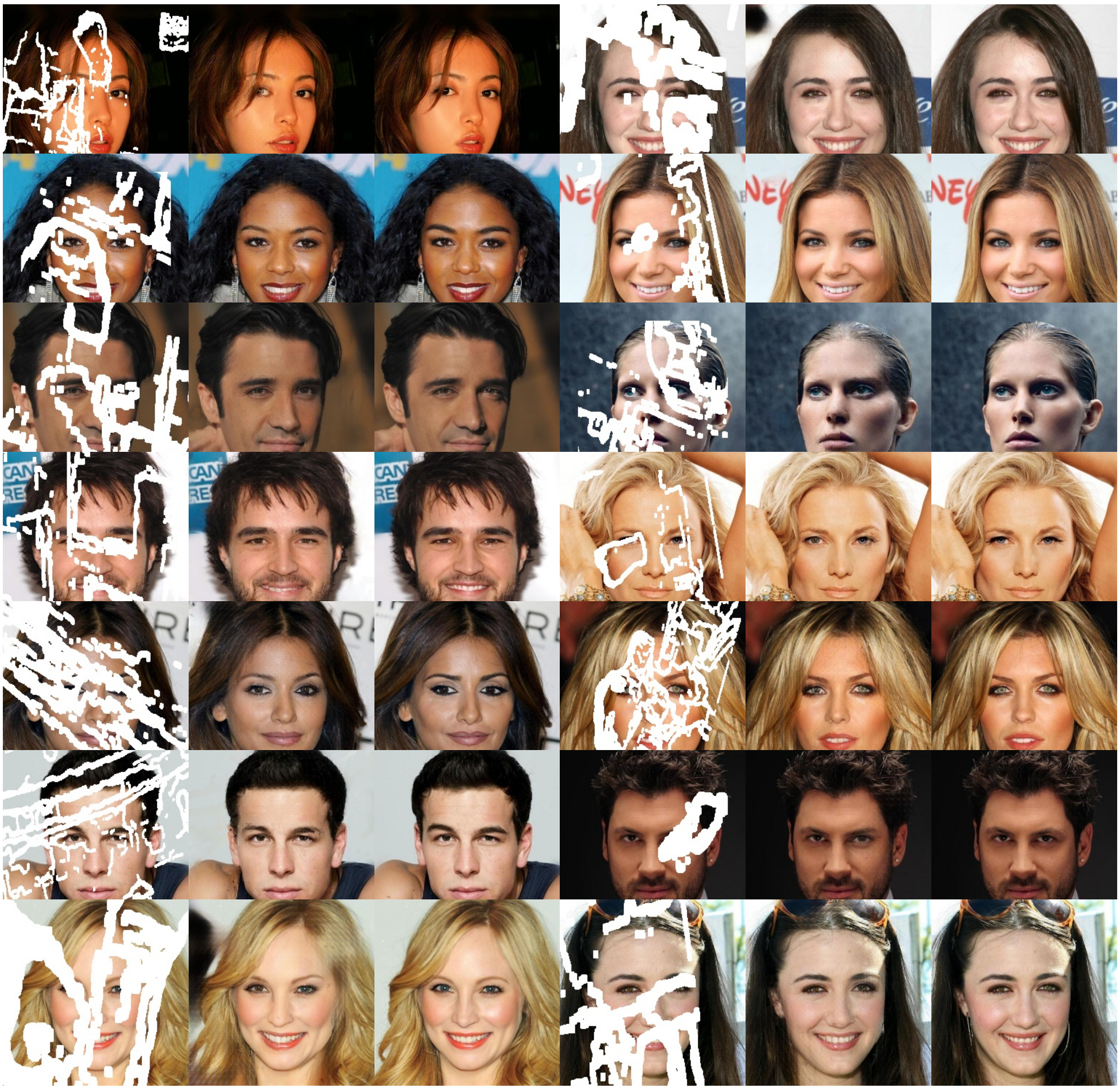
(a) **Masked** (b) **CE** (c) **PConv** (d) **GC** (e) **RMNet** (f) **GT**

Figure A.2: Examples of predictions by **CE**, **PConv** and **RMNet** on Quick Draw dataset[84] as masking method on CelebA-HQ [124].



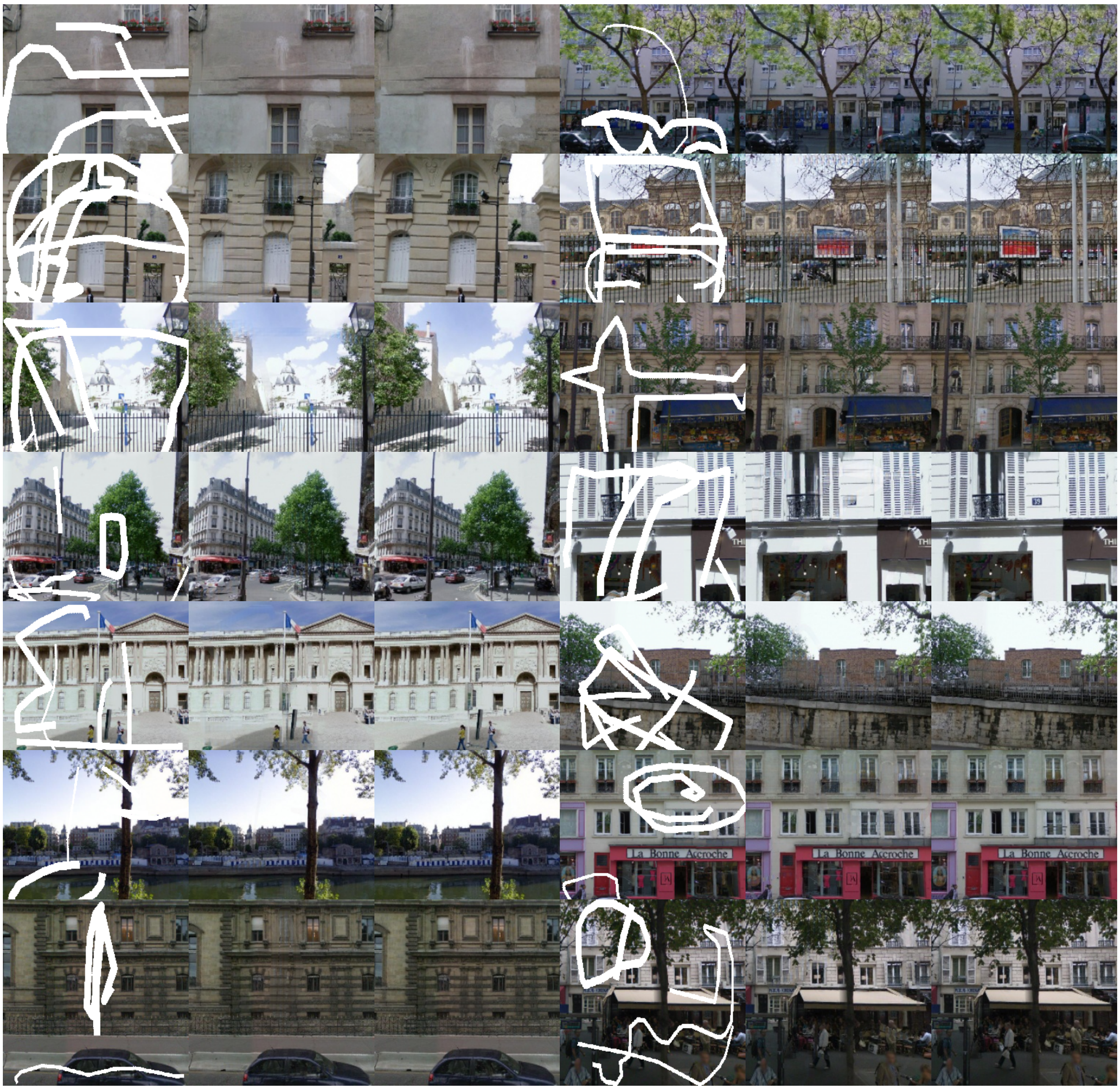
(a) **Masked** (b) **CE** (c) **PConv** (d) **GC** (e) **RMNet** (f) **GT**

Figure A.3: Failure cases by **CE**, **PConv**, **GC** and **RMNet** on Quick Draw dataset[84] as masking method on CelebA-HQ [124]. Visually, RMNET results are closer to ground-truth when compared to all other models.



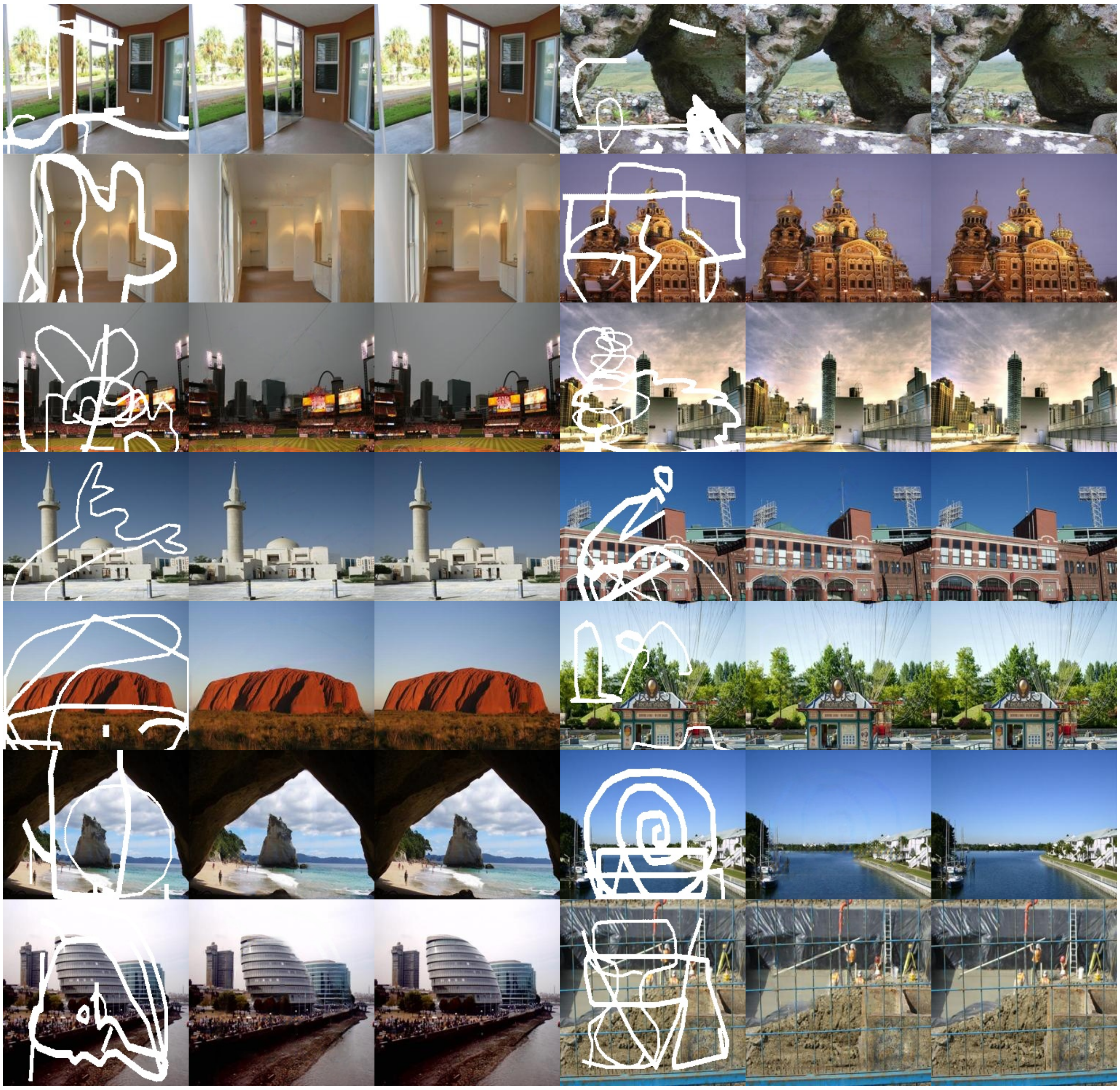
(a) Masked (b) **RMNet** (c) **GT** (d) Masked (e) **RMNet** (f) **GT**

Figure A.4: Examples of predictions using RMNet on Quick-Draw [84] Dataset with Nvidia Mask [124] used as masking method.



(a) Masked (b) **RMNet** (c) **GT** (d) Masked (e) **RMNet** (f) **GT**

Figure A.5: Examples of predictions using RMNet on Paris Street View [45] with Quick-Draw [84] used as masking method.



(a) Masked (b) **RMNet** (c) **GT** (d) Masked (e) **RMNet** (f) **GT**

Figure A.6: Examples of predictions using RMNet on Places [241] with Quick-Draw [84] used as masking method.

Appendix B

Application of Facial Inpainting

This section contains a prototype that will be used to demonstrate the inpainting algorithms proposed in this thesis. This is an interactive tool for facial inpainting in which users can apply hand-drawn masks to a facial image to evaluate the quality of inpainting by their preferred model. The inpainting results are evaluated using the pre-trained models from the proposed models in Chapter 4, Chapter 5, Chapter 6, Chapter 7. The application and pre-trained weights can be downloaded from GitHub.

B.1 Image Inpainting Prototype

Image inpainting is not all about research but also demonstrating that the research can be utilized in real world scenarios. A free-form inpainting prototype is created to demonstrate the capability of our models with real-world facial images. This application is designed as part of the final stages of the work to provide an interactive tool for facial inpainting. This prototype allows a user to draw a mask on the region they want to remove or inpaint and then input it. This prototype will work with the models created in Chapters 5 and 7. The figure's left side (Figure B.1 depicts the masked image. The user must upload the image using the buttons described on the prototype and then draw the mask on this image before inpainting with the middle button. The results are shown on the right side of Figure B.1. It takes about 1.93 seconds for smaller mask regions and up to a minute or two for very large mask regions, depending on the size of the mask. However, because the prototype was created in PyQt5, further development into a software App that can be deployed on a mobile device will be excellent. Furthermore, additional features such as brush size and other parameter selection would improve how a user interacts with the system. To test this prototype, it is required that Python 3.6- Python 3.8 is used due to some functions that have been removed in Python 3.9 which requires Tensorflow 2.5.

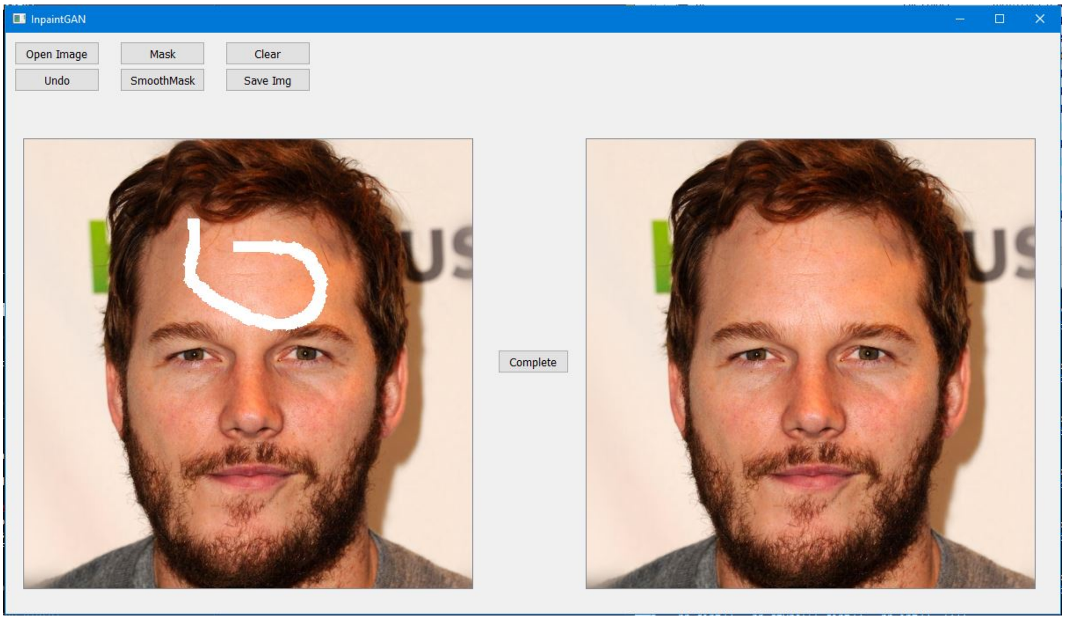


Figure B.1: Application for Facial Inpainting. An interactive tool designed to showcase the capability of our model to inpaint images online.

Bibliography

[1] Abdelghafour Abbad, Omar Elharrouss, Khalid Abbad, and Hamid Tairi. Application of meemd in post-processing of dimensionality reduction methods for face recognition. *Iet Biometrics*, 8(1):59–68, 2018.

[2] Nikolas Adaloglou. Intuitive explanation of skip connections in deep learning. *https://theaisummer.com/*, 2020.

[3] Forest Agostinelli, Matthew Hoffman, Peter Sadowski, and Pierre Baldi. Learning activation functions to improve deep neural networks. *arXiv preprint arXiv:1412.6830*, 2014.

[4] Adib Akl, Charles Yaacoub, Marc Donias, Jean-Pierre Da Costa, and Christian Germain. A survey of exemplar-based texture synthesis methods. *Computer Vision and Image Understanding*, 172:12–24, 2018.

[5] Cédric Allène and Nikos Paragios. Image renaissance using discrete optimization. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, volume 3, pages 631–634. IEEE, 2006.

[6] Brandon Amos. Image Completion with Deep Learning in TensorFlow. http://bamos.github.io/2016/08/09/deep-completion. Accessed: [Insert date here].

[7] Anelia Angelova, Yaser Abu-Mostafam, and Pietro Perona. Pruning training sets for learning of object categories. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 494–501. IEEE, 2005.

[8] Dragomir Anguelov, Carole Dulong, Daniel Filip, Christian Frueh, Stéphane Lafon, Richard Lyon, Abhijit Ogale, Luc Vincent, and Josh Weaver. Google street view: Capturing the world at street level. *Computer*, 43(6):32–38, 2010.

[9] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017.

[10] Michael Ashikhmin. Synthesizing natural textures. In *Proceedings of the 2001 symposium on Interactive 3D graphics*, pages 217–226. Citeseer, 2001.

[11] Dipali Vasant Atkale, Meenakshi Mukund Pawar, Shabdali Charudtta Deshpande, and Dhanashree Madhukar Yadav. Residual network for face progression and regression. In *Techno-Societal 2020*, pages 257–267. Springer, 2021.

[12] Ismail Avcıbaş, Bülent Sankur, and Khalid Sayood. Statistical evaluation of image quality measures. *Journal of Electronic imaging*, 11(2):206–223, 2002.

[13] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. Patchmatch: A randomized correspondence algorithm for structural image editing. *ACM Trans. Graph.*, 28(3):24, 2009.

[14] Nazre Batool and Rama Chellappa. Detection and inpainting of facial wrinkles using texture orientation fields and markov random field modeling. *IEEE transactions on image processing*, 23(9):3773–3788, 2014.

[15] Yoshua Bengio et al. Learning deep architectures for ai. *Foundations and trends® in Machine Learning*, 2(1):1–127, 2009.

[16] Marcelo Bertalmio, Andrea L Bertozzi, and Guillermo Sapiro. Navier-stokes, fluid dynamics, and image and video inpainting. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–I. IEEE, 2001.

[17] Marcelo Bertalmio, Guillermo Sapiro, Vincent Caselles, and Coloma Ballester. Image inpainting. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 417–424. ACM Press/Addison-Wesley Publishing Co., 2000.

[18] Marcelo Bertalmio, Luminita Vese, Guillermo Sapiro, and Stanley Osher. Simultaneous structure and texture image inpainting. *IEEE transactions on image processing*, 12(8):882–889, 2003.

[19] Andrea L Bertozzi, Selim Esedoglu, and Alan Gillette. Inpainting of binary images using the cahn–hilliard equation. *IEEE Transactions on image processing*, 16(1):285–291, 2006.

[20] Ali Borji. Pros and cons of gan evaluation measures. *Computer Vision and Image Understanding*, 179:41–65, 2019.

[21] Raphaël Bornard, Emmanuelle Lecan, Louis Laborelli, and Jean-Hugues Chenot. Missing data correction in still images and image sequences. In *Proceedings of the tenth ACM international conference on Multimedia*, pages 355–361, 2002.

[22] Jason Brownlee. A gentle introduction to generative adversarial networks (gans), Jul 2019.

[23] Antoni Buades, Bartomeu Coll, and J-M Morel. A non-local algorithm for image denoising. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 60–65. IEEE, 2005.

[24] Aurélie Bugeau, Marcelo Bertalmío, Vicent Caselles, and Guillermo Sapiro. A comprehensive framework for image inpainting. *IEEE Transactions on Image Processing*, 19(10):2634–2645, 2010.

[25] Pierre Buyssens, Maxime Daisy, David Tschumperlé, and Olivier Lézoray. Exemplar-based inpainting: Technical review and new heuristics for better geometric reconstructions. *IEEE transactions on image processing*, 24(6):1809–1824, 2015.

[26] Frédéric Cao, Yann Gousseau, Simon Masnou, and Patrick Pérez. Geometrically guided exemplar-based inpainting. *SIAM Journal on Imaging Sciences*, 4(4):1143–1179, 2011.

[27] Punarjay Chakravarty, Praveen Narayanan, and Tom Roussel. Gen-slam: Generative modeling for monocular simultaneous localization and mapping. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 147–153. IEEE, 2019.

[28] Tony F Chan and Jianhong Jackie Shen. *Image processing and analysis: variational, PDE, wavelet, and stochastic methods*, volume 94. Siam, 2005.

[29] Rong-Chi CHANG and Timothy K SHIH. Multilayer inpainting on digitalized artworks. *Journal of information science and engineering*, 24(4):1241–1255, 2008.

[30] Jun-xin Chen, Zhi-liang Zhu, Chong Fu, and Hai Yu. A fast image encryption scheme with a novel pixel swapping-based confusion approach. *Nonlinear Dynamics*, 77(4):1191–1207, 2014.

[31] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017.

[32] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

[33] Mingyi Chen, Changchun Li, Ke Li, Han Zhang, and Xuanji He. Double encoder conditional gan for facial expression synthesis. In *2018 37th Chinese Control Conference (CCC)*, pages 9286–9291. IEEE, 2018.

[34] Qiang Chen, Philippe Montesinos, Quan Sen Sun, Peng Ann Heng, et al. Adaptive total variation denoising based on difference curvature. *Image and vision computing*, 28(3):298–306, 2010.

[35] Yanxiang Chen, Guang Wu, Jie Zhou, and Guojun Qi. Image generation via latent space learning using improved combination. *Neurocomputing*, 340:8–18, 2019.

[36] Taeg Sang Cho, Moshe Butman, Shai Avidan, and William T Freeman. The patch transform and its applications to image editing. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[37] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8188–8197, 2020.

[38] Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 539–546. IEEE, 2005.

[39] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*, 2015.

[40] Antonio Criminisi, Patrick Pérez, and Kentaro Toyama. Region filling and object removal by exemplar-based image inpainting. *IEEE Transactions on image processing*, 13(9):1200–1212, 2004.

[41] Soheil Darabi, Eli Shechtman, Connelly Barnes, Dan B Goldman, and Pradeep Sen. Image melding: Combining inconsistent images using patch-based synthesis. *ACM Trans. Graph.*, 31(4):82–1, 2012.

[42] Ismaël Daribo and Béatrice Pesquet-Popescu. Depth-aided image inpainting for novel view synthesis. In *Multimedia Signal Processing (MMSP), 2010 IEEE International Workshop on*, pages 167–170. IEEE, 2010.

[43] Li Deng. The mnist database of handwritten digit images for machine learning research [best of the web]. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

[44] James J DiCarlo, Davide Zoccolan, and Nicole C Rust. How does the brain solve visual object recognition? *Neuron*, 73(3):415–434, 2012.

[45] Carl Doersch, Saurabh Singh, Abhinav Gupta, Josef Sivic, and Alexei Efros. What makes paris look like paris? *ACM Transactions on Graphics*, 31(4), 2012.

[46] Iddo Drori, Daniel Cohen-Or, and Hezy Yeshurun. Fragment-based image completion. In *ACM Transactions on graphics (TOG)*, volume 22, pages 303–312. ACM, 2003.

[47] Michal Drozdzal, Eugene Vorontsov, Gabriel Chartrand, Samuel Kadoury, and Chris Pal. The importance of skip connections in biomedical image segmentation. In *Deep learning and data labeling for medical applications*, pages 179–187. Springer, 2016.

[48] Vincent Dumoulin and Francesco Visin. A guide to convolution arithmetic for deep learning. *arXiv preprint arXiv:1603.07285*, 2016.

[49] Alexei A Efros and William T Freeman. Image quilting for texture synthesis and transfer. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 341–346. ACM, 2001.

[50] Alexei A Efros and Thomas K Leung. Texture synthesis by non-parametric sampling. In *iccv*, page 1033. IEEE, 1999.

[51] Omar Elharrouss, Noor Al-Maadeed, and Somaya Al-Maadeed. Video summarization based on motion detection for surveillance systems. In *2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC)*, pages 366–371. IEEE, 2019.

[52] Omar Elharrouss, Noor Almaadeed, Somaya Al-Maadeed, and Younes Akbari. Image inpainting: A review. *Neural Processing Letters*, pages 1–22, 2019.

[53] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

[54] Olivier Faugeras and Renaud Keriven. *Variational principles, surface evolution, PDE's, level set methods and the stereo problem*. IEEE, 2002.

[55] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*, 2015.

[56] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Texture synthesis and the controlled generation of natural stimuli using convolutional neural networks. *arXiv preprint arXiv:1505.07376*, 12, 2015.

[57] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2414–2423, 2016.

[58] Mrinmoy Ghorai, Sekhar Mandal, and Bhabatosh Chanda. A group-based image inpainting using patch refinement in mrf framework. *IEEE Transactions on Image Processing*, 27(2):556–567, 2018.

[59] D Goldman, E Shechtman, C Barnes, I Belaunde, and J Chien. Content-aware fill. *Accessed on*, 25, 2014.

[60] Jonas Gomes, Lucia Darsa, Bruno Costa, and Luiz Velho. *Warping & morphing of graphical objects*. Morgan Kaufmann, 1999.

[61] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

[62] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[63] Fiona Govan. Elderly woman who botched religious fresco demands royalties. *The Telegraph*, Sep 2012.

[64] Pulkit Goyal, Sapan Diwakar, et al. Fast and enhanced algorithm for exemplar based image inpainting. In *Image and Video Technology (PSIVT), 2010 Fourth Pacific-Rim Symposium on*, pages 325–330. IEEE, 2010.

[65] Petr Gronat, Michal Havlena, Josef Sivic, and Tomas Pajdla. Building streetview datasets for place recognition and city reconstruction. *Research Reports of CMP, Czech Technical University in Prague*, 2011.

[66] Christine Guillemot and Olivier Le Meur. Image inpainting: Overview and recent advances. *IEEE signal processing magazine*, 31(1):127–144, 2014.

[67] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In *Advances in Neural Information Processing Systems*, pages 5767–5777, 2017.

[68] Zongyu Guo, Zhibo Chen, Tao Yu, Jiale Chen, and Sen Liu. Progressive image inpainting with full-resolution residual network. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 2496–2504, 2019.

[69] Christopher Haccius and Thorsten Herfet. Computer vision performance and image quality metrics: Areciprocal relation. *Computer Vision Performance and Image Quality Metrics-A Reciprocal Relation*, 1:27–37, 2017.

[70] James Hays and Alexei A Efros. Scene completion using millions of photographs. *ACM Transactions on Graphics (TOG)*, 26(3):4, 2007.

[71] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020.

[72] Kaiming He and Jian Sun. Image completion approaches using the statistics of similar patches. *IEEE transactions on pattern analysis and machine intelligence*, 36(12):2423–2435, 2014.

[73] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[74] Lei He, Yan Xing, Kangxiong Xia, and Jieqing Tan. An adaptive image inpainting method based on continued fractions interpolation. *Discrete Dynamics in Nature and Society*, 2018, 2018.

[75] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in neural information processing systems*, pages 6626–6637, 2017.

[76] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006.

[77] Alain Hore and Djemel Ziou. Image quality metrics: Psnr vs. ssim. In *2010 20th International Conference on Pattern Recognition*, pages 2366–2369. IEEE, 2010.

[78] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[79] Ying Huang, Maorui Wang, Ying Qian, Shuohao Lin, and Xiaohan Yang. Image completion based on gans with a new loss function. In *Journal of Physics: Conference Series*, volume 1229, page 012030. IOP Publishing, 2019.

[80] Yu-Kai Huang, Tsung-Han Wu, Yueh-Cheng Liu, and Winston H Hsu. Indoor depth completion with boundary consistency and self-attention. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[81] Forrest N Iandola, Song Han, Matthew W Moskewicz, Khalid Ashraf, William J Dally, and Kurt Keutzer. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and¡ 0.5 mb model size. *arXiv preprint arXiv:1602.07360*, 2016.

[82] Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. Globally and locally consistent image completion. *ACM Transactions on Graphics (TOG)*, 36(4):107, 2017.

[83] Akiko Ikkai, Trenton A Jerde, and Clayton E Curtis. Perception and action selection dissociate human ventral and dorsal cortex. *Journal of cognitive neuroscience*, 23(6):1494–1506, 2011.

[84] Karim Iskakov. Semi-parametric image inpainting. *arXiv preprint arXiv:1807.02855*, 2018.

[85] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[86] Viren Jain and Sebastian Seung. Natural image denoising with convolutional networks. In *Advances in Neural Information Processing Systems*, pages 769–776, 2009.

[87] Jireh Jam, Connah Kendrick, Vincent Drouard, Kevin Walker, Gee-Sern Hsu, and Moi Hoon Yap. R-mnet: A perceptual adversarial network for image inpainting. *arXiv preprint arXiv:2008.04621*, 2020.

[88] Jireh Jam, Connah Kendrick, Vincent Drouard, Kevin Walker, Gee-Sern Hsu, and Moi Hoon Yap. Symmetric skip connection wasserstein gan for high-resolution facial image inpainting. *arXiv preprint arXiv:2001.03725*, 2020.

[89] Jireh Jam, Connah Kendrick, Vincent Drouard, Kevin Walker, Gee-Sern Hsu, and Moi Hoon Yap. R-mnet: A perceptual adversarial network for image inpainting. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2714–2723, 2021.

[90] Jireh Jam, Connah Kendrick, Vincent Drouard, Kevin Walker, and Moi Hoon Yap. Foreground-guided facial inpainting with fidelity preservation. In Nicolas Tsapatsoulis, Andreas Panayides, Theo Theocharides, Andreas Lanitis, Constantinos Pattichis, and Mario Vento, editors, *Computer Analysis of Images and Patterns*, pages 231–241, Cham, 2021. Springer International Publishing.

[91] Jireh Jam, Connah Kendrick, Kevin Walker, Vincent Drouard, Jison Gee-Sern Hsu, and Moi Hoon Yap. A comprehensive review of past and present image

inpainting methods. *Computer Vision and Image Understanding*, page 103147, 2020.

[92] Jiaya Jia and Chi-Keung Tang. Image repairing: Robust image synthesis by adaptive nd tensor voting. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 1, pages I–I. IEEE, 2003.

[93] Xiao Jin, Yuting Su, Liang Zou, Yongwei Wang, Peiguang Jing, and Z Jane Wang. Sparsity-based image inpainting detection via canonical correlation analysis with low-rank constraints. *IEEE Access*, 6:49967–49978, 2018.

[94] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016.

[95] Alexia Jolicoeur-Martineau. The relativistic discriminator: a key element missing from standard gan. *arXiv preprint arXiv:1807.00734*, 2018.

[96] Nick Kanopoulos, Nagesh Vasanthavada, and Robert L Baker. Design of an image edge detection filter using the sobel operator. *IEEE Journal of solid-state circuits*, 23(2):358–367, 1988.

[97] Bekir Karlik and A Vehbi Olgac. Performance analysis of various activation functions in generalized mlp architectures of neural networks. *International Journal of Artificial Intelligence and Expert Systems*, 1(4):111–122, 2011.

[98] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

[99] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *International Conference on Learning Representations*, 2018.

[100] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4401–4410, 2019.

[101] Norihiko Kawai, Tomokazu Sato, and Naokazu Yokoya. Image inpainting considering brightness change and spatial locality of textures. In *VISAPP (1)*, pages 66–73. Citeseer, 2008.

[102] Vahid Kazemi and Josephine Sullivan. One millisecond face alignment with an ensemble of regression trees. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1867–1874, 2014.

[103] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[104] Diederik P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-supervised learning with deep generative models. In *Advances in neural information processing systems*, pages 3581–3589, 2014.

[105] Durk P Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling. Improved variational inference with inverse autoregressive flow. *Advances in neural information processing systems*, 29:4743–4751, 2016.

[106] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[107] Reese Kuppig. Image quilting for texture synthesis and transfer,http://cs.brown.edu/courses/cs129/results/proj4/rkuppig/. *CS129-CS-Brown Edu*, 2015.

[108] Tsz-Ho Kwok, Hoi Sheung, and Charlie CL Wang. Fast query for exemplar-based image completion. *IEEE Transactions on Image Processing*, 19(12):3106–3115, 2010.

[109] Tsz-Ho Kwok and Charlie CL Wang. Interactive image inpainting using dct based exemplar matching. In *International Symposium on Visual Computing*, pages 709–718. Springer, 2009.

[110] Y-K Lai, S-M Hu, DX Gu, and Ralph R Martin. Geometric texture synthesis and transfer via geometry images. In *Proceedings of the 2005 ACM symposium on Solid and physical modeling*, pages 15–26, 2005.

[111] Olivier Le Meur, Josselin Gautier, and Christine Guillemot. Examplar-based inpainting based on local geometry. In *Image Processing (ICIP), 2011 18th IEEE International Conference on*, pages 3401–3404. IEEE, 2011.

[112] Olivier Le Meur and Christine Guillemot. Super-resolution-based inpainting. In *European Conference on Computer Vision*, pages 554–567. Springer, 2012.

[113] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[114] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017.

[115] Cheng-Han Lee, Ziwei Liu, Lingyun Wu, and Ping Luo. Maskgan: Towards diverse and interactive facial image manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5549–5558, 2020.

[116] Chuan Li and Michael Wand. Combining markov random fields and convolutional neural networks for image synthesis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2479–2486, 2016.

[117] Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets. *arXiv preprint arXiv:1712.09913*, 2017.

[118] Haodong Li and Jiwu Huang. Localization of deep inpainting using high-pass fully convolutional network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8301–8310, 2019.

[119] Haodong Li, Weiqi Luo, and Jiwu Huang. Localization of diffusion-based inpainting in digital images. *IEEE Transactions on Information Forensics and Security*, 12(12):3050–3064, 2017.

[120] Jingyuan Li, Fengxiang He, Lefei Zhang, Bo Du, and Dacheng Tao. Progressive reconstruction of visual structure for image inpainting. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5962–5971, 2019.

[121] Jingyuan Li, Ning Wang, Lefei Zhang, Bo Du, and Dacheng Tao. Recurrent feature reasoning for image inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7760–7768, 2020.

[122] Yijun Li, Sifei Liu, Jimei Yang, and Ming-Hsuan Yang. Generative face completion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3911–3919, 2017.

[123] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid. Refinenet: Multipath refinement networks for high-resolution semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1925–1934, 2017.

[124] Guilin Liu, Fitsum A Reda, Kevin J Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. Image inpainting for irregular holes using partial convolutions. *arXiv preprint arXiv:1804.07723*, 2018.

[125] Hongyu Liu, Bin Jiang, Yibing Song, Wei Huang, and Chao Yang. Rethinking image inpainting via a mutual encoder-decoder with feature equalizations. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 725–741. Springer, 2020.

[126] Hongyu Liu, Bin Jiang, Yi Xiao, and Chao Yang. Coherent semantic attention for image inpainting. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4170–4179, 2019.

[127] Huaming Liu, Xuehui Bi, Guanming Lu, and Weilan Wang. Exemplar-based image inpainting with multi-resolution information and the graph cut technique. *IEEE Access*, 7:101641–101657, 2019.

[128] Shaoguo Liu, Ying Wang, Jue Wang, Haibo Wang, Jixia Zhang, and Chunhong Pan. Kinect depth restoration via energy minimization with tv 21 regularization. In *Image Processing (ICIP), 2013 20th IEEE International Conference on*, pages 724–724. IEEE, 2013.

[129] Xinhua Liu, Yao Zou, Chengjuan Xie, Hailan Kuang, and Xiaolin Ma. Bidirectional face aging synthesis based on improved deep convolutional generative adversarial networks. *Information*, 10(2):69, 2019.

[130] Ziwei Liu, Xiaoxiao Li, Ping Luo, Chen-Change Loy, and Xiaoou Tang. Semantic image segmentation via deep parsing network. In *Proceedings of the IEEE international conference on computer vision*, pages 1377–1385, 2015.

[131] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages 3730–3738, 2015.

[132] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[133] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Large-scale celebfaces attributes (celeba) dataset. *Retrieved August*, 15:2018, 2018.

[134] Olivier Losson, Ludovic Macaire, and Yanqin Yang. Comparison of color demosaicing methods. In *Advances in Imaging and Electron Physics*, volume 162, pages 173–265. Elsevier, 2010.

[135] Shenlong Lou, Qiancong Fan, Feng Chen, Cheng Wang, and Jonathan Li. Preliminary investigation on single remote sensing image inpainting through a modified gan. In *2018 10th IAPR Workshop on Pattern Recognition in Remote Sensing (PRRS)*, pages 1–6. IEEE, 2018.

[136] Chenyang Lu and Gijs Dubbelman. Semantic foreground inpainting from weak supervision. *IEEE Robotics and Automation Letters*, 5(2):1334–1341, 2020.

[137] Yongyi Lu, Shangzhe Wu, Yu-Wing Tai, and Chi-Keung Tang. Image generation from sketch constraint using contextual gan. In *Proceedings of the European conference on computer vision (ECCV)*, pages 205–220, 2018.

[138] Mario Lucic, Karol Kurach, Marcin Michalski, Sylvain Gelly, and Olivier Bousquet. Are gans created equal? a large-scale study. *arXiv preprint arXiv:1711.10337*, 2017.

[139] Xin Ma, Xiaoqiang Zhou, Huaibo Huang, Zhenhua Chai, Xiaolin Wei, and Ran He. Free-form image inpainting via contrastive attention network. *arXiv preprint arXiv:2010.15643*, 2020.

[140] Frederik Maes, Andre Collignon, Dirk Vandermeulen, Guy Marchal, and Paul Suetens. Multimodality image registration by maximization of mutual information. *IEEE transactions on Medical Imaging*, 16(2):187–198, 1997.

[141] Julien Mairal, Michael Elad, and Guillermo Sapiro. Sparse representation for color image restoration. *IEEE Transactions on image processing*, 17(1):53–69, 2008.

[142] Xiao-Jiao Mao, Chunhua Shen, and Yu-Bin Yang. Image restoration using convolutional auto-encoders with symmetric skip connections. *arXiv preprint arXiv:1606.08921*, 2016.

[143] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[144] Jiangchun Mo and Yucai Zhou. The research of image inpainting algorithm using self-adaptive group structure and sparse representation. *Cluster Computing*, 22(3):7593–7601, 2019.

[145] Kamyar Nazeri, Eric Ng, Tony Joseph, Faisal Qureshi, and Mehran Ebrahimi. Edgeconnect: Structure guided image inpainting using edge prediction. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[146] James A Nichols, Hsien W Herbert Chan, and Matthew AB Baker. Machine learning: applications of artificial intelligence to imaging and diagnosis. *Biophysical reviews*, 11(1):111–118, 2019.

[147] Chigozie Nwankpa, Winifred Ijomah, Anthony Gachagan, and Stephen Marshall. Activation functions: Comparison of trends in practice and research for deep learning. *arXiv preprint arXiv:1811.03378*, 2018.

[148] Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier gans. In *International conference on machine learning*, pages 2642–2651. PMLR, 2017.

[149] Lidia Ogiela. Innovation approach to cognitive medical image interpretation. In *2008 International Conference on Innovations in Information Technology*, pages 722–726. IEEE, 2008.

[150] Nikos Paragios, Yunmei Chen, and Olivier D Faugeras. *Handbook of mathematical models in computer vision*. Springer Science & Business Media, 2006.

[151] Taesung Park, Jun-Yan Zhu, Oliver Wang, Jingwan Lu, Eli Shechtman, Alexei A Efros, and Richard Zhang. Swapping autoencoder for deep image manipulation. *arXiv preprint arXiv:2007.00653*, 2020.

[152] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2536–2544, 2016.

[153] Dabal Pedamonti. Comparison of non-linear activation functions for deep neural networks on mnist classification task. *arXiv preprint arXiv:1804.02763*, 2018.

[154] Patrick Pérez, Michel Gangnet, and Andrew Blake. Poisson image editing. *ACM Transactions on graphics (TOG)*, 22(3):313–318, 2003.

[155] Muhammad Ali Qureshi, Mohamed Deriche, Azeddine Beghdadi, and Asjad Amin. A critical survey of state-of-the-art image inpainting quality assessment metrics. *Journal of Visual Communication and Image Representation*, 49:177–191, 2017.

[156] Lara Raad and Bruno Galerne. Efros and freeman image quilting algorithm for texture synthesis. *Image Processing On Line*, 7:1–22, 2017.

[157] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[158] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative adversarial text to image synthesis. In *International Conference on Machine Learning*, pages 1060–1069. PMLR, 2016.

[159] Yurui Ren, Xiaoming Yu, Ruonan Zhang, Thomas H Li, Shan Liu, and Ge Li. Structureflow: Image inpainting via structure-aware appearance flow. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 181–190, 2019.

[160] Manuel M Oliveira Brian Bowen Richard and McKenna Yu-Sung Chang. Fast digital image inpainting. In *Appeared in the Proceedings of the International Conference on Visualization, Imaging and Image Processing (VIIP 2001), Marbella, Spain*, pages 106–107, 2001.

[161] Myung-Cheol Roh and Seong-Whan Lee. Performance analysis of face recognition algorithms on korean face database. *International Journal of Pattern Recognition and Artificial Intelligence*, 21(06):1017–1033, 2007.

[162] O Ronneberger, P Fischer, and TU-net Brox. Convolutional networks for biomedical image segmentation. In *Paper presented at: International Conference on Medical Image Computing and Computer-Assisted Intervention2015*, pages 234–241, 2015.

[163] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[164] A Rosebrock. Keras: Gans with keras and tensorflow. 2019.

[165] Adrian Rosebrock. *Deep Learning for Computer Vision with Python*. PyImageSearch.com, 2.1.0 edition, 2019.

[166] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.

[167] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[168] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[169] Tijana Ružić and Aleksandra Pižurica. Context-aware patch-based image inpainting using markov random field modeling. *IEEE transactions on image processing*, 24(1):444–456, 2014.

[170] Michał Sadowski and Aleksandra Grzegorczyk. Image inpainting with gradient attention. *Schedae Informaticae*, 27, 2018.

[171] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *Advances in neural information processing systems*, 29:2234–2242, 2016.

[172] Samuel Schulter, Menghua Zhai, Nathan Jacobs, and Manmohan Chandraker. Learning to look around objects for top-view representations of outdoor scenes. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 787–802, 2018.

[173] Sagar Sharma and Simone Sharma. Activation functions in neural networks. *Towards Data Science*, 6(12):310–316, 2017.

[174] Hamid R Sheikh and Alan C Bovik. Image information and visual quality. *IEEE Transactions on image processing*, 15(2):430–444, 2006.

[175] Bin Shen, Wei Hu, Yimin Zhang, and Yu-Jin Zhang. Image inpainting via sparse representation. In *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, pages 697–700. IEEE, 2009.

[176] Jianhong Shen and Tony F Chan. Mathematical models for local nontexture inpaintings. *SIAM Journal on Applied Mathematics*, 62(3):1019–1043, 2002.

[177] Jianhong Shen, Sung Ha Kang, and Tony F Chan. Euler's elastica and curvature-based inpainting. *SIAM journal on Applied Mathematics*, 63(2):564–592, 2003.

[178] Yujun Shen, Ping Luo, Junjie Yan, Xiaogang Wang, and Xiaoou Tang. Faceidgan: Learning a symmetry three-player gan for identity-preserving face synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 821–830, 2018.

[179] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1874–1883, 2016.

[180] Timothy K Shih and Rong-Chi Chang. Digital inpainting-survey and multilayer image inpainting algorithms. In *Information Technology and Applications, 2005. ICITA 2005. Third International Conference on*, volume 1, pages 15–24. IEEE, 2005.

[181] Timothy K Shih, Liang-Chen Lu, Ying-Hong Wang, and Rong-Chi Chang. Multi-resolution image inpainting. In *2003 International Conference on Multimedia and Expo. ICME'03. Proceedings (Cat. No. 03TH8698)*, volume 1, pages I–485. IEEE, 2003.

[182] Denis Simakov, Yaron Caspi, Eli Shechtman, and Michal Irani. Summarizing visual data using bidirectional similarity. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[183] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[184] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. In *Advances in neural information processing systems*, pages 3483–3491, 2015.

[185] Yuhang Song, Chao Yang, Yeji Shen, Peng Wang, Qin Huang, and C-C Jay Kuo. Spg-net: Segmentation prediction and guidance network for image inpainting. *arXiv preprint arXiv:1805.03356*, 2018.

[186] G Sridevi and S Srinivas Kumar. Image inpainting based on fractional-order nonlinear diffusion for image reconstruction. *Circuits, Systems, and Signal Processing*, 38(8):3802–3817, 2019.

[187] Jian Sun, Lu Yuan, Jiaya Jia, and Heung-Yeung Shum. Image completion with structure propagation. In *ACM Transactions on Graphics (ToG)*, volume 24, pages 861–868. ACM, 2005.

[188] Richard Szeliski, Heung-Yeung Shum, Heung-Yeung Shum, and Heung-Yeung Shum. Creating full view panoramic image mosaics and environment maps. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 251–258. ACM Press/Addison-Wesley Publishing Co., 1997.

[189] Zinovi Tauber, Ze-Nian Li, and Mark S Drew. Review and preview: Disocclusion by inpainting for image-based rendering. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 37(4):527–540, 2007.

[190] Alexandru Telea. An image inpainting technique based on the fast marching method. *Journal of graphics tools*, 9(1):23–34, 2004.

[191] Lucas Theis, Aäron van den Oord, and Matthias Bethge. A note on the evaluation of generative models. *arXiv preprint arXiv:1511.01844*, 2015.

[192] Isabel Thottam. The cost of conservation and restoration,http://artbusinessnews.com/2015/12/the-cost-of-conservation-and-restoration/. *Art Business News*, 2015.

[193] Ilya O Tolstikhin, Sylvain Gelly, Olivier Bousquet, Carl-Johann SIMON-GABRIEL, and Bernhard Schölkopf. Adagan: Boosting generative models. *Advances in Neural Information Processing Systems*, 30, 2017.

[194] David Tschumperlé. Fast anisotropic smoothing of multi-valued images using curvature-preserving pde's. *International Journal of Computer Vision*, 68(1):65–82, 2006.

[195] Radim Tyleček and Radim Šára. Spatial pattern templates for recognition of objects with regular structure. In *Proc. GCPR*, Saarbrucken, Germany, 2013.

[196] Dmitry Ulyanov, Vadim Lebedev, Andrea Vedaldi, and Victor S Lempitsky. Texture networks: Feed-forward synthesis of textures and stylized images. In *ICML*, pages 1349–1357, 2016.

[197] Huy V Vo, Ngoc QK Duong, and Patrick Perez. Structural inpainting. *arXiv preprint arXiv:1803.10348*, 2018.

[198] Haixia Wang, Li Jiang, Ronghua Liang, and Xiao-Xin Li. Exemplar-based image inpainting using structure consistent patch matching. *Neurocomputing*, 269:90–96, 2017.

[199] Miaohui Wang, Bo Yan, and Hamid Gharavi. Pyramid model based down-sampling for image inpainting. In *2010 IEEE International Conference on Image Processing*, pages 429–432. IEEE, 2010.

[200] Ning Wang, Jingyuan Li, Lefei Zhang, and Bo Du. Musical: Multi-scale image contextual attention learning for inpainting. In *IJCAI*, pages 3748–3754, 2019.

[201] Ning Wang, Sihan Ma, Jingyuan Li, Yipeng Zhang, and Lefei Zhang. Multistage attention network for image inpainting. *Pattern Recognition*, page 107448, 2020.

[202] Qiang Wang, Huijie Fan, Gan Sun, Yang Cong, and Yandong Tang. Laplacian pyramid adversarial network for face completion. *Pattern Recognition*, 88:493–505, 2019.

[203] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8798–8807, 2018.

[204] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European conference on computer vision (ECCV) workshops*, pages 0–0, 2018.

[205] Yi Wang, Xin Tao, Xiaojuan Qi, Xiaoyong Shen, and Jiaya Jia. Image inpainting via generative multi-column convolutional neural networks. In *Advances in neural information processing systems*, pages 331–340, 2018.

[206] Zhou Wang and Alan C Bovik. A universal image quality index. *IEEE signal processing letters*, 9(3):81–84, 2002.

[207] Zhou Wang, Alan C Bovik, Hamid R Sheikh, Eero P Simoncelli, et al. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[208] Zhou Wang, Eero P Simoncelli, and Alan C Bovik. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, volume 2, pages 1398–1402. Ieee, 2003.

[209] Li-Yi Wei and Marc Levoy. Fast texture synthesis using tree-structured vector quantization. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 479–488. ACM Press/Addison-Wesley Publishing Co., 2000.

[210] Jing Xiao, Liang Liao, Qiegen Liu, and Ruimin Hu. Cisi-net: Explicit latent content inference and imitated style rendering for image inpainting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 354–362, 2019.

[211] Chaohao Xie, Shaohui Liu, Chao Li, Ming-Ming Cheng, Wangmeng Zuo, Xiao Liu, Shilei Wen, and Errui Ding. Image inpainting with learnable bidirectional attention maps. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 8858–8867, 2019.

[212] Junyuan Xie, Linli Xu, and Enhong Chen. Image denoising and inpainting with deep neural networks. In *Advances in neural information processing systems*, pages 341–349, 2012.

[213] Saining Xie and Zhuowen Tu. Holistically-nested edge detection. In *Proceedings of the IEEE international conference on computer vision*, pages 1395–1403, 2015.

[214] Wei Xiong, Jiahui Yu, Zhe Lin, Jimei Yang, Xin Lu, Connelly Barnes, and Jiebo Luo. Foreground-aware image inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5840–5848, 2019.

[215] Zongben Xu and Jian Sun. Image inpainting by patch propagation using patch sparsity. *IEEE transactions on image processing*, 19(5):1153–1165, 2010.

[216] Zhaoyi Yan, Xiaoming Li, Mu Li, Wangmeng Zuo, and Shiguang Shan. Shift-net: Image inpainting via deep feature rearrangement. In *Proceedings of the European conference on computer vision (ECCV)*, pages 1–17, 2018.

[217] Chao Yang, Xin Lu, Zhe Lin, Eli Shechtman, Oliver Wang, and Hao Li. High-resolution image inpainting using multi-scale neural patch synthesis. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, page 3, 2017.

[218] Shuai Yang, Jiaying Liu, Sijie Song, Mading Li, and Zongming Quo. Structure-guided image completion via regularity statistics. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1711–1715. IEEE, 2016.

[219] Yang Yang, Xiaojie Guo, Jiayi Ma, Lin Ma, and Haibin Ling. Lafin: Generative landmark guided face inpainting. *arXiv preprint arXiv:1911.11394*, 2019.

[220] Yizhong Yang, Zhihang Cheng, Haotian Yu, Yongqiang Zhang, Xin Cheng, Zhang Zhang, and Guangjun Xie. Mse-net: generative image inpainting with multi-scale encoder. *The Visual Computer*, pages 1–13, 2021.

[221] Xu Yao, Gilles Puy, Alasdair Newson, Yann Gousseau, and Pierre Hellier. High resolution face age editing. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 8624–8631. IEEE, 2021.

[222] Moi Hoon Yap, Jhan S Alarifi, Choon-Ching Ng, Nazre Batool, and Kevin Walker. Automated facial wrinkles annotator. In *ECCV Workshops (4)*, pages 676–680, 2018.

[223] Raymond A Yeh, Chen Chen, Teck-Yian Lim, Alexander G Schwing, Mark Hasegawa-Johnson, and Minh N Do. Semantic image inpainting with deep generative models. In *CVPR*, volume 2, page 4, 2017.

[224] Zili Yi, Qiang Tang, Shekoofeh Azizi, Daesik Jang, and Zhan Xu. Contextual residual aggregation for ultra high-resolution image inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7508–7517, 2020.

[225] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015.

[226] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Generative image inpainting with contextual attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5505–5514, 2018.

[227] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Free-form image inpainting with gated convolution. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4471–4480, 2019.

[228] Yanhong Zeng, Jianlong Fu, Hongyang Chao, and Baining Guo. Learning pyramid-context encoder network for high-quality image inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1486–1494, 2019.

[229] Yanhong Zeng, Jianlong Fu, Hongyang Chao, and Baining Guo. Aggregated contextual transformations for high-resolution image inpainting. *arXiv preprint arXiv:2104.01431*, 2021.

[230] He Zhang, Vishwanath Sindagi, and Vishal M Patel. Image de-raining using a conditional generative adversarial network. *IEEE transactions on circuits and systems for video technology*, 30(11):3943–3956, 2019.

[231] Hongying Zhang and Shimei Dai. Image inpainting based on wavelet decomposition. *Procedia Engineering*, 29:3674–3678, 2012.

[232] Jianfu Zhang, Li Niu, Dexin Yang, Liwei Kang, Yaoyi Li, Weijie Zhao, and Liqing Zhang. Gain: Gradient augmented inpainting network for irregular holes. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 1870–1878, 2019.

[233] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.

[234] Yinda Zhang and Thomas Funkhouser. Deep depth completion of a single rgb-d image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 175–185, 2018.

[235] Zhifei Zhang, Yang Song, and Hairong Qi. Age progression/regression by conditional adversarial autoencoder. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5810–5818, 2017.

[236] Hang Zhao, Orazio Gallo, Iuri Frosio, and Jan Kautz. Loss functions for image restoration with neural networks. *IEEE Transactions on computational imaging*, 3(1):47–57, 2016.

[237] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017.

[238] Lei Zhao, Qihang Mo, Sihuan Lin, Zhizhong Wang, Zhiwen Zuo, Haibo Chen, Wei Xing, and Dongming Lu. Uctgan: Diverse image inpainting based on unsupervised cross-space translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5741–5750, 2020.

[239] Chuanxia Zheng, Tat-Jen Cham, and Jianfei Cai. Pluralistic image completion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1438–1447, 2019.

[240] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1452–1464, 2017.

[241] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1452–1464, 2018.

[242] Hang Zhou, Ziwei Liu, Xudong Xu, Ping Luo, and Xiaogang Wang. Vision-infused deep audio inpainting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 283–292, 2019.

[243] Tong Zhou, Changxing Ding, Shaowen Lin, Xinchao Wang, and Dacheng Tao. Learning oracle attention for high-fidelity face completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7680–7689, 2020.

[244] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.

[245] Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A Efros, Oliver Wang, and Eli Shechtman. Toward multimodal image-to-image translation. In *Advances in neural information processing systems*, pages 465–476, 2017.

[246] Yue-ting Zhuang, Yu-shun Wang, Timothy K Shih, and Nick C Tang. Patch-guided facial image inpainting by shape propagation. *Journal of Zhejiang University-SCIENCE A*, 10(2):232–238, 2009.