Deep Generative Models for Unsupervised Scale-Based and Position-Based Disentanglement of Concepts from Face Images

by

Mahla Abdolahnejad

A Dissertation submitted to
the Faculty of Graduate Studies and Research
in partial fulfilment of
the requirements for the degree of
Doctor of Philosophy

Ottawa-Carleton Institute for Electrical and Computer Engineering (OCIECE)
Department of Systems and Computer Engineering
Carleton University
August 2022

Abstract

Among the different categories of natural images, face images are very important because of the role they play in human social interactions. Face images are also considered very challenging subjects in computer vision due to the uniqueness of information contained in individual face images and the wide range of important information that can be perceived from a single face image. It is recognised that despite all the recent advances of artificial intelligence using deep neural networks, computers are still struggling at achieving a rich and flexible understanding of face images comparable to humans' face perception abilities. This thesis aims at finding fully unsupervised ways for learning a transformation from face images pixel space to a representation space in which the underlying facial concepts are captured and disentangled. The objective of this thesis is to move from a representation of face images in which all facial concepts are captured in a single large cluster towards a representation in which facial concepts are separated into distinct groups. We propose that it is possible to utilize clues from the real 3D world in order to guide the representation learner in the direction of disentangling facial concepts. We conduct two studies in order to test this hypothesis. First, we propose a deep autoencoder model for extracting facial concepts based on their scales. We introduce an adaptive resolution reconstruction loss inspired by the fact that different categories of concepts are encoded in (and can be captured from) different resolutions of face images. With

the help of this new reconstruction loss, the deep autoencoder model is able to receive a real face image and compute its representation vector, which not only makes it possible to reconstruct the input image faithfully, but also separates the concepts related to specific scales. We demonstrate that the autoencoder trained using the adaptive resolution reconstruction loss is able to outperform benchmark models in generating faithful and high quality reconstructions of real face images and is able to successfully transfer the facial concepts associated with a specific scale from one input image to another. Second, we introduce a new scheme to enable generative adversarial networks to learn a representation for face images which is composed of the representations for smaller facial components. This is inspired by the fact that all face images display the same underlying structure. As a result, a face image can be divided into parts with fixed positions each containing specific facial components only. Learning a separate distribution for each of these parts is equivalent to disentangling these components in the representation space. We demonstrate that the proposed compositional generative adversarial network is able to produce realistic high-quality face images by generating and piecing together the parts. Additionally, we demonstrate that the model learns the relations between the facial components and their representations. Therefore, the specific facial components are interchangeable between generated face images. Lastly, we show that the proposed compositional generative adversarial network is able to outperform benchmark methods in generating realistic face images while performing compositions in image domain and allowing for local control over generated faces.

Acknowledgments

I would like to take this opportunity to thank my advisor, Professor Peter Xiaoping Liu, for his guidance, assistance, and encouragement. Additionally, I would like to thank my family and friends for their endless love and support. I am also grateful to my lab mates and co-workers for making the lab an inspiring environment to work at.

Table of Contents

List of Tables

List of Figures

List of Acronyms

Acronym	Definition
AdaIN	Adaptive Instance Normalization
ADAM	Adaptive Moment
AI	Artificial Intelligence
ALI	Adversarially Learned Inference
AR	Adaptive Resolution
AUs	Action Units
AVAE	Autoencoding Variational AutoEncoder
BEGAN	Boundary Equilibrium Generative Adversarial Networks
CelebA	Celebrity Faces Attributes
CelebA-HQ	Celebrity Faces Attributes High Quality
cGAN	Conditional Generative Adversarial Networks
CNN	Convolutional Neural Network
CoGAN	Coupled Generative Adversarial Networks

CSVAE	Conditional Subspace Variational AutoEncoder
CycleGAN	Cycle-Consistent Generative Adversarial Networks
DL	Deep Learning
DMN	Dense Mapping Network
DualGAN	Dual Generative Adversarial Networks
ELEGANT	Exchanging Latent Encodings with Generative Adversarial Networks
FC	Fully-Connected
FFHQ	Flickr Faces High Quality
FID	Frechet Inception Distances
GAN	Generative Adversarial Networks
GANimation	Generative Adversarial Networks Animation
GANotation	Generative Adversarial Networks Annotation
GP	Gradient Penalty
HD	High Definition
IAN	Introspective Adversarial Network
IcGAN	Invertible Conditional Generative Adversarial Networks
Info-GAN	Information Maximizing Generative Adversarial Networks
IntroVAE	Introspective Variational AutoEncoder
LAPGAN	Laplacian Generative Adversarial Networks

Leaky-ReLU	Leaky Rectified Linear Unit
MaskGAN	Mask Conditioned Generative Adversarial Networks
ML	Machine Learning
ML-VAE	Multi-Level Variational AutoEncoder
MSE	Mean Squared-Error
NN	Neural Network
PGGAN	Progressive Growing of Generative Adversarial Networks
PIONEER	Progressively Growing Generative Autoencoder
Pix2Pix	Pixels to Pixels Conditional Generative Adversarial Networks
PPL	Perceptual Path Length
RSGAN	Region-Separative Generative Adversarial Networks
SGAN	Stack Generative Adversarial Networks
SPADE	Spatially-Adaptive Denormalization
StarGAN	Star-topology Multi-Domains Generative Adversarial Networks
StyleALAE	Style Adversarial Aatent AutoEncoder
StyleGAN	Style-Based Generative Adversarial Networks
TCVAE	Total Correlation Variational AutoEncoder
UNIT	UNsupervised Image-to-image Translation
UTKFace	UTK Faces

VAE	Variational AutoEncoder
WGAN	Wasserstein Generative Adversarial Networks

List of Symbols

Symbol	Definition
x	Data points
y	Class labels or target outputs
$p(y\|x)$	Conditional probability distribution of y given x
$p(x)$	Probability distribution of x
$x(t)$	Input function
$w(t)$	Kernel or weight function
$x(t) * w(t)$	Convolution of two function $x(t)$ and $w(t)$
$s(t) = x(t) * w(t)$	Feature map of $x(t)$ and $w(t)$
$z \in \mathcal{Z}$	Entangled latent representation space
$p(z)$	Prior probability distribution of latent variable z
$w \in \mathcal{W}$	Space of latent representations in which concepts are disentangled based on scale also referred to as style vectors
$\zeta : \mathcal{X} \to \mathcal{Z}$	Encoder network of a standard autoencoder

$\eta : \mathcal{Z} \to \mathcal{X}$	Decoder network of a standard autoencoder
$\mathcal{X} = \{x_1, ..., x_N\}$	Dataset of N data points
$\mathcal{L}(x_n)$	Reconstruction loss for a data point x_n
$\delta : \mathcal{X} \to \mathcal{W}$	Encoder network of the deep autoencoder proposed in this thesis
$\phi : \mathcal{W} \to \mathcal{X}$	Decoder network of the deep autoencoder proposed in this thesis
$\mu(x)$	Mean of x
$\sigma^2(x)$	Variance of x
N_l	The number of convolutional layers of the decoder network
$p \sim U[1, N_l]$	Sample from a discrete uniform distribution
$mix(w_1, w_2, p)$	Outputs w_{mixed} which is equal to w_1 before splitting point p and is equal to w_2 after it
$r \times r$	Image resolution
$D(x; \theta_d) : \mathcal{X} \to \mathbb{R}$	Discriminator Network of a standard GAN
$G(z; \theta_g) : \mathcal{Z} \to \mathcal{X}$	Generator Network of a standard GAN
$V(D, G)$	Value function of GAN
γ	Weight of the adversarial loss in the proposed deep autoencoder

$x \sim p_{data}(x),\ x \in \mathcal{X}$	Data distribution
$\{z_i\}_{i=1}^{K}$	Parts latent representations of the proposed compositional GAN
$\{p(z_i)\}_{i=1}^{K}$	Prior probability distribution of latent variables $\{z_i\}_{i=1}^{K}$
$\{x_i\}_{i=1}^{K}$	Image parts of the proposed compositional GAN
R_z	Syntactic method of latent representations $\{z_i\}_{i=1}^{K}$
R_x	Syntactic method of image parts $\{x_i\}_{i=1}^{K}$
$h()$	Maps $\{z_i\}_{i=1}^{K}$ to $\{x_i\}_{i=1}^{K}$ and R_z to R_x
$f()$	Maps image parts $\{x_i\}_{i=1}^{K}$ and R_x to whole images x
$G(z_i, \theta_h, \theta_f)$	Generator Network of the proposed Compositional GAN
$v = w \times h \times c$	Volume of size w by h by c
$\{v_i\}_{i=1}^{K}$	Set of K volumes
x_{n,z_t}	N images with every latent representation fixed except for z_t
σ_{z_t}	Pixel-wise standard deviation of x_{n,z_t}
v_{z_t}	Normalized pixel-wise standard deviation of x_{n,z_t}
α	Parameter of Leaky-Relu activation function

β_1, β_2, and ϵ	Parameters of ADAM optimizer

Chapter 1

Introduction

Understanding human faces is an essential topic of social intelligence. Let us imagine a person who does not have the ability to recognize and re-identify faces and/or is not able to understand the social clues in the faces such as anger or surprise. It is clear that such disability will cause tremendous difficulties in daily life especially in forming personal and professional relationships. Processing faces is an important part of human intelligence and the goal of artificial intelligence (AI) is to build machines that think and learn like humans. Consequently, a significant step toward gaining true intelligence for computers is for them to fully understand human faces. In fact, computers cannot be considered truly intelligent without fully understanding human faces.

1.1 Motivation: The Gap Between Face Perception by Human Brain and by Computers

The idea that perception is a unique ability of biological systems has been accepted for decades. However, in recent years, dramatic improvements in computer models of perception through the use of deep learning (DL) approaches have changed this idea. Neural networks (NN), which are inspired to some degree by the hierarchical

architecture of the primate visual system, are able to outperform humans in many vision tasks [1]. However, in regard to face perception, it is apparent that the current state of machines' face perception is still well behind humans'.

It is widely recognised that face perception is considered to be the most developed visual perception skill in the human brain [2]. Our brain has the capacity to perceive the unique identity of an endless number of different faces. Identity recognition is an essential part of human intelligence and is crucial for human social interactions. Identity recognition is based on perceiving aspects of facial structure that do not change with expression and facial movements. However there are changeable aspects of a face such as facial expressions, eyes gaze direction, and speech-related movements of the mouth which play a very important role in social interactions as well. Processing and representing this changeable aspect probably requires a further developed visual perceptual skill. Face perception in the human brain represents both the invariant (related to identity) and the variant (related to expressions and attributes) aspects of faces by a specialized distributed neural system consisting of three regions [3]. Furthermore, it had been revealed that recognition of identity, and recognition of expressions and speech-related movements of the mouth are done using two different regions [4]. Recent advances of AI using DL-based methods and the increased computational power have resulted in computers achieving a near human performance in identity recognition [5]. However, when it comes to recognition of the variant aspects of faces such as facial expressions, there is still a large gap between human and computers levels of performance.

Faces are also considered as one of the most informative visual stimuli our brains ever receive [6]. They are complex multidimensional patterns providing information such as identity, gender, mood, age, race, and direction of attention which can be perceived by human adults simply through a glimpse of an individual's face. Many studies have been focused on understanding the temporal dynamics of face perception

in the human brain at a global level. The results of these studies show that the complete face processing is achieved 200 ms after receiving stimulus [7]. Despite all the recent progress and remarkable advances of AI using DL, achieving nearly the same level of accuracy for computers requires tens or hundreds of labeled examples and hours of training through expensive computations by multiple GPUs. Therefore computers are not able to learn a concept from only one example. Moreover, they are far behind humans in unsupervised learning of concepts. The reason that computers have been able to achieve a better performance in identity recognition in comparison to the recognition of variant aspects of face such as mood and emotions, could be the fact that less amount of accurately labeled data is available for the second group.

Finally, it has also been argued in several studies that computers fail to use learned concepts in a rich and flexible way like humans, and therefore are not able to generalize well to the examples outside of the training examples. In [8] some experiments are designed to compare human visual perception with deep neural networks used in the computer vision field. It was shown in this study that deep CNNs are much more proficient than humans in specific tasks such as counting but they experience much more difficulty in learning semantic concepts such as symmetry, uniformity, and conformance. In summary, it is evident that machine visual perception of semantic concepts, such as many of the concepts embedded in a face image, is still far behind human perception even when enough training data is available.

1.2 Problem Statement: Unsupervised Disentanglement of Concepts from Face Images

The majority of the recent advances in face processing by computers are based on supervised identity recognition or supervised face detection. It means that millions of

face images each labelled with identity or millions of images each annotated with the location of faces are utilized to train large deep neural networks. Using these training examples x and their corresponding target outputs y, these models learn an estimation for the complicated conditional probability distribution $p(y|x)$. After training is completed, they can be used to predict the target output y for any given input x. In a supervised learning process, the deep neural networks learn to extract the information related to a label only. Consequently, it is not possible to extract/disentangle information related to a concept for which no label is available. However, face images are highly informative meaning that a wide spectrum of information including head pose, gaze, identity, gender, expression, and more can be perceived by looking at a single face image. Given the great cost of labelling, preparing labels for the wide range of concepts that can be perceived from a single face image is not feasible. Moreover, deep neural networks require very large datasets to be trained. For these reasons, it is necessary to move towards unsupervised ways of disentangling concepts if we wish to improve the computers face perception abilities.

It is widely recognized that for addressing this problem we should find ways to guide computers towards learning a comprehensive, rich, and flexible representation of face images. Such representation must capture and disentangle all the underlying explanatory concepts from face images, and not only the ones related to the available labels [9] [10]. The underlying facial concepts that need to be extracted and disentangled include abstract high-level concepts such as identity and age, and also low-level concepts at a range of scales (i.e. from coarse scales such as head pose and hairstyle to fine details such as color of eyes, and shape of the mouth).

Gaining such a rich, comprehensive, and disentangled understanding of faces for computers has numerous potential applications. For example, computers can become much better in person re-identification through different appearance changes.

This has great potential for security purposes such as disguise detection and disguise-invariant face recognition. Additionally, computers will be able to understand and notice changes in appearance, emotions and expressions and respond/behave accordingly. This can dramatically improve the field of Human Machine Interactions (HCI). Furthermore, computers can synthesize non existing faces of high quality with strict control over their features. This has the potential to advancing the AI models by improving the current face datasets.

1.3 Research Objectives: Learning Improved Representations for Face Images

Thus far, it was discussed that the use of DL-based models and the increase in the amount of available data and the computers' computational power in recent years have led to dramatic improvements in computers face processing abilities. These improvements resulted in computers achieving a near human performance in supervised tasks such as face recognition. As a result of these advancements, the goals regarding machines' face perception were transformed; The focus of many studies currently is for machines to be able to predict what a person would look like in 20 years, or when they were a child, or with significant changes in their appearance. Solving such problems requires a rich and flexible understanding of face images in which their underlying explanatory concepts are captured and disentangled. This type of understanding is much closer to the way humans perceive faces; When a person is asked to describe a face they have the ability to separate facial concepts into groups and describe each group. For instance, they can describe the more general concepts such as the shape of the face or the general hairstyle. They can also go into more detailed features of a face such as color of eyes, shape of nose, texture of skin, and so on. Additionally,

there are other types of facial concepts that are more abstract such as age, gender, ethnicity, and identity. Humans excel at recognizing this type of concepts as well. An ideal face representation for computers must have similar abilities. More precisely, an ideal face representation must be able to extract and separate both low-level and high-level facial concepts from face images and without the help of any labels or supervised clues.

This thesis can be considered as part of the broad research efforts for enabling computers to learn more comprehensive and less entangled representations for face images. More precisely, the objective of this thesis is to find fully unsupervised ways for learning a transformation from face images pixel space to a representation space in which the underlying facial concepts are captured and disentangled. In other words, we would like to move from a representation of face images in which all facial concepts are captured in a single large cluster towards a representation in which facial concepts are separated into distinct meaningful groups.

1.4 Hypothesis: Scale and Position as Unsupervised Clues for Disentanglement of Concepts from Face Images

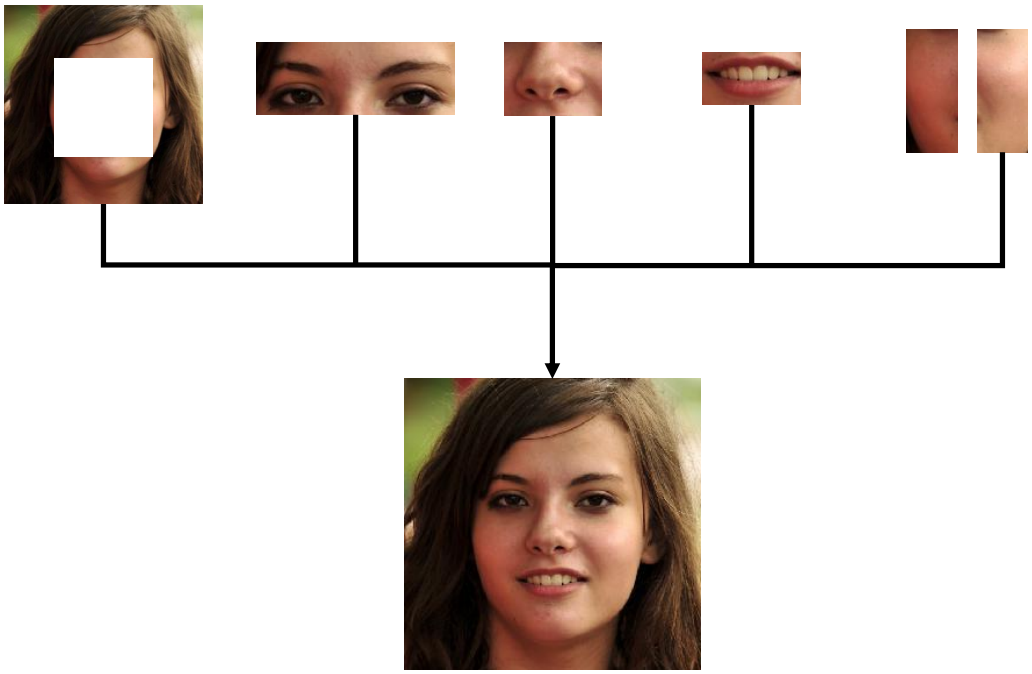
The main hypothesis of this thesis is that it is possible to utilize unsupervised clues from the real 3-dimensional world in order to guide the model learning a representation for face images in the direction of disentangling facial concepts. In this thesis we examine the applicability of two potential clues for disentangling facial concepts; the first is the scale at which a facial concept appears, and the second is the position within the face structure at which a facial component/concept is located.

The idea of scale-based disentanglement of facial concepts is inspired by the fact

that different categories of concepts are encoded in different resolutions of a face image. As it is illustrated in Figure 1.1 (a), information about the overall structure of a face image including head pose, hairstyle, and face shape are encoded in lower resolutions or blurrier versions of face images. Meanwhile, information related to identity and expressions are added to the overall structure by slightly higher resolutions or less blurry versions of face images. Finally, more subtle details are added to the face by higher resolutions or high definition (HD) versions of face images. In other words, facial concepts of different scales appear in different resolutions of face images. Given this, we propose that it is possible to capture different groups of facial concepts by enforcing a constraint on different resolutions of face images. More precisely, it is possible to make the representation learner divide the facial concepts into groups based on the scale/resolution in which they appear.

The idea of disentangling facial concepts base on position is inspired the natural structure that is present in all face images. Face images show a complex visual pattern that always follows the same underlying structure. We propose that since all face images display the same structure, a sense of position for meaningful facial components within a face image can be achieved by dividing the image into parts with fixed positions each containing specific components only. We can guide the representation learner to learn distinct representations for these parts and then piece them together in order to come up with a representation for the entire face. This is in fact a way of integrating compositionality in computers way of understanding faces. Compositionality is the idea that a new concept can be constructed by combining the primitive components and it plays an important role in the way humans perceive the visual world. An example of dividing a face image into parts each containing specific facial components is shown in Figure 1.1 (b).

(a) Scale-based Disentanglement of Facial Concepts



(b) Position-based Disentanglement of Facial Concepts

Figure 1.1: It is illustrated for a sample from FFHQ [11] dataset that; (a) Information about the overall structure of a face are encoded in blurrier versions, information related to identity and expressions are added to the overall structure by less blurry versions, and higher resolutions only add more details to the face, (b) It is possible to divide a face into parts with fixed positions and each containing specific facial components.

1.5 Research Methods

This section starts with introducing the general methodologies that have been shown promising in the past for similar objectives and problems. It then proceeds to going over the specific methodologies that are proposed in this thesis.

1.5.1 Deep Generative models

An important group of methods that have shown promising results towards unsupervised disentanglement of facial concepts are the deep generative models that learn the underlying distribution of a dataset through the help of a latent space representation. Deep generative models are an important group of unsupervised machine learning methods. A generative model uses many training examples to estimate the probability distribution of a training dataset. Deep generative models are particularly useful in the inference of latent feature representations from data which is a very important component of unsupervised learning. Additionally, deep generative models are able to generate realistic samples similar to training examples. This makes them useful for numerous applications such as super resolution, colorization, and image completion. Deep generative models can be grouped into two main categories; the generative models that estimate the probability function explicitly, and the generative models that perform implicit density estimation. The first group comes up with an estimation for the dataset distribution function, while the second group is only able to generate new samples from this distribution.

Autoencoders [12] are a very important group of deep generative models that estimate the dataset probability function explicitly. Autoencoders are a data dimensionality reduction approach that are usually implemented using neural networks. The objective of the training process for autoencoders is to create feature representations that are able to reconstruct the original input data. An autoencoder consists

of two networks with reverse/opposite architectures; the encoder network and the decoder network (see Figure 1.2). The encoder network is trained to generate a latent representation from an image. The decoder network is trained to perform reconstruction from this latent representation. The loss function is defined to force the reconstructed data to be similar to the input data. The ability of autoencoders to reconstruct training data makes it possible to examine and explore the latent representation space through performing tasks such as interpolations between datapoints in the representation space or performing meaningful modifications of datapoints in the representation space. Autoencoders are specifically successful in inferring an interpretable latent representation that captures the global structure of a natural image.

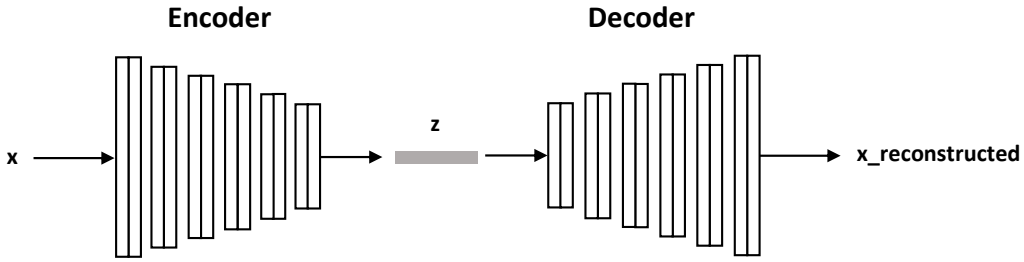


Figure 1.2: A schematic view of an autoencoder model.

Generative Adversarial Networks (GAN) [13] are another very important group of deep generative models that estimate the dataset probability function implicitly. It means, instead of explicitly modelling and solving the dataset density function, GAN focuses on generating samples from this distribution. GAN's main idea is to first sample from a known latent distribution and then learn to transform this sample into a sample of training distribution. Two networks are involved in a GAN model; discriminator network and generator network (see Figure 1.3). The discriminator network is trained to distinguish correctly between real and fake samples. Meanwhile, the generator network is trained to fool the discriminator into thinking that its output is real. The training process of GANs constantly alternates between these two

steps which can be interpreted as the two networks playing a game. GANs are very powerful tools for modelling data distributions. Their sampling process is fast and exact and results in generating high quality samples. Some extensions of GAN framework achieve state-of-the-art results in generating realistic face images of non-existing identities.

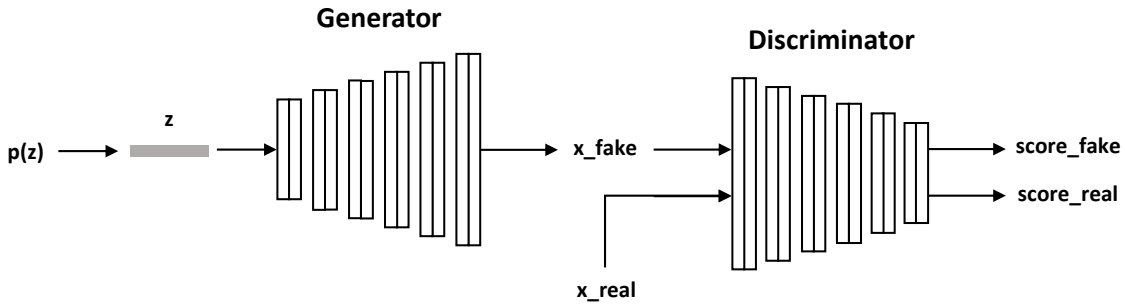


Figure 1.3: Schematic view of a generative adversarial networks (GAN) model.

There are several studies that focus on modifying the standard autoencoders and/or GANs in order to achieve unsupervised disentanglement of facial concepts in their latent representations. A number of these studies show promising results in learning and disentangling meaningful facial concepts from face images in a fully unsupervised way. These studies provide evidence for the potential of these methods in achieving better representations for face images. That is why we too utilise the principles of autoencoders and GANs as the foundation for the methodologies proposed in this thesis. The next two sections briefly introduce these proposed methodologies.

1.5.2 A Deep Autoencoder for Disentanglement of Facial Concepts Based on Scale

The first study conducted in this thesis is a deep autoencoder model built on the principles of standard autoencoders and for the purpose of extracting facial concepts based on their scale. Similar to the standard autoencoders, the autoencoder proposed

here consists of an encoder network and a decoder network. The encoder network is trained to receive an input face image and compute a latent representation vector for it. Meanwhile, the decoder network is trained to generate an image given an input representation vector. In order for the facial concepts related to specific scales to be separated in the latent representations, we introduce an adaptive resolution (AR) reconstruction loss for training the autoencoder. The proposed AR reconstruction loss is inspired by the fact that facial concepts of different scales appear in different resolutions of face images. Therefore, it is possible to disentangle different scales of facial concepts by enforcing a constraint on different resolutions of face images.

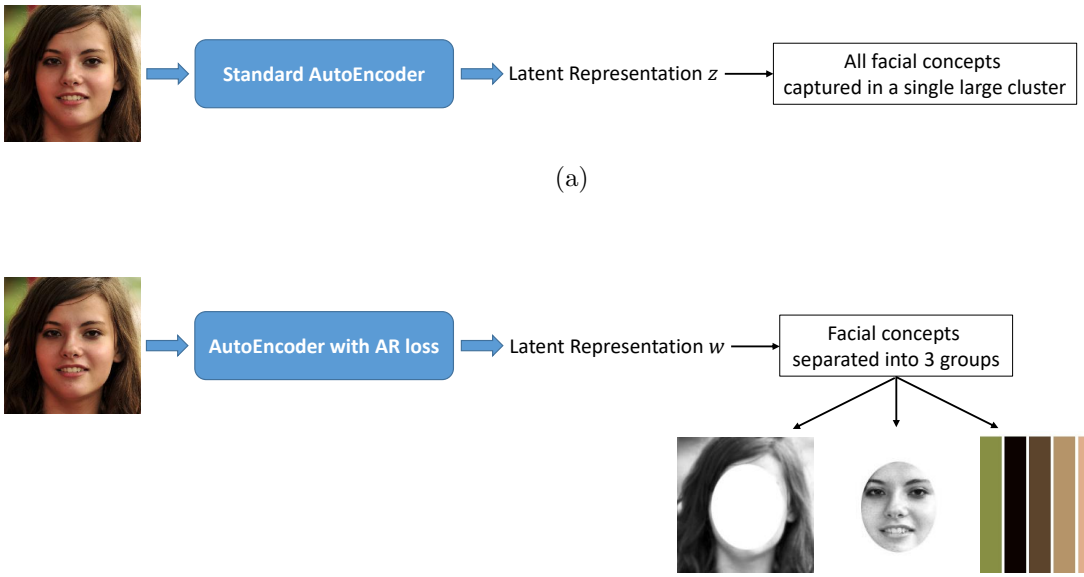


Figure 1.4: (a) Standard autoencoders learn a latent representation in which all facial concepts are captured in a single group. (b) The deep autoencoder proposed here learns a latent representation in which facial concepts are separated into three distinct groups. The displayed face is a sample from FFHQ [11] dataset.

With the help of the AR reconstruction loss, the proposed deep autoencoder is able to compute a representation of face images that makes it possible to not only reconstruct the input images faithfully, but also disentangle the concepts related

to specific scales. As a result, it is possible to use this deep autoencoder in order to generate realistic reconstructed images associated with a combination of latent representations from different source faces. In other words, it is possible to modify a given face image in meaningful and controlled ways by transferring the facial concepts associated with a specific scale from another face. We demonstrate through extensive evaluations that, as illustrated in Figure 1.4, the proposed deep autoencoder moves from a latent representation in which all facial concepts are captured in a single large group towards learning a latent representation in which facial concepts are separated into three distinct groups; The first group captures coarse-scale facial concepts such as background, head pose, hairstyle, and face shape. The second group captures middle-scale facial features such as information related to identity and facial expressions. Finally, the third group captures fine-scale features such as colour themes including the background colours, hair colour, skin colour, and the general lighting of the image

1.5.3 A Compositional GAN for Disentanglement of Facial Components Based on Position

The deep autoencoder model introduced in the previous section uses a coarse-to-fine hierarchy in order to achieve disentanglement of facial concept in its representation learning process. In the second study of this thesis we propose that it is possible to introduce another way of adding structure and hierarchy to the representation learning of face images and possibly achieve a different type of disentanglement. We introduce a compositional GAN inspired by the natural structure that is present in all face images. Given this natural structure, we propose a methodology for building and training of GANs in order to learn the distribution of face images as compositions of distributions of smaller parts with fixed positions. We define each part to only contain specific facial components. As a result, learning a separate distribution for

each part is equivalent to disentangling these components in the representation space.

The compositional GAN proposed here achieves a more flexible and less entangled representation of faces and leads to easier generalization to examples outside training data. As it is illustrated in Figure 1.5, the proposed compositional GAN moves from a latent representation for GANs in which all facial concepts are captured in a single large group towards learning multiple latent representations for different groups of facial concepts/components. Two models are developed in this study; 1) A two-part compositional GAN for learning the representation of face images composed of two parts; one representing the face and the other representing hair&background. 2) A four-part compositional GAN for learning the representation of cropped faces composed of four components; eyes, nose, mouth, and jaw&cheeks. We demonstrate that the proposed models are able to produce realistic high-quality face images by sampling from the learned distributions for parts and then generating the parts and piecing them together. Additionally, we demonstrate that the model learns the relations between the facial parts and their distributions. Therefore, the specific facial parts are interchangeable between generated face images. In other words, the models can generate realistic whole faces given any combination of samples from components/parts distributions.

1.6 Contributions

The main contribution of this thesis is conducting two studies in order to examine the applicability of two unsupervised clues for learning a representation of face images in which different groups of facial concepts are disentangled. These two unsupervised clues, which are introduced inspired by facts from the physical world, include the following; the scale at which facial concepts appear, and the position within the face structure at which facial components/concepts are located.

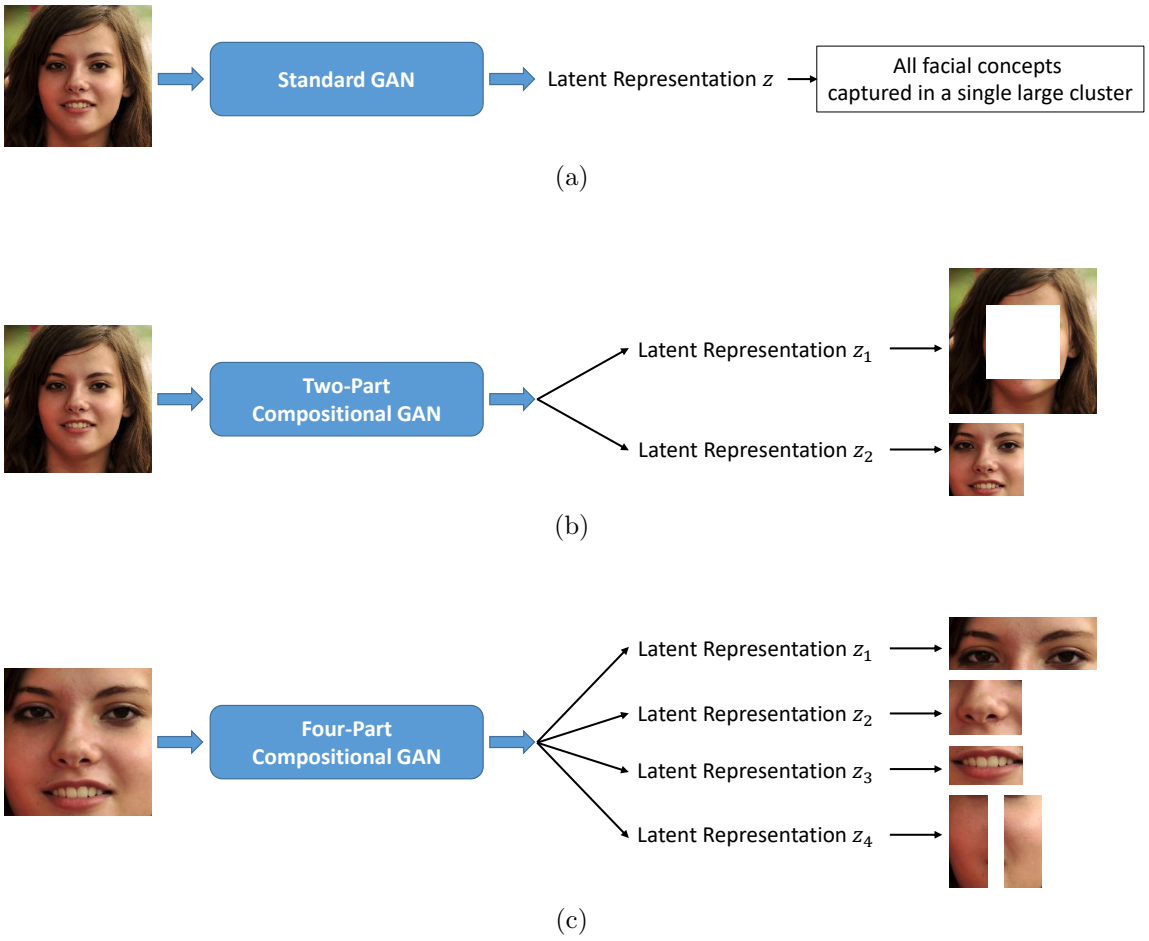


Figure 1.5: (a) Standard GANs learn a latent representation in which all facial concepts are captured in a single group. (b) The two-part compositional GAN model proposed here learns two distinct latent representations for face images. (c) The four-part compositional GAN model proposed here learns four distinct latent representations for cropped faces. The displayed face is a sample from FFHQ [11] dataset.s

The contributions of the first study of this thesis are as follows. A deep autoencoder is proposed for computing latent representations for face images in which the facial concepts are disentangled based on scale. An adaptive-resolution reconstruction loss is introduced for enforcing the autoencoder to achieve such disentanglement in its learned latent representation. It is shown that the proposed autoencoder achieves improvement in faithful and realistic reconstruction of real face images in comparison to benchmark models, in addition to being able to transfer the facial concepts associated with a specific scale from one input face image to another.

The contributions of the second study of this thesis are as follows. It proposes a methodology for enabling GANs to learn the distribution of face images as a composition of distributions of facial components. A straightforward method is introduced in order to improve a standard GANs architecture such that the modified architecture is able to learn multiple localized representations for a face. It is demonstrated that the proposed compositional GAN not only learns the representations for facial components but also the relations between them. As a result, it can generate realistic whole faces given any combination of samples from components/parts distributions. Moreover, it is shown that the proposed compositional GAN is able to outperform benchmark methods in generating realistic face images while performing compositions in image domain and allowing for local control over generated faces.

The following publications are based on this thesis.

J1 Abdolahnejad, Mahla, and Peter Xiaoping Liu. "Deep learning for face image synthesis and semantic manipulations: a review and future perspectives." Artificial Intelligence Review, 53.8, (2020): 5847-5880. [14]

J2 Abdolahnejad, Mahla, and Peter Xiaoping Liu. "A Deep Autoencoder With

Novel Adaptive Resolution Reconstruction Loss for Disentanglement of Concepts in Face Images." IEEE Transactions on Instrumentation and Measurement, 71, (2022): 1-13. [15]

J3 Abdolahnejad, Mahla, and Peter Xiaoping Liu. "A Compositional Generative Adversarial Networks for Learning Disentangled Representation of Face Images." Journal paper under review.

1.7 Thesis Organization

The rest of this thesis is organized as follows. In Chapter 2, we provide a literature review of deep generative models based on autoencoders, GANs, or both that achieve promising results in disentanglement of facial concepts in their learned representation for face images. In Chapter 3, we present the deep autoencoder with an adaptive-resolution reconstruction loss proposed for learning a representation for face images in which the facial concepts are disentangled based on scale. In Chapter 4, we present the compositional GAN proposed for learning disentangled representation of facial concepts based on their position in the face structure. In Chapter 5, we summarize the methodologies introduced in this thesis and outline the conclusions of the studies conducted in this thesis.

Chapter 2

Literature Review

This chapter starts with introducing the background, basic principles, and methods of machine learning (ML) and deep learning that will be referred to throughout this thesis. It then proceeds to reviewing the major deep generative models that use autoencoders, GANs, or a combination of both for learning a representation for face images and achieve some type of disentanglement in their representation space.

2.1 Background

The reference for this entire section is the "Deep Learning" textbook by Goodfellow et al. [10].

2.1.1 Hand-Crafted Knowledge versus Machine Learning

From a historical point of view, the early successes in the field of AI are related to problems that involve hand-crafting and hard-coding a certain list of formal math-based rules into a program. These types of problems are usually very difficult for humans to solve but relatively straight-forward for computers to process. This group of methods are usually referred to as knowledge-based. The knowledge-based AI methods attempt to extract knowledge about the world in formal languages and build

algorithms with the knowledge hard-coded in them. The algorithms are then able to reason based on these knowledge and by using logical inference rules. Nonetheless, these approaches to AI did not lead to major breakthroughs in the field.

Human intelligence includes a huge amount of subjective and intuitive knowledge about the world. A knowledge that is very difficult to be described by a finite set of formal rules. In order to achieve intelligence, computers are required to capture this type of knowledge as well. However, to find a way for computers to gain this type of informal knowledge was a serious challenge in AI. This challenge brought about the need for computers to have the ability to acquire their own knowledge by looking at examples exactly like humans do. This idea led to the introduction of a group of methods called machine learning; "a machine learning algorithm is an algorithm that is able to learn from data". Machine learning allows computers to gather information from experience and observation. Therefore, they eliminate the need for human programmers to capture, formalize, and hard-code all the information that computers will need. The introduction of machine learning led to major successes for computers regarding solving real-world problems.

2.1.2 Supervised versus Unsupervised Machine Learning

Machine learning methods fall under two main categories; supervised methods, and unsupervised methods. Supervised learning algorithms are provided with a dataset containing both data examples and the labels or targets associated with data examples. They learn an optimum mapping function from data points to labels. This learned mapping function can be utilized to perform tasks such as classification, regression, and object detection. Most of the recent advances in AI and deep learning are based on supervised learning. Supervised clues or labels are strong tools to help the algorithm in the direction of learning the representation for a desired concept only. More precisely, supervised methods learn to extract the information related to

the label while discarding the information non-related to the label. As a result, they are not able to extract information related to a concept for which no label is available.

In unsupervised learning no labels or targets are provided for data points. Unsupervised learning algorithms are provided with a dataset containing a large number of data examples only and learn the underlying structures of data or at least some useful properties of these structures. More precisely, unsupervised learning algorithms attempt to either learn the entire probability distribution that generated a dataset or perform tasks such as dividing the dataset into clusters. The unsupervised learning of the probability distribution of an entire dataset is a task that deep learning has been particularly successful in.

2.1.3 Representation Learning

The performance of machine learning algorithms is very much dependent on the set of features or the representation of data that they are provided with. For instance, when a machine learning algorithm is used to diagnose a patient it does not examine the patient itself. The algorithm receives only a set of measurements for the patient and therefore its performance is completely dependent on how well the features/measurements are correlated with the various possible outcomes. It is common for many information processing tasks in computer science or even in daily life to become easier or more difficult depending on how the information is represented.

In machine learning a good representation is a representation that makes a desired task easier to perform. Therefore, the choice of representation may vary depending on the task on hand. A complicated AI task can be solved using simple machine learning algorithms if the right set of features for that task are designed and extracted. However, for many tasks it is extremely difficult to choose the features to extract. Therefore, an alternative solution is to use the machine learning algorithm itself to first learn and extract the representations and then learn the mapping from the

representations to output. This technique is called representation learning and usually results in a better performance for learning algorithms in comparison to using hand-designed representations.

The goal in building representation learning algorithms is to separate/disentangle factors of variations behind the data examples. Factors of variations are in fact the sources of influence from the real physical world and their combination or configuration causes or explains the data examples. However, it is usually quite difficult to extract high-level abstract features from raw data. For example, extracting age from a face image requires near-human understanding of faces. As a result, designing representation learning algorithms that result in disentangling all factors of variation behind data is an extremely challenging problem in machine learning. Deep learning is an exceptionally helpful tool for addressing this central problem of representation learning.

2.1.4 Deep Learning

Deep learning (or deep neural networks) is a type of machine learning that achieves great power and flexibility by representing the world as a nested hierarchy of concepts. In deep learning each concept is defined by simpler ones and more abstract concepts are computed in terms of less abstract ones. Deep neural networks are in fact complex computational models consisting of stacked layers of artificial neurons each working as a processing layer. They are to some extent inspired by the neuroscience of human brain and that is why they are referred to as neural networks. However, modern deep neural networks are not considered to be the perfect models of the brain. Nevertheless, they are powerful function approximation tools that are loosely inspired by the brain neurons architectures.

It is important to note that the supervised deep neural networks can be considered as a type of representation learning model. More precisely, the hidden layers of a deep

neural network learn a good representation from raw data and provide it to the output layer which acts as a simple linear classifier for classifying the learned representations. As a result, training for a specific label leads to the representation at each layer to only extract and carry out the information from raw data that makes that particular classification task easier. For instance, some very challenging classification problem in the input data level can turn into a linearly separable classification problem in the last hidden layer. Figure 2.1 displays a Venn diagram of how representation learning, which is the main focus of this thesis, relates to the other AI disciplines.

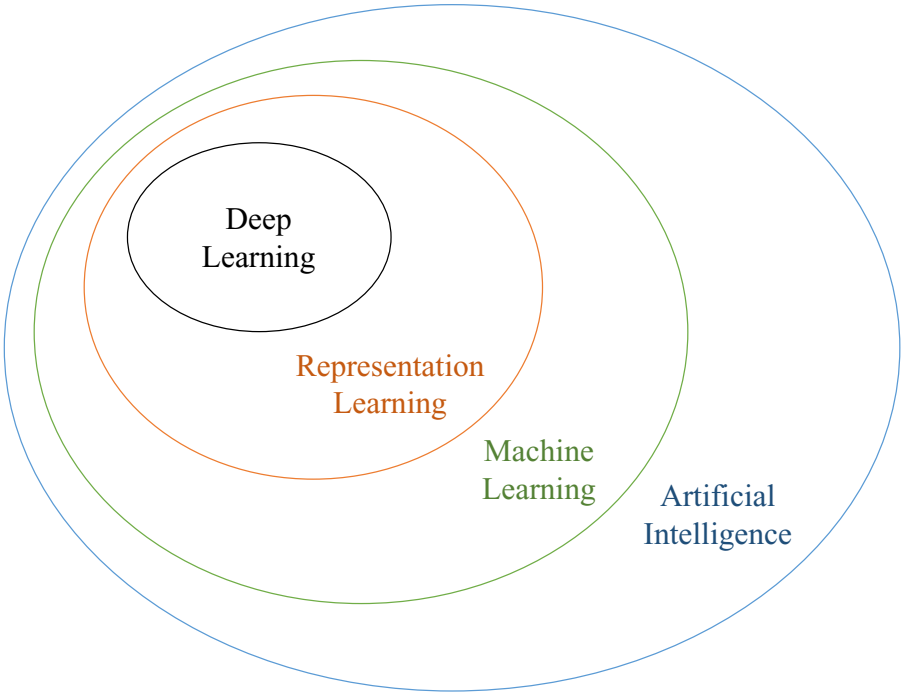


Figure 2.1: The Venn diagram of relationship among different AI disciplines including machine learning, representation learning, and deep learning (adapted from [10]).

2.1.5 Convolutional Neural Networks

Convolutional neural networks (CNNs) are a type of neural networks that are specialized for processing multi-dimensional grid-like data such as images. These networks

utilize convolution operation instead of general matrix multiplication operation in at least one of their layers. Convolution is an operation on two functions of real-valued inputs defined as

$$s(t) = x(t) * w(t) = \int_{-\infty}^{\infty} x(a)w(t-a)da. \tag{2.1}$$

In a convolutional layer, the first function $x(t)$ is referred to as the input, while the second function $w(t)$ is referred to as the kernel. Additionally, the output function $s(t)$ is usually referred to as the feature map. In deep CNNs, the input is usually a multidimensional array of data such as an RGB image and the kernel is a multidimensional array of parameters/weights that their values are optimized by the learning algorithm and according to a cost function.

Convolution operation benefits from three important properties that make them ideal for improving machine learning systems. These three properties include sparse interactions, parameter sharing, and equivariant representations. In a traditional neural network every output unit is influenced by every input unit. In CNNs, by having a kernel smaller than the input, every output unit is influenced only by the number of units from input that the kernel can cover. This helps convolutional layers to detect small meaningful features such as edges in the image. Furthermore, in a convolutional layer a kernel scans the whole image starting from the top left corner. This means that in convolutional layers a weight parameter in used everywhere in the input and therefore the value of a weight used at one location is tied to the value of weight applied to other locations. This is in contrast to the traditional neural networks where each weight parameter is used only once for computing the output of a layer. As a result, instead of learning a set of weights for each location, CNNs learn one set for the entire input image. This property of CNNs is called parameter sharing. Parameter sharing in CNNs leads to a third beneficial property called equivariant representations. The equivariant property for convolutional layers means that if we

transform the input in some ways and then apply the convolution the result will be the same as applying the convolution first and the transformation then. For example when the input is an image, convolutional layers create a 2-dimensional map showing what features appear where in the image. If we translate the input in some way, the feature map will undergo the same translation as well.

In short, CNNs are neural networks specialized for dealing with grid-like topology of input and make it possible to create large networks for this type of data. They are the most successful neural networks for processing images and are also some of the first deep learning models that were able to perform well and solve commercial problems.

2.2 Deep Generative Models

Deep generative models are an important group of unsupervised methods used for understanding a complete dataset. A deep generative model uses many training examples to estimate the probability distribution of training data. Generative models are particularly useful in the inference of latent feature representations in data. They can generate realistic samples similar to training examples which makes them useful for many different applications such as super resolution, colorization, image completion, etc. An important group of methods that have shown promising results towards disentanglement of facial concepts are the deep generative models that learn the underlying distribution of a dataset through the help of a latent space representation. Different types of autoencoders and GANs are two important methods in this group.

2.2.1 AutoEncoders

As discussed briefly in the previous chapter, autoencoders are a data dimensionality reduction approach that have long been implemented using neural networks [12]. The

objective of the training process for autoencoders is to create feature representations that are able to reconstruct the original input data. An autoencoder consists of two networks with reverse/opposite architectures; the encoder network and the decoder network. The encoder network is trained to generate a latent representation z from an image. The decoder network is trained to perform reconstruction from this latent representation z. The loss function is defined to force the reconstructed data to be similar to the input data. After the training process is completed, the encoder network is used to compute the latent representations and the decoder network is used to generate images corresponding to samples from the latent representation space. The ability of autoencoders to reconstruct training data makes it possible to examine and explore the latent representation space through performing tasks such as interpolations between datapoints in the representation space or performing meaningful modifications of datapoints in the representation space.

One important modification of autoencoders are a group of methods called the Variational AutoEncoders (VAEs) [16] [17]. The main assumption in VAEs is that the training data $\{x^{(i)}\}_{i=1}^{N}$ is generated from an underlying latent representation $\{z^{(i)}\}_{i=1}^{N}$ with a prior distribution $p(z)$. As a result, VAEs are able to sample from a known latent prior in order to generate new data points. In VAEs an intractable density function for the marginal likelihood of training data is defined with the additional latent variable z. Instead of optimizing the likelihood, VAEs optimize a tractable lower bound on the likelihood. The training process of a VAE can be interpreted as finding estimations of true parameters for encoder and decoder in order to maximize the lower bound on the marginal likelihood of training data points.

There are several variations of autoencoders and VAEs that approach the problem of disentangling the facial concepts in the latent space. conditional VAE [18] uses the encoder and decoder networks in order to separate the hidden factors of variation in the data from variations related to class labels. To achieve this, the encoder output

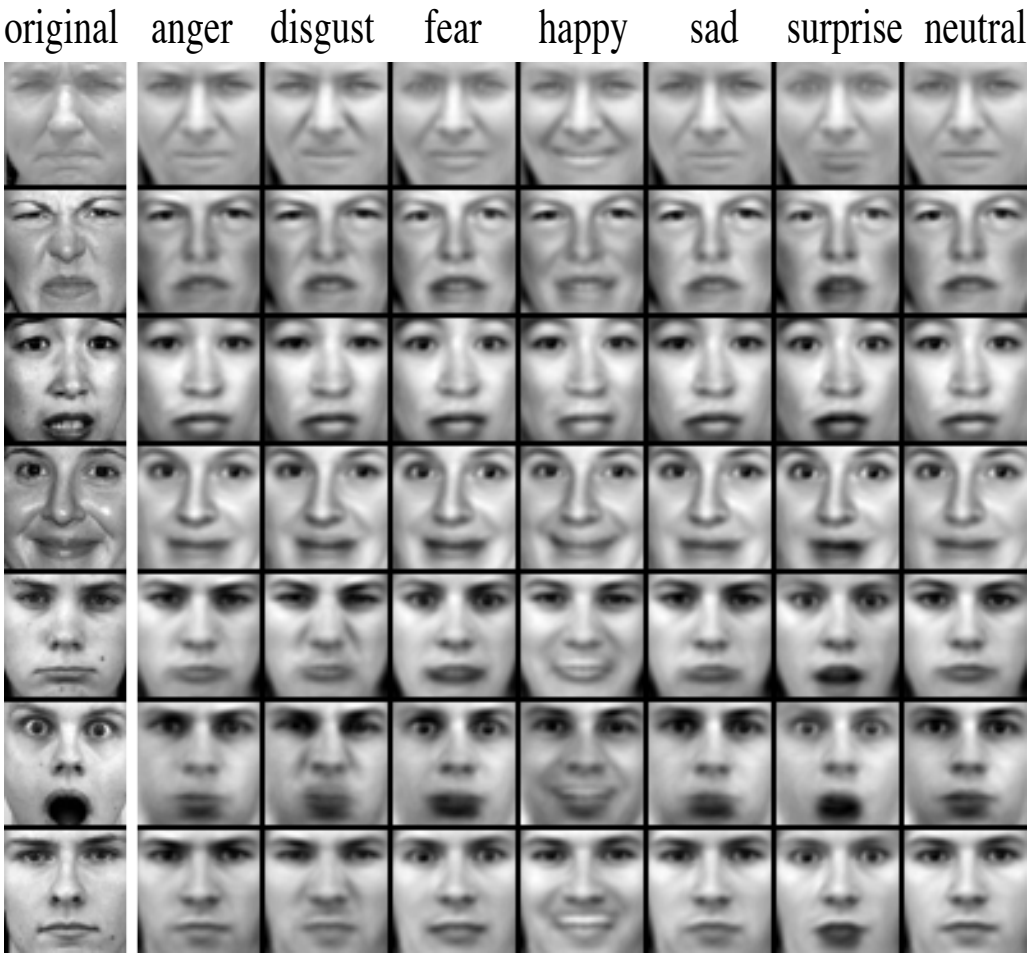


Figure 2.2: Examples of generated samples by a conditional VAE model which separates the hidden factors of variation (z) in the training data from variations related to expression labels (y). Reprinted from [18] with permission.

is divided into two sets of variables: the class label variable y, and the latent variable z. Additionally, two regularization terms are added to the original VAE loss function in order to detach representations in the autoencoder. The first term is a simple cross-entropy term which implements discriminative cost on class labels. The second term computes cross-covariance penalty between the observed and latent variables which detaches the two types of variables. As a result, this framework prevents the latent variables from encoding variations related to the class labels. Therefore,

when trained on Toronto Face Dataset (TFD) [19] where images are labeled based on expressions (anger, disgust, fear, happy, sad, surprise, and neutral), it is possible to generate non-existing images of the same person with different expressions by feeding a fixed value of z and different values of y to the decoder (See Fig. 2.2 for examples). Similarly, the method is able to generate face images of the same person with different camera poses when trained on Multi-PIE face dataset [20]. Neural Statistician [21] extends the VAE framework by including an extra latent variable c in the data likelihood. Latent variable c represents the context, and is different among different datasets while it is the same for the data points within a dataset. Dataset in this study is defined as an unordered collection of data points such as photos of a particular person. The main idea behind this study is to work with datasets rather than data points, and for machines to understand the similarities among datasets. The model is able to generate diverse unseen images of a particular identity when trained on the YouTube Faces dataset. Conditional Subspace VAE (CSVAE) [22] minimizes the mutual information between the latent variable z and class label variable y. As a result, CSVAE is able to extract features correlated to binary labels and structure them in a latent subspace. Multi-Level Variational Autoencoder (ML-VAE) [23] learns a representation that disentangle the identity related information from other information in a face image. The model is trained using a set of face images grouped by identity and with the assumption that the samples within a group share a common factor of variation.

There are variations of autoencoders that approach the problem of disentangling concepts in the latent space in a fully unsupervised way. These models are usually developed based on adding structure into the latent space. For example β-VAE [24] integrates an adjustable hyperparameter β into VAEs framework which enforces conditional independence of the latent variable z. When $\beta = 1$, the model is the same

as VAE, however when $\beta > 1$, it applies a constraint on z and limits its representation capacity which encourages the disentanglement. Another example is Total Correlation Variational Autoencoder or β-TCVAE [25] which decomposes the variational lower bound on the marginal likelihood of training data in order to include a term measuring the total correlation between latent variables. They show that there is a strong relation between total correlation and disentanglement. Therefore, the model is able to learn disentangled representations, where unlike β-VAE no additional hyperparameter is required to be adjusted during training. Autoencoding VAE (AVAE) [26] argues that the inconsistency between the decoder and encoder in standard VAEs prevents them from achieving properties such as adversarial robustness or disentanglement in their representations. More precisely, the standard VAE loss does not enforce samples generated by the decoder of a VAE to be mapped to the corresponding representations by the encoder. To address this issue, AVAE adds an additional constraint on the VAEs distribution approximation process. This constraint is based on a new lower bound of the true marginal likelihood. The main idea of AVAE is making the encoder and the decoder to be consistent both on the training data and on the samples generated by the decoder.

Similar to other methods of data distribution estimation, autoencoders and VAEs possess their strengths and weaknesses. Autoencoders are extremely effective in learning the hidden factors of variations. They can learn semantic concepts in a face in an unsupervised manner. However, finding ways for them to learn latent representations in which different meaningful factors of variations are nicely detached and also learning latent representations for one specific semantic concept only, in a fully unsupervised manner, are still challenging research problems. Furthermore, autoencoders and VAEs maximize a lower bound on data likelihood. As a result, they do not perform as well as generative models which maximize the data likelihood itself. That is why the samples generated by autoencoders are blurrier and lower quality

in comparison to other generative models. Improving the quality of VAEs generated samples is an open problem and actively being researched.

2.2.2 Generative Adversarial Networks

Generative Adversarial Networks (GANs) introduced by Goodfellow et al. [13] is a notable method for generating images using deep neural networks. Instead of explicitly modeling and solving the dataset density function, GANs focus on generating samples from this distribution. GANs' main idea is to first sample from a known latent distribution $p(z)$ and then learn to transform this sample into a sample of training distribution. Two networks are involved in a GAN model; discriminator network and generator network. The discriminator network receives examples and estimates the probability of whether the sample is real or fake. The generator network receives a sample from latent variable's true prior distribution $p(z)$ and transforms it into a sample from the distribution of training set. GANs are trained through a two-step training process. The first step is to train discriminator, which requires feeding the discriminator with the real training samples and the fake samples in order to learn to distinguish between them. The second step is to train the generator network to deceive the discriminator network. The training process of GANs constantly alternates between these two steps which can be interpreted as the two networks playing a game; The discriminator network tries to distinguish correctly between real and fake samples. Meanwhile, the generator network tries to fool the discriminator into thinking that its output is real. Through game theory analysis techniques, it can be proven that this game will eventually converge to an equilibrium state. In that state, the samples generated by generator are identical to real training data and discriminator assigns probability of $\frac{1}{2}$ to every input regardless of whether it is real or fake.

Since their introduction, GANs have gained lots of attention among researchers and many extensions and variations of GANs have been developed. For instance,

Conditional GAN (cGAN) [27] is a variation of GAN in which there is an additional condition such as class label on both the discriminator and the generator causing the generator to only generate new samples of a specific class. In [28] incorporation of conditional information into GAN framework specifically for face generation is examined more formally. More precisely, some arbitrary conditional information is provided to the GAN in order to generate faces with specific set of attributes. By controlling this conditional information, it is possible to control attributes in the generated face. The additional information in fact describes the image which is being generated. Age-cGAN [29] proposes an automatic face aging method based on cGAN. In Age-cGAN, the generator is conditioned on both the latent space vector approximation of input image (identity) and the target age category. The training dataset contains at least 5000 examples in each of the following six age categories: 0-18, 19-29, 30-39, 40-49, 50-59, and 60+ years old. This method emphasizes on preserving the identity of the subject by introducing an identity preserving optimization.

Coupled GANs (CoGAN) [30] is a variation of GANs aiming to learn a joint distribution of multi-domain images. The method can successfully generate pairs of corresponding images in two different domains. This is achieved by applying a weight-sharing constraint to the layers of the generator network whose role is to decode abstract semantics. This weight-sharing constraint puts a limitation on network capacity and converges to a solution for a joint distribution over a product of marginal distributions. The method does not require the training set to contain corresponding images in different domains and only uses random samples drawn separately from the marginal distributions. CoGAN is shown to be successful in learning a joint distribution of face images of different attributes domains such as blond-hair, smiling, and eyeglasses attributes (See Fig. 2.3 for examples). The concept of discovering relations between different domains and performing Image-to-image translation between domains is approached in DiscoGAN [31], CycleGAN [32], DualGAN [33], and

XGAN [34], as well. In these models, two different GANs are coupled together to transform an input image from one domain to another. Each of the GANs is responsible to make sure that the generative functions can map each domain to the other domain. In other words, assuming that there are two domains A and B, generator G_{AB} maps images from domain A to domain B. Meanwhile, generator G_{BA} maps images from domain B to domain A.

In pix2pix [35] the task of image-to-image translation is approached using a conditional GAN. The training of pix2pix requires the images from the two domain to be paired. The generator network learns to transform an input image from domain A into its corresponding version in domain B. The discriminator learns to distinguish between the real pairs and the pairs synthesised by the generator. Pix2pixHD [36] extends the previous work to generate high-resolution realistic face images from very simple edge sketches and is also able to edit the facial attributes such as changing the skin colour or adding beards. ELEGANT [37] proposes to exchange attributes between two faces by exchanging certain part of their latent codes. They train the model for each particular attribute by feeding it with a pair of images with opposite attribute. Their training iteratively goes over all attributes repeatedly. StarGAN [38] proposes a method for performing multi-domain image translation using a single network conditioned on the target domain label. Instead of learning the translation from only one pair of domains, their model learns the mappings between all available domains in training data.

Semantic segmentation masks (in which every pixel of an image is labeled according to what they show) are helpful tools that have been used for disentangling concepts and for allowing local control over synthesised images. SPADE [39] introduces an architecture modification for converting a semantic segmentation mask into a photorealistic image. They propose to use a spatially-adaptive normalization layer that can propagate the semantic segmentation mask information throughout

Figure 2.3: Examples of corresponding face pairs in different attribute domains generated by CoGAN reprinted from [30] with permission.

the network. In their architecture the activations of layers throughout the network are modulated using the input semantic segmentation mask via a spatially-adaptive learned transformation. MaskGAN [40] lets the user edit a semantic segmentation mask. This manipulated mask is then used as an intermediate representation for performing flexible face manipulations. MaskGAN consists of a Dense Mapping Network (DMN) that learns to map a user modified semantic segmentation mask into a realistic target manipulated face image.

Image-to-image translation models, such as pix2pixHD, inspired the development of several image and video face-swap models. For example, GANimation [41], which uses facial Action Units (AUs) annotations for GANs conditioning, is able to animate an input image by generating novel continuous facial expressions. GANnotation [42] introduces a triple consistency loss for transferring both the pose an expression from a source image to a target image. In [43] a recurrent neural network is trained using many hours of Obamas speeches to learn the mapping from raw audio features to mouth shapes. Once trained, it is able to receive an audio file of Obama and generate a video of him speaking. The videos generated by this model are of high quality and display accurate lip-sync with the audio file. Deep video portraits [44] uses a generative neural network with a novel architecture and careful adversarial training to transfer not only facial expressions but also the head position, head rotation, face expression, eye gaze, and eye blinking from a source person to a portrait video of a target person. Region-Separative GAN (RSGAN) [45] learns to separate latent representations for face and hair and then performs face swap by reconstructing face images from modified latent representations generated by replacing the face and hair representations in the two original latent representations. Most of these models must be trained for each subject or pair of subjects and so require expensive subject-specific data.

There are variations of GANs that attempt to disentangle concepts from images

in a fully unsupervised manner. For instance, Information Maximizing GANs (InfoGAN) [46] is an information theoretic extension of the GANs that tries to extract meaningful and interpretable representations from completely unlabeled data. The method is based on the idea that the latent representation of GAN is not entirely unstructured noise and can be decomposed into different variables (or subset of variables) to force displaying meaningful visual concepts in the generated image. Introducing new structured subsets in latent vector z and training InfoGAN to generate samples by maximizing the mutual information between a subset of latent variables z and the generated sample, resulted in each of the structured subsets in vector z to learn to represent a meaningful semantic in image. For example the method was able to discover meaningful visual concepts such as hairstyle, presence/absence of eyeglasses, and emotions in face images in a fully unsupervised manner. Another important group of GANs variations that attempt to disentangle concepts from images in a fully unsupervised manner are the ones that add structure and hierarchy to the training of GANs. For instance, LAPGAN [47] decomposes the image into a set of band-pass images with a separate GANs model at each level of the pyramid for generating images in a coarse-to-fine manner. SGAN [48] proposes a top-down stack of GANs in which the GANs model at each level is conditioned on a higher-level representation. The hierarchy and structure proposed in most methods in this group is of a coarse-to-fine type. It means they first build the coarse structure for the whole image and then add more details to this coarse structure. Two very important methods in this group that dramatically improved the quality, realism, and variations of the generated faces are styleGAN [11] and its predecessor Progressive Growing of GANs (PGGAN) [49].

The main idea behind PGGAN is to begin with smaller networks for the generator and discriminator and then grow their size progressively by adding layers during training. The initial networks work with low-resolution 4×4 images. As the networks

grow larger in size, the resolution of the images increases gradually as well. This approach speeds up the training of GANs, makes it more stable, and therefore makes it possible for GANs to generate realistic-looking face images of an unprecedented quality and resolution of 1024×1024. Furthermore, it is shown that the gradual increase in the size of the networks and the resolution of the images adds some structure in the training by forcing the network to first learn large-scale concepts of the image distribution and then gradually shift toward learning finer scale details once the larger concepts are tackled. Examples of generated faces by PGGAN are shown in Fig. 2.4.

Figure 2.4: Examples of face images generated by PGGAN [49]. Reprinted from Progressive Growing of GANs source code [50] copyrighted © 2018, NVIDIA CORPORATION under a Creative Commons licence (Creative Commons disclaims all liability for damages resulting from their use to the fullest extent possible).

StyleGAN model builds on the PGGAN by modifying the generator network with inspirations from the style transfer framework [51]. More precisely, the discriminator network is the same as in PGGAN, but the generator network is composed of two

distinct networks; the mapping network and the synthesis network. The mapping network is made up of 8 fully-connected layers and receives the latent variable z and transforms it into an intermediate latent variable w. The synthesis network receives a constant as input of the first layer and w is integrated into its layers through learned affine transformations that transform w into styles $y = (y_s, y_b)$ that are then used to control Adaptive Instance Normalization (AdaIN) operations after each convolution layer. Additionally, a dedicated single-channel noise image is fed to each layer of the synthesis network that makes it possible for it to directly generate the stochastic variations in a face image such as the exact arrangement of hairs, freckles, or skin pores. By integrating distinct style inputs into the different layers of the synthesis network, this architecture makes it possible to control the generated faces at different scales. For example, face images generated by mixing two latent codes w at various scales display meaningful high-level attributes from the two source images. More precisely, style mixing shows that the styles corresponding to lower resolutions (i.e. 4×4 and 8×8) control high-level concepts such as head pose, face shape, and eyeglasses, while styles corresponding to middle resolutions (i.e. 16×16 and 32×32) control smaller scale facial features such as hair style, and eyes open/closed. Finally, styles corresponding to higher resolutions (i.e. 64×64 to 1024×1024) mainly control the color schemes and microstructures. Examples of such style mixings by styleGAN model are shown in Fig. 2.5.

Lastly, there are variations of GANs that attempt to identify concepts learned in the activations of the intermediate layers of a pre-trained GAN generator. As a result, it would be possible to change these activations in order to modify the output image in a desired way. Bau et al. [53] use a segmentation-based network dissection method in order to identify a group of activation units that are closely related to object concepts. By zeroing these activations, it is possible to remove that object from the image. Similarly, it is also possible to insert these object concepts into

Figure 2.5: Examples of face images generated by StyleGAN model [11]. The images in the first row and the first column are used as the source of styles for performing style mixing. Reproduced using StyleGAN source code [52] copyrighted © 2019, NVIDIA CORPORATION under a Creative Commons licence (Creative Commons disclaims all liability for damages resulting from their use to the fullest extent possible).

images. Editing in Style [54] performs clustering on hidden layers activations of the generator network of StyleGAN [11]. This clustering results in the disentanglement of semantic objects and makes it possible to perform local semantically-aware edits to GAN generated human faces. Latent Regression [55] combines a regressor network and a pre-trained generator network in order to generate composite realistic images from a collage of random image parts. The regressor network is trained to predict the latent vector from an image. It can receives a rough, incoherent template of the scene and predicts the latent vector for it. The generator network then receives this

predicted latent vector as input and synthesises a realistic coherent image.

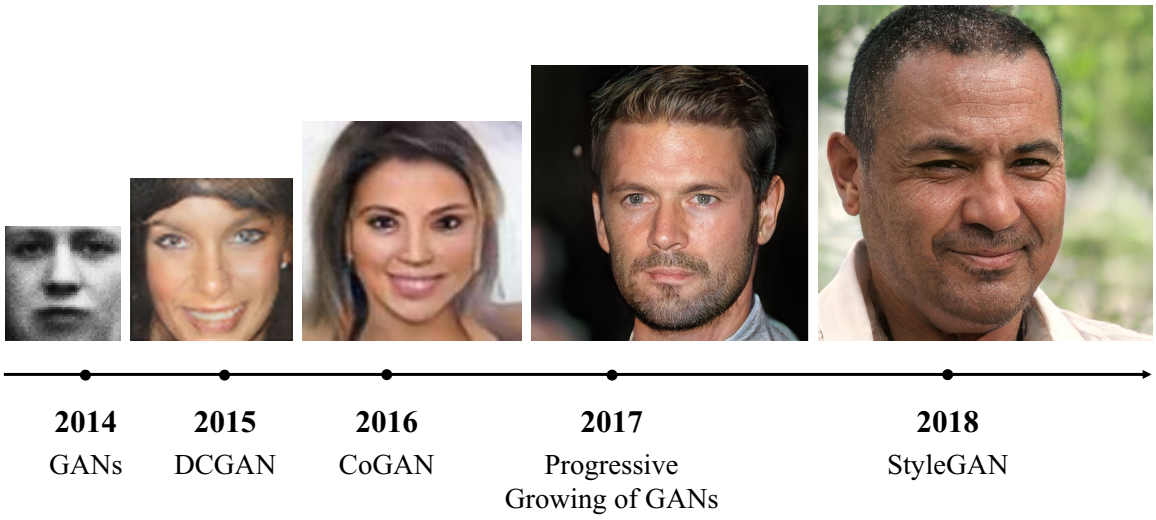


Figure 2.6: The progress in generating face images of non-existing identities using GAN-base models in a nearly 5-year period adapted from [56]. The images from left to right are respectively reprinted from [13] with author's permission, [57] with author's permission, [30] with author's permission, [50] © 2019, NVIDIA CORPORATION, and [52] © 2019, NVIDIA CORPORATION.

In summary, the GANs framework and its extensions have caused revolutionary improvements in face synthesis and semantic manipulations leading to state-of-the-art results in generating non-existing face images. Moreover, they are able to generate diverse realistic face images, with high visual quality at high resolutions while allowing for control over the face synthesis process in meaningful ways. The amount of attention that the GAN models have gained among researchers since its first introduction in 2014 is very noticeable. Many variations of GANs have been developed bringing about rapid improvements to the performance of GAN models. Adapted from [56], an illustration of the impressive progress in synthesising faces of non-existing identities using GAN models during a 5-year period is displayed in Fig. 2.6. Other than the studies reviewed in this chapter, numerous variations of the GAN framework were

introduced during the past few years that were not focused on face images. These extensions attempt to improve the quality of generated results, make the training process more stable, define better loss functions, use GANs for a variety of applications, or examine new ideas using GAN framework.

2.2.3 Hybrid Deep Generative Models

A powerful trend in deep generative models is combining autoencoders with GANs in one framework. This allows for taking advantage of the benefits of the two methods while avoiding their drawbacks. This group of methods will be referred to as hybrid deep generative models and this section will review notable models of this group applied to the problem of face understanding.

Boundary Equilibrium Generative Adversarial Networks (BEGAN) [58] is an autoencoder-based GANs framework that uses an autoencoder as the discriminator. It means that the discriminator extracts a latent representation from input images using an encoder network and then reconstructs the input images from the latent feature representations using a decoder network. The discriminator is trained to minimize a reconstruction loss. More precisely, the reconstruction loss is high for the generated images that are not realistic looking, and it is low for the generated images that look realistic. Additionally, an improved training of GANs is achieved by balancing generator and discriminator during training by adding an equilibrium term to the original GAN objective function. BEGAN is able to generate diverse realistic face images with high visual quality (See Fig. 2.7 for examples). Adversarially Learned Inference (ALI) [59] trains a generator and an inference network together to fool the discriminator network. The generator maps samples from latent space to data space. The inference network maps training examples from data space to latent space. These two networks play an adversarial game. A discriminator network is also trained to distinguish between joint latent-data samples from generator and

joint latent-data samples from inference network. By integrating inference of latent space variables into the GANs framework, this framework is able to generate face images of high visual quality by interpolation between two face images, or editing facial attributes/expressions by adjusting the latent space variables.

Figure 2.7: BEGAN generated faces by interpolating between two real faces images. Reprinted from [58] with permission.

In VAE/GAN [60], the learned features by the discriminator of GAN are used as the basis for defining VAE's reconstruction objective. The intuitive idea is that what the discriminator learns through distinguishing between real and fake data can be used for measuring similarity between images. It is in fact replacing pixel-wise reconstruction loss in VAE framework with an error that is more capable of capturing data distribution. Results show that the method is able to improve the visual quality of generated images in comparison to standard VAEs. Additionally, it is shown that the method achieves some level of attribute disentanglement in the latent representation space. They compute latent vector representations for all images, and then for each attribute they compute the mean vector for faces with the attribute and the mean vector for faces without the attribute. The difference between the two mean vectors is called visual attribute vector. By moving a representation vector in the direction of visual attribute vector and then reconstructing it using the decoder, the face images with transformed facial attribute are generated as shown in Fig. 2.8. Introspective

Adversarial Network (IAN) [61] is another hybrid model with a similar architecture to VAE/GAN (i.e. the decoder network of VAE as the generator network of the GAN). Nevertheless, instead of having a separate discriminator network, the encoder and discriminator are combined in a single network. The objective function of IAN consists of three terms; pixel-wise reconstruction loss, feature-wise reconstruction loss, and adversarial loss. IAN was able to perform realistic and meaningful editing on face images such as changing hair color and adding facial hair.

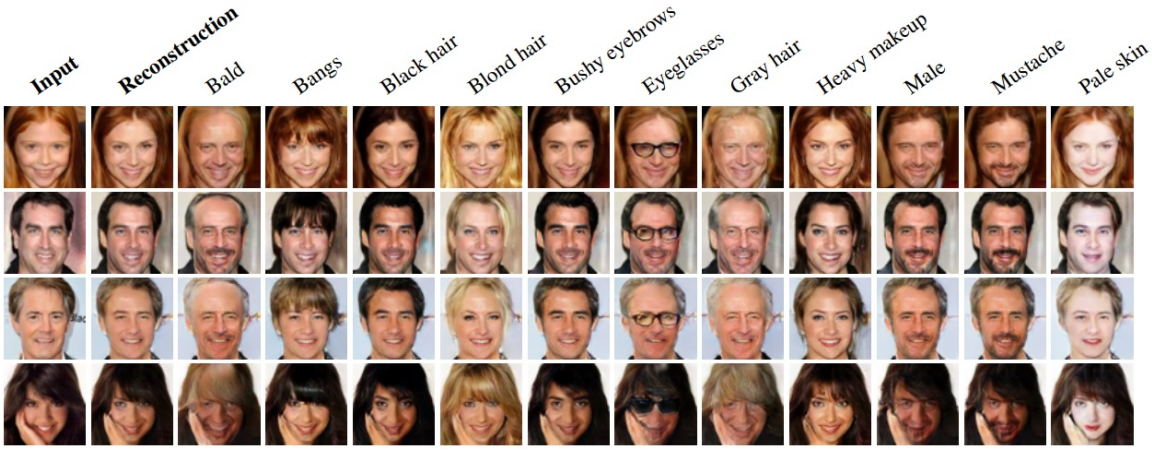


Figure 2.8: Examples of facial attributes transformations by VAE/GAN reprinted from [60] with permission.

UNsupervised Image-to-image Translation (UNIT) [62] is another hybrid model based on VAE and GANs that learns the mapping for images from one domain to another in a fully unsupervised manner. The method learns the two-way translation function between two image domains without any corresponding pair of images in those two domains. The training dataset only consists of two subsets of images from each domain. First an image fidelity function is defined for each of the two domains using an adversarial training objective. Then, the adversarial training objective is used to generate corresponding images in those two domains through a weight-sharing constrain. Meanwhile, VAE's role is to relate translated images in the target domain

with input images in the source domain. UNIT was also able to generate face images of high visual quality by transforming face images between different attribute domains (See Fig. 2.9 for examples). Another example of hybrid models is Invertible cGAN (IcGAN) [63] which introduces an encoder into the cGAN framework. The encoder reverses the cGANs' workflow; it compresses a real image x into a latent representation z and a conditional vector y. This process enables the IcGAN to re-generate real images. As a result, the model was able to perform meaningful transformations on face images by arbitrarily changing the conditional information y.

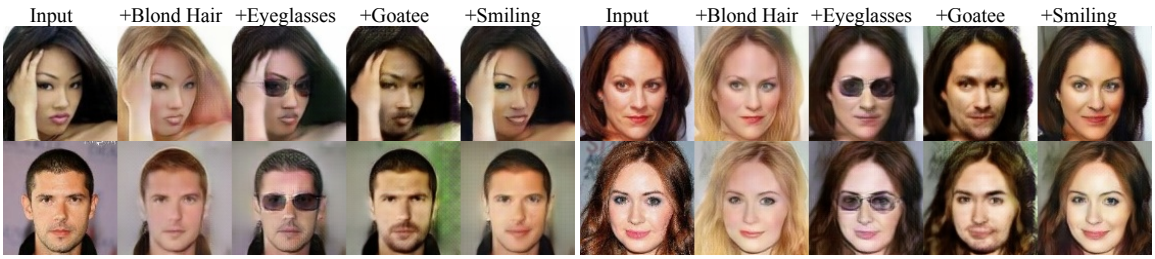


Figure 2.9: Examples of face image translations between attribute domains by UNIT reprinted from [62] with permission.

Progressively Growing Generative Autoencoder (PIONEER) [64] proposes an autoencoder model containing of only two networks, an encoder and a decoder/generator. The architecture of both the encoder and the decoder grows progressively and synchronously. The model is trained using a balanced combination of reconstruction loss and adversarial loss. The reconstruction loss makes it possible for a training sample to get encoded into the latent space and then decoded back into a generated sample. The model does not include a discriminator network and the adversarial loss is implemented by applying a reconstruction loss on the latent vectors. It means that a random latent vector is encoded into a random generated sample that is then fed back to the encoder to generate the reconstructed latent vector. Balanced PIONEER [65] builds on PIONEER model using new normalization schemes

to modify the training dynamics of the model. These modifications result in the improvements of the results both in terms of faithful reconstruction and in terms of fully unsupervised disentanglement of concepts in the latent space. Style adversarial latent autoencoders (StyleALAE) [66] is proposed inspired by styleGAN. StyleALAE introduces a novel GAN-Autoencoder architecture which consists of four networks; $F : z \rightarrow w$, $G : w \rightarrow x$, $E : x \rightarrow w$, and $D : w \rightarrow \mathbb{R}$. Both an adversarial loss and a reconstruction loss are applied to w instead of x and the training of the model alternates between these two optimization processes. The reconstruction loss enforces that the latent spaces at the interface between networks F and G and between networks E and D are the same. The adversarial loss enforces that the distribution of w resulted from real images and the one resulted from synthetic image are the same. Introspective Variational Autoencoder (IntroVAE) [67] introduces the idea of training a VAE adversarially. It means training the encoder network of VAE to discriminate between real data samples and generated ones. Soft-IntroVAE [68] modifies IntroVAE by replacing its hinge-loss terms with a smooth exponential loss on generated samples. The modifications result in increased training stability. Soft-IntroVAE is able to generate faithful and high-quality reconstructions and also disentangle class and content in its learned representations.

To summarize, it can be concluded that hybrid deep generative models are a promising group of deep generative models for understanding face images and for disentangling facial concepts. They take advantage of the bests in autoencoders and GANs, while compensating for their drawbacks. As a result, they are able to achieve various types of disentanglements in their latent representation space.

2.3 Summary

Deep generative models are a group of unsupervised methods that try to understand and extract the structures buried in unlabeled examples. Among them, autoencoders are specifically successful in inferring an interpretable latent representation that captures the global structure of a natural image. Therefore, they are very useful in extracting high-level concepts from face images. GANs are another important group of deep generative models. They are powerful tools to model data distributions implicitly. Their sampling process is fast and exact and results in generating high quality samples. Some extensions of GANs framework achieve state-of-the-art results in generating realistic face images of non-existing identities. Additionally, GANs extensions are shown to be promising for learning and disentangling meaningful facial concepts by adding structure and hierarchy to the data distribution learning process or learning multi-scale representations. Lastly, hybrid deep generative models are the group of deep generative models that take advantage of the benefits of both autoencoders and GANs, and tackle their disadvantages by including more than one generative model in their framework. The flexibility gained by such approach makes them a very promising group of models as they have already resulted in notable results.

Chapter 3

An Autoencoder with Adaptive Resolution (AR) Reconstruction Loss

In this chapter we propose a deep autoencoder model that takes advantage of an adaptive resolution reconstruction loss. The adaptive resolution reconstruction loss is inspired by the fact that different categories of concepts are encoded in (and can be captured from) different resolutions of an image. We propose that it is possible to control the coarser concepts in a generated face image by enforcing a reconstruction loss on only the lower-resolution versions of that image. This new type of reconstruction loss facilitates learning a latent representation for real face images in which facial concepts are disentangled based on scale. As a result, in addition to reconstructing input face images faithfully, the autoencoder is able to generate realistic reconstructed images associated with a combination of latent vectors from different sources. In other words, it is possible to modify a given image in meaningful and controlled ways by feeding a mix of its representations and the representations of other real face images to the decoder network.

3.1 Methodology

3.1.1 Background: Standard Autoencoders

As it was discussed in Chapter 2, autoencoders are a data dimensionality reduction approach that have long been implemented using neural networks [12]. An autoencoder consists of two networks; The encoder and the decoder. The encoder network learns a mapping from data space to a latent space ($\zeta : \mathcal{X} \rightarrow \mathcal{Z}$). The decoder network, which has the reverse/opposite architecture to the encoder network, learns the mapping from the latent representation space back to the data space ($\eta : \mathcal{Z} \rightarrow \mathcal{X}$). The objective of the training process for autoencoders is to create latent representations that are able to reconstruct the original input data. Therefore, the autoencoder loss function is defined to force the reconstructed data to be similar to the training data $\mathcal{X} = \{x_1, ..., x_N\}$ as follows

$$\mathcal{L}(x_n) = ||x_n - \eta(\zeta(x_n))||^2, \tag{3.1}$$

and the autoencoders objective function is defined to minimize the loss function over the entire training dataset

$$\min_{\zeta,\eta} \sum_{n=1}^{N} \mathcal{L}(x_n) = \min_{\zeta,\eta} \sum_{n=1}^{N} ||x_n - \eta(\zeta(x_n))||^2. \tag{3.2}$$

After the training process is completed, the encoder network is used to compute the latent representations and the decoder network is used to generate images corresponding to samples from the feature representation space. Autoencoders are able to reconstruct the training data and consequently make it possible to examine and explore the representation space through performing tasks such as interpolations between datapoints in the representation space or performing meaningful modifications

of datapoints in the representation space.

3.1.2 Adaptive Resolution (AR) Reconstruction Loss

We introduce a regularization method to be added to the standard autoencoder framework in order to achieve unsupervised disentanglement of concepts based on scale in the latent representations. This regularization is inspired by the fact that different categories of concepts are encoded in (and can be captured from) different resolutions of an image [49]. For example, as it is illustrated in Figure 3.1, information about the overall structure of a face image including head pose, hairstyle and face shape are encoded in lower resolutions (i.e. 4×4 and 8×8), while information related to identity and expressions are added to the overall structure by slightly higher resolutions (i.e. 16×16 and 32×32). Finally, higher resolutions (i.e. 64×64 and above) of face images only add more details to the image. Given this, we propose that it is possible to control the coarser concepts in a generated face image by enforcing a reconstruction loss on only the lower-resolution versions of that image.

Instead of learning the transformation $\zeta : \mathcal{X} \rightarrow \mathcal{Z}$, the encoder network of our model learns a transformation $\delta : \mathcal{X} \rightarrow \mathcal{W}$. The latent space $\mathcal{W}$ is the latent space in which scale-based concepts are separated. Similar to StyleGAN [11] model, we call the latent space representations $w \in \mathcal{W}$ the style vectors. Subsequently, the decoder network of our model learns a transformation $\phi : \mathcal{W} \rightarrow \mathcal{X}$. Furthermore, the decoder network takes advantage of an architecture similar to the synthesis network of StyleGAN model with an injection of Gaussian noise followed by an AdaIN [69] layer after each convolutional block. More precisely, the decoder architecture receives a constant as the input of the first layer and the latent representation w is integrated into its layers through AdaIN operations after each convolution layer. An AdaIN layer aligns the mean and variance of the content features with those of the style features. More specifically, an AdaIN receives a content input x and a style input y,

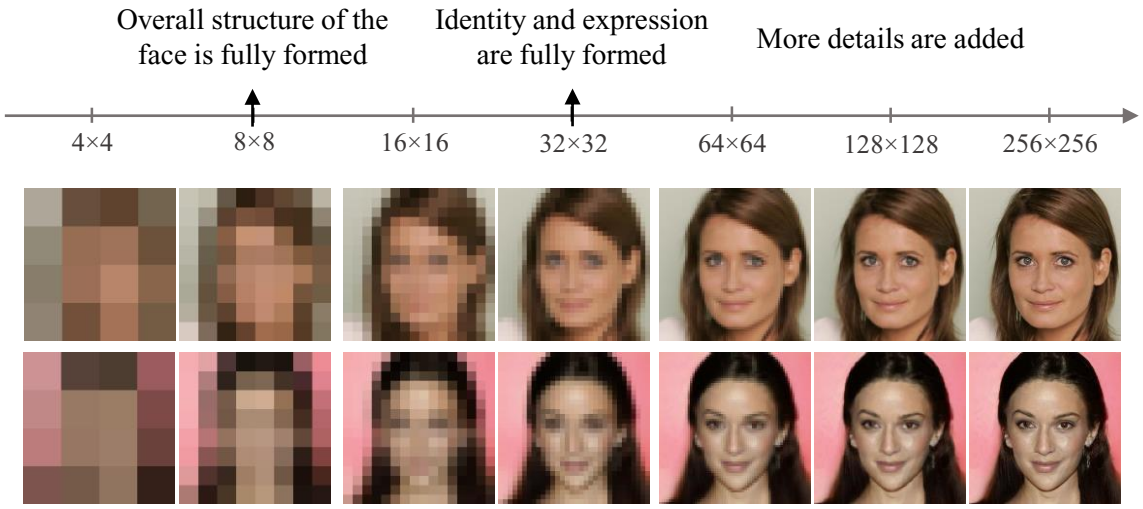


Figure 3.1: An illustration showing that information about the overall structure of a face image are encoded in 4×4 and 8×8 resolutions, information related to identity and expressions are added to the overall structure by 16×16 and 32×32 resolutions, and higher resolutions (i.e. 64×64 and above) only add more details to the image. The displayed faces are samples from CelebA-HQ [49] dataset.

and aligns the channel-wise mean and variance of x to match those of y using the formula below

$$AdaIN(x, y) = \sigma(y)(\frac{x - \mu(x)}{\sigma(x)}) + \mu(y). \tag{3.3}$$

Our model requires that two sets of real images x_1 and x_2 pass through the encoder network in each step of the training and the $w_1 = \delta(x_1)$ and $w_2 = \delta(x_2)$ associated with them are computed. Similarly, two different sets of latent representations pass through the decoder network in every step of the training; one without mixing and one with mixing. The latent representation with mixing $w_{mixed} = mix(w_1, w_2, p)$ is equal to w_1 before a randomly selected splitting point p and is equal to w_2 after it. The splitting point p is sampled from a discrete uniform distribution $p \sim U[1, N_l]$ where N_l is equal to the number of convolutional layers of the decoder network. In

every step of the training, the latent vector generated without mixing is responsible for reconstructing the input images faithfully enforced by a standard reconstruction loss similar to the loss function defined in Equation 3.1, while the one generated with mixing is responsible for making sure that different scales of concepts in a face image are separated in the learned latent vectors enforced by AR reconstruction loss which will be explained in the following.

By feeding $w_{mixed} = mix(w_1, w_2, p)$ to the decoder network, we expect that the output images will combine the concepts from x_1 and x_2. We introduced AR reconstruction loss to minimize the difference between the lower resolution version of $x_{mixed} = \phi(\delta(w_{mixed}))$ and the lower resolution version of the input image used as the source for the first part of w_{mixed} (i.e. before the random splitting point p). The second reconstruction loss term is defined as

$$\mathcal{L}_{AR}(x_1, x_2, p) = \\ ||[x_1]_{r\times r} - [\phi(mix(\delta(x_1), \delta(x_2), p))]_{r\times r}||^2 \tag{3.4}$$

where $r \times r$ is a lower resolution selected according to the one-to-one correspondence between splitting points $p \in \{1, 2, ..., N_l\}$ and resolutions $r \in \{4, 8, ..., 2^{N_l+1}\}$ (N_l is equal to the number of convolutional layers of the decoder network). This reconstruction loss can be considered as a type of structure preserving regularizer while allowing for the higher level details to change. Please note that by sampling a splitting point from a discrete uniform distribution $p \sim U[1, N_l]$ and adapting the resolution of the images accordingly to be given to $\mathcal{L}_{AR}$, we make sure that the model learns to separate the facial concepts based on scale/image-resolution in its latent representations. The alternative way of assuring that the same is learned by our model would be to have N_l different w^p_{mixed} for each $p \in \{1, .., N_l\}$ and N_l different reconstruction errors for each resolution $r \in \{4, 8, ..., 2^{N_l+1}\}$ in every training step. However, this alternative

way is not computationally feasible and therefore the adapting resolution approach is introduced.

Lastly, the objective function for our model is defined to minimize the sum of the reconstruction loss for images generated without mixing w (similar to the loss function defined in Equation 3.1) and the lower-resolution reconstruction loss for images generated with mixing w (defined in Equation 3.4) as follows

$$\min_{\delta,\phi} [\sum_{i=1}^{N} \mathcal{L}(x_i) + \sum_{i \neq j}^{N} \mathcal{L}_{AR}(x_i, x_j, p)\,], \tag{3.5}$$

$$\begin{aligned} \min_{\delta,\phi} [\ & \sum_{i=1}^{N} ||x_i - \phi(\delta(x_i))||^2 \\ & + \sum_{i \neq j}^{N} |||[x_1]_{r\times r} - [\phi(mixs(\delta(x_1), \delta(x_2), p))]_{r\times r}||^2\,]. \end{aligned} \tag{3.6}$$

Algorithm 1 Calculating reconstruction loss in every epoch of the training

```
mixing = true, L = 0
for i := 1 to batches_per_epoch do
    Load x_1, x_2
    w_1, w_mixed ← δ(x_1)
    w_2 ← δ(x_2)
    if mixing then
        Sample p from U[1, ..., N_l]
        w_mixed[p : N_l] ← w_2[p : N_l]
        r ← 2^{p+1}
    end if
    x̂_1 ← φ(w_1)
    x̂_mixed ← φ(w_mixed)
    L += ||x_1 − x̂_1||^2
    [x_1]_{r×r}, [x̂_mixed]_{r×r} ← x_1, x̂_mixed resized to r × r
    L += |||[x_1]_{r×r} − [x̂_mixed]_{r×r}||^2
end for
return L
```

This process is summarized in Algorithm 1. Additionally, Figure 3.2 illustrates how the first convolutional block of the decoder network is associated with the 4×4 resolution of the reconstructed output, the second convolutional block is associated with the 8×8 resolution of the reconstructed output, and so on. In Figure 3.2 (a), the reconstruction loss is defined for the highest resolution of images since the style vectors w all come from the same source image. This results in the output image to be a faithful reconstruction of the source image. In Figure 3.2 (b), the AR reconstruction loss is defined for lower resolutions of images since the style vectors w come from two different source images. This lower resolution $r = 2^{p+1}$ is different in every step of the training since p is randomly selected in every step of the training. However, the figure only shows two example cases; $p = 2$ and $p = 4$. As shown in Figure 3.2 (b), the AR reconstruction loss results in the output image to display a combination of concepts from both source images.

3.1.3 Weighted Adversarial Loss

In order to reduce the blurriness associated with autoencoder generated images, a discriminator network along with an adversarial loss is included in the model as well [70]. The discriminator network receives the real data samples and also the samples synthesised by the decoder network and estimates the probability of whether they are real or fake via a score ($D : \mathcal{X} \to \mathbb{R}$). The adversarial loss simultaneously trains the discriminator network to distinguish correctly between real and fake samples and trains the decoder network to fool the discriminator into thinking that its outputs are real using the following objective function

$$\min_{\phi} \max_{D} V(D, \phi) = \mathbb{E}_{x \sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{x \sim p_{data}(x)}[\log(1 - D(\phi(\delta(x))))]. \tag{3.7}$$

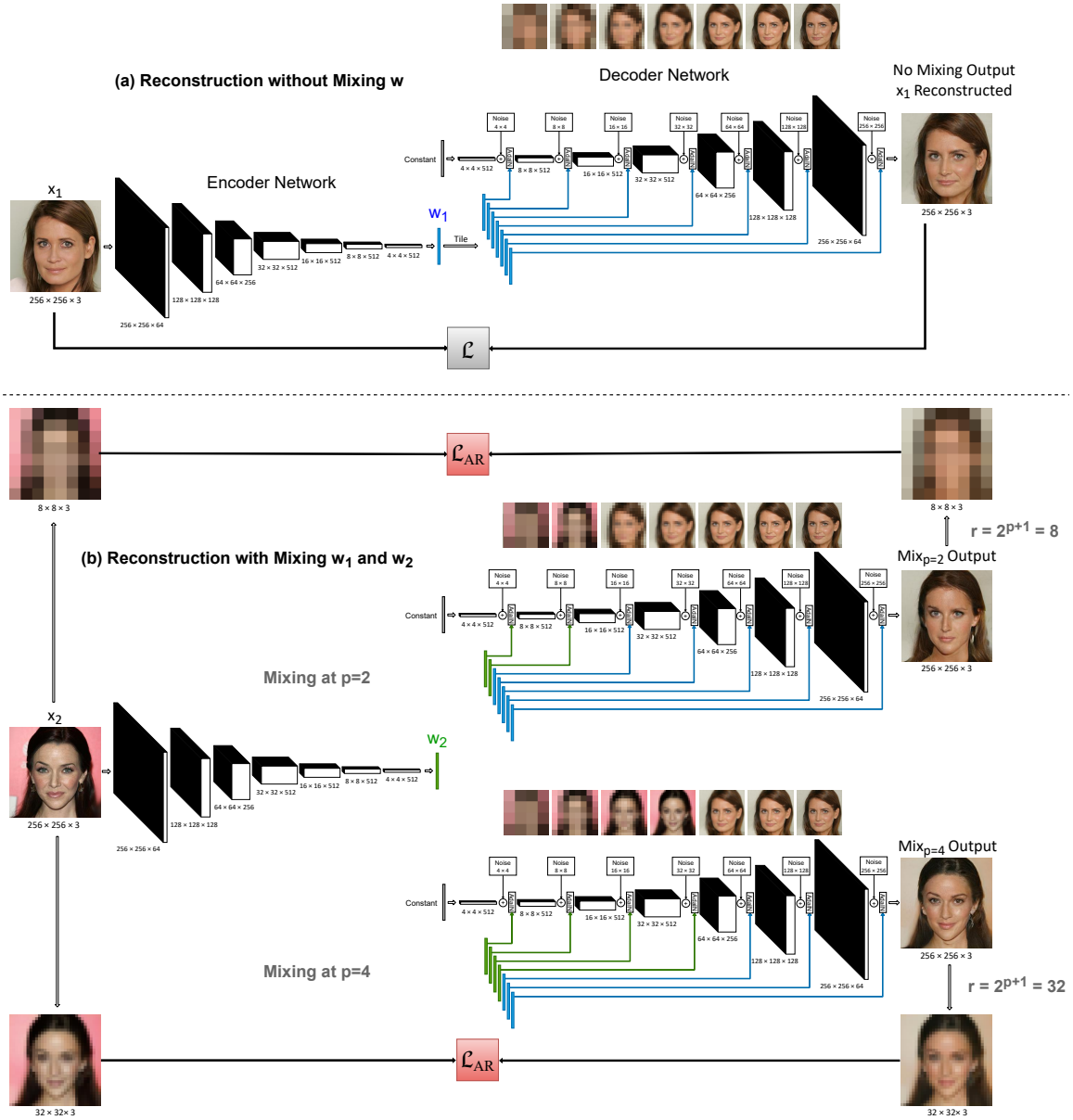


Figure 3.2: A schematic view of the proposed autoencoder model. (a) Similar to a standard autoencoder, the reconstruction loss is defined for the highest resolution of images when the style vectors w all come from the same source image. (b) The AR reconstruction loss is defined for lower resolutions of images when the style vectors w come from two different source images. The displayed faces are samples from CelebA-HQ [49] dataset.

However, the adversarial loss is included in the model with a small weight for the sole purpose of improving the quality of reconstructed images and does not dominate the image generation process. In later sections, we will provide evidence that the inclusion of the discriminator network and the adversarial loss in this model does not play a role in the disentanglement of concepts and only improves the quality of the reconstructed images. Therefore, the final objective function for our model is

$$\min_{\delta,\phi} [\sum_{i=1}^{N} \mathcal{L}(x_i) + \sum_{i \neq j}^{N} \mathcal{L}_{AR}(x_i, x_j, p)] + \min_{\phi} \max_{D} [\gamma \times V(D, \phi)] \tag{3.8}$$

where γ is the weight of the adversarial loss. Please note that γ is not a learnable parameter and the best fixed value for it is found through several rounds of experimentation further explained in the section on quantitative analysis of results.

3.2 Experiments and Setup

3.2.1 Architecture

As it is displayed in Table 3.1, the encoder network of the proposed autoencoder model is composed of 7 convolutional blocks when the input color image is of size 256×256. The encoder network transforms the color images into latent style vectors of size 512. Each Convolutional block of the encoder network consists of two convolutional layers followed by a down-sampling layer. The activation function used for every layer is *Leaky-Relu* with $\alpha = 0.2$. Similarly, as shown in Table 3.2, the decoder network architecture is composed of 7 convolutional blocks when the input color image is of size 256×256. The decoder network receives a constant as input. The latent style vector w generated by the encoder network is fed to the decoder network after each

convolutional layer to control AdaIN operations. Each convolutional block of the decoder network is made up of an upsampling layer followed by two convolutional layers. The activation function used for every layer is *Leaky-Relu* with $\alpha = 0.2$ in this network as well. Additionally, similar to the synthesis network of the StyleGAN model, noise inputs are added to each convolutional block of the decoder network in order to generate stochastic variations in face images such as exact placement of hair. Likewise, as displayed in Table 3.3, the discriminator network of our model is composed of 7 convolutional blocks when the input color image is of size 256×256. It has an architecture similar to the discriminator network of PGGAN model [49] with each convolutional block consisting of two convolutional layers followed by a down-sampling layer. The activation function used here is *Leaky-Relu* with $\alpha = 0.2$ as well. Additionally, the value of weight for the adversarial loss resulting in the best quality of generated images is found to be $\gamma = 0.5 \times 10^{-3}$.

3.2.2 Datasets and Training

We train our deep autoencoder model using four benchmark face image datasets; celebrity faces attributes high quality dataset (CelebA-HQ) [49], Flickr faces high quality dataset (FFHQ) [11], large-scale celebrity faces attributes dataset (CelebA) [71], and UTK face dataset (UTKFace) [70]. We use the aligned and cropped version of each of these datasets in which images only contain the face. The model is trained for CelebA-HQ dataset and FFHQ dataset at resolution 256×256 and for large-scale CelebA dataset and UTKFace dataset at resolution 128×128. We train the deep autoencoder model for each of the four mentioned datasets for 10 million images which is equivalent of around 300 epochs for CelebA-HQ dataset, 150 epochs for FFHQ dataset, 50 epochs for CelebA dataset, and 430 epochs for UTKFace dataset.

Lastly, similar to PGGAN model [49], we start the training of our model with lower resolution of input images and smaller network sizes and grow them progressively and

Encoder	**Activation**	**Output Shape**
Input Image	–	$256 \times 256 \times 3$
Conv 3×3	LeakyReLU	$256 \times 256 \times 64$
Conv 3×3	LeakyReLU	$256 \times 256 \times 128$
Downsample	–	$128 \times 128 \times 128$
Conv 3×3	LeakyReLU	$128 \times 128 \times 128$
Conv 3×3	LeakyReLU	$128 \times 128 \times 256$
Downsample	–	$64 \times 64 \times 256$
Conv 3×3	LeakyReLU	$64 \times 64 \times 256$
Conv 3×3	LeakyReLU	$64 \times 64 \times 512$
Downsample	–	$32 \times 32 \times 512$
Conv 3×3	LeakyReLU	$32 \times 32 \times 512$
Conv 3×3	LeakyReLU	$32 \times 32 \times 512$
Downsample	–	$16 \times 16 \times 512$
Conv 3×3	LeakyReLU	$16 \times 16 \times 512$
Conv 3×3	LeakyReLU	$16 \times 16 \times 512$
Downsample	–	$8 \times 8 \times 512$
Conv 3×3	LeakyReLU	$8 \times 8 \times 512$
Conv 3×3	LeakyReLU	$8 \times 8 \times 512$
Downsample	–	$4 \times 4 \times 512$
Conv 3×3	LeakyReLU	$4 \times 4 \times 512$
Fully-Connected	linear	$1 \times 1 \times 512$
Fully-Connected	linear	$1 \times 1 \times 512$

Table 3.1: Details of the architectures used for the encoder network of the proposed deep autoencoder model.

Decoder	**Activation**	**Output Shape**
Constant Vector	–	$1 \times 1 \times 51$
Conv 4×4	LeakyReLU	$4 \times 4 \times 512$
Conv 3×3	LeakyReLU	$4 \times 4 \times 512$
Noise Input and Latent Input through AdaIN		
Upsample	–	$8 \times 8 \times 512$
Conv 3×3	LeakyReLU	$8 \times 8 \times 512$
Conv 3×3	LeakyReLU	$8 \times 8 \times 512$
Noise Input and Latent Input through AdaIN		
Upsample	–	$16 \times 16 \times 512$
Conv 3×3	LeakyReLU	$16 \times 16 \times 512$
Conv 3×3	LeakyReLU	$16 \times 16 \times 512$
Noise Input and Latent Input through AdaIN		
Upsample	–	$32 \times 32 \times 512$
Conv 3×3	LeakyReLU	$32 \times 32 \times 512$
Conv 3×3	LeakyReLU	$32 \times 32 \times 512$
Noise Input and Latent Input through AdaIN		
Upsample	–	$64 \times 64 \times 512$
Conv 3×3	LeakyReLU	$64 \times 64 \times 256$
Conv 3×3	LeakyReLU	$64 \times 64 \times 256$
Noise Input and Latent Input through AdaIN		
Upsample	–	$128 \times 128 \times 256$
Conv 3×3	LeakyReLU	$128 \times 128 \times 128$
Conv 3×3	LeakyReLU	$128 \times 128 \times 128$
Noise Input and Latent Input through AdaIN		
Upsample	–	$256 \times 256 \times 128$
Conv 3×3	LeakyReLU	$256 \times 256 \times 64$
Conv 3×3	LeakyReLU	$256 \times 256 \times 64$
Noise Input and Latent Input through AdaIN		
Conv 1×1	LeakyReLU	$256 \times 256 \times 3$

Table 3.2: Details of the architectures used for the decoder network of the proposed deep autoencoder model.

Discriminator	**Activation**	**Output Shape**
Input Image	–	$256 \times 256 \times 3$
Conv 3×3	LeakyReLU	$256 \times 256 \times 64$
Conv 3×3	LeakyReLU	$256 \times 256 \times 128$
Downsample	–	$128 \times 128 \times 128$
Conv 3×3	LeakyReLU	$128 \times 128 \times 128$
Conv 3×3	LeakyReLU	$128 \times 128 \times 256$
Downsample	–	$64 \times 64 \times 256$
Conv 3×3	LeakyReLU	$64 \times 64 \times 256$
Conv 3×3	LeakyReLU	$64 \times 64 \times 512$
Downsample	–	$32 \times 32 \times 512$
Conv 3×3	LeakyReLU	$32 \times 32 \times 512$
Conv 3×3	LeakyReLU	$32 \times 32 \times 512$
Downsample	–	$16 \times 16 \times 512$
Conv 3×3	LeakyReLU	$16 \times 16 \times 512$
Conv 3×3	LeakyReLU	$16 \times 16 \times 512$
Downsample	–	$8 \times 8 \times 512$
Conv 3×3	LeakyReLU	$8 \times 8 \times 512$
Conv 3×3	LeakyReLU	$8 \times 8 \times 512$
Downsample	–	$4 \times 4 \times 512$
Minibatch stddev	–	$4 \times 4 \times 513$
Conv 3×3	LeakyReLU	$4 \times 4 \times 512$
Conv 4×4	LeakyReLU	$1 \times 1 \times 512$
Fully-Connected	linear	$1 \times 1 \times 1$

Table 3.3: Details of the architectures used for the discriminator network of the proposed deep autoencoder model.

synchronously during the training. This approach of gradual growing of the network sizes and input resolutions results in speeding up the training of the model and making it more stable. The optimizer used for training our deep autoencoder model is an ADAM optimizer with $\beta = 0.5$. The learning rate of the optimizer is initiated at 0.1×10^{-3} and follows a decay rate of 0.9 in every 10^4 training steps.

3.3 Qualitative Results

3.3.1 Reconstructions with Mixing Style Vectors

After training is completed, we are able to use the encoder network of our deep autoencoder to compute the latent style vectors w for different images. We are also able to create different versions of w_{mixed} from different source images and feed them to the decoder network in order to examine the properties of the learned representations. Following the guideline by StyleGAN model, we divide the latent style vectors into three groups of scale; coarse-scale styles, middle-scale styles, and fine-scale styles. Coarse-scale styles correspond to resolutions 4×4 and 8×8, while middle-scale styles correspond to resolutions 16×16 and 32×32, and finally fine-scale styles correspond to resolutions 64×64 and higher of input face images. In this section we examine the learned latent style vectors for each of these groups of scales in order to examine what information/concepts are captured by each of them.

Figure 3.3 displays examples of exchanging coarse-scale styles, middle-scale styles and fine-scale styles between two sets of face images (sources A and B) by our model trained using CelebA-HQ dataset. The faces in Source A and Source B are generated by reconstructing real face images (i.e. without mixing w), the rest of the images are generated by mixing a specified subset of styles from source B and taking the rest from source A. Figures 3.4, 3.5, and 3.6 display examples of the images generated

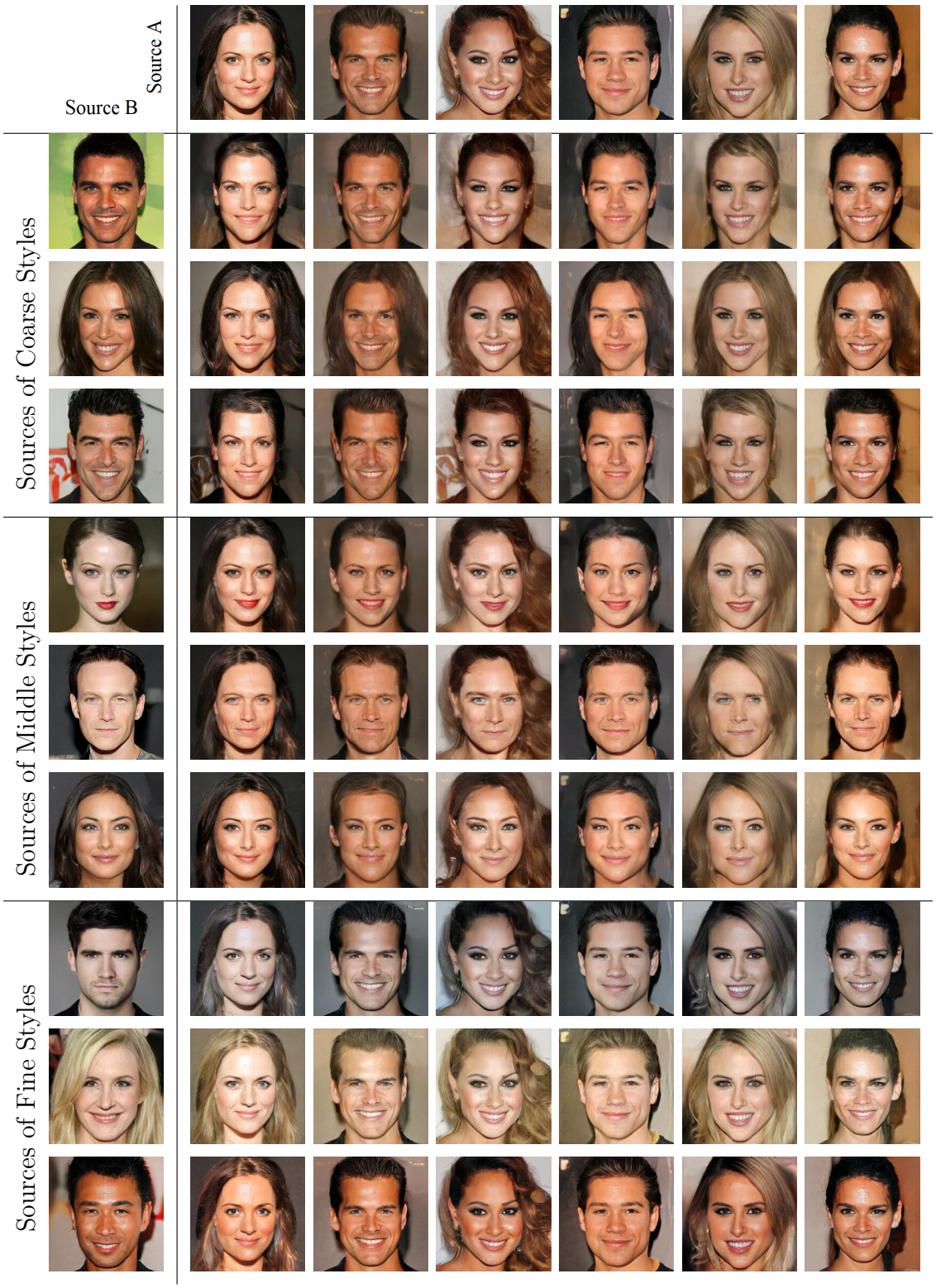


Figure 3.3: Examples of exchanging coarse-scale styles, middle-scale styles and fine-scale styles between two sets of face images (sources A and B) by the proposed deep autoencoder trained using CelebA-HQ 256×256 dataset.

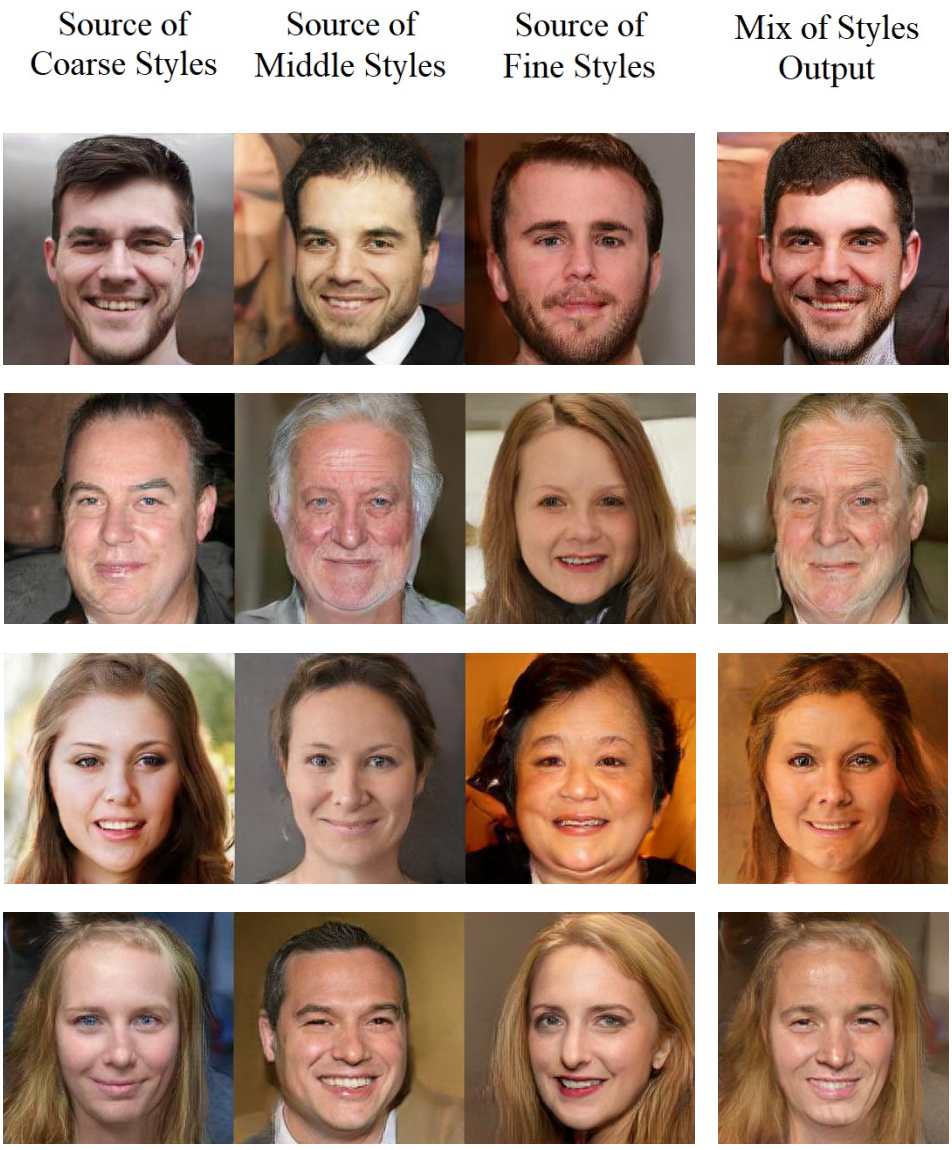


Figure 3.4: Examples of faces generated by combining coarse-scale styles, middle-scale styles and fine-scale styles from three different source images. The model is trained using FFHQ 256×256 dataset.

Figure 3.5: Examples of faces generated by combining coarse-scale styles, middle-scale styles and fine-scale styles from three different source images. The model is trained using CelebA 128×128 dataset.

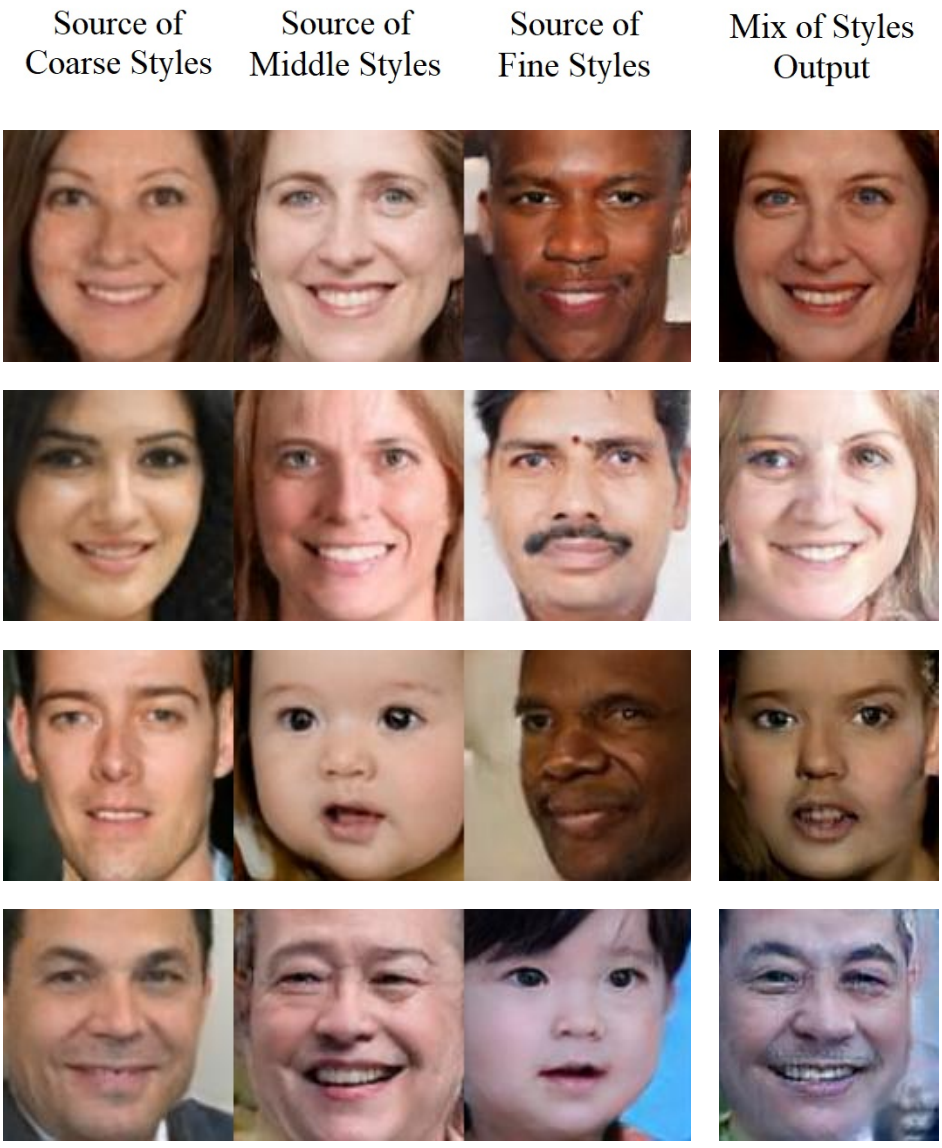
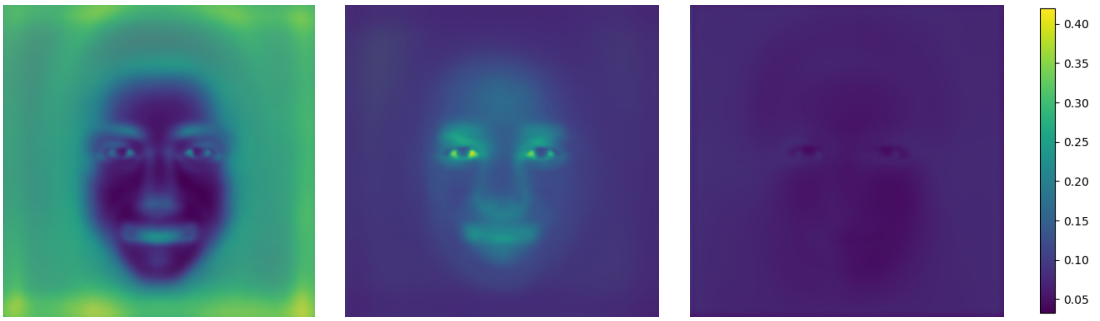


Figure 3.6: Examples of faces generated by combining coarse-scale styles, middle-scale styles and fine-scale styles from three different source images. The model is trained using UTKFace 128×128 dataset.

by copying coarse-scale styles, middle-scale styles, and fine-scale styles from three different source images by our model trained using FFHQ dataset, CelebA dataset, and UTKFace dataset. In these figures, the faces in the source columns are generated by reconstructing real face images without mixing styles. The faces in the output column are generated by mixing specified subsets of styles from the source images.

It can be observed from these reconstructions that coarse styles capture coarse-scale facial concepts such as background, head pose, hairstyle, and face shape. Meanwhile, middle styles extract middle-scale facial features such as information related to identity and facial expressions. Finally, fine styles capture fine-scale features such as colour themes including the background colours, hair colour, skin colour, and the general lighting of the image. Additionally, it can be noted that faces created by combining styles from different source images are slightly blurry in comparison to images generated without style mixing. This is because the outputs of mixing styles are achieved by a RA reconstruction loss enforced on lower resolutions of the input images (i.e. the blurry versions). For example, the model seems to have an issue with removing/adding long hair. More precisely, when removing long hair from a face image traces of long hair remain in the background or the ears look blurry. Similarly, when adding long hair the general hairstyle is transferred but the long hair appears less sharp than the source image. As stated previously, hairstyles are encoded in coarse styles, and coarse styles are enforced by RA reconstruction loss on the lowest resolutions of the input images. This issue seems to resolve as we move toward middle and fine scales of concepts which are enforced by RA reconstruction loss on higher resolutions of input images. The model seems to perform adequately in transferring identity related concepts from one image to another even in cases where the two source images have different head poses. It is also worth mentioning that exchanging the middle-scale latent styles between two source face images is in fact similar to performing face-swap between them as it transfers the identity and facial



(a) Coarse (b) Middle (c) Fine

Figure 3.7: Mean squared-error (MSE) heatmaps computed between 50,000 randomly selected pairs of reconstructed images and their edited counterparts created by replacing a single scale of styles only. The model is trained using CelebA-HQ 256×256 dataset. The brighter parts of the heatmaps display the parts that are most affected when modifying (a) the coarse styles, (b) the middle styles , and (c) the fine styles.

expression from one input image to another. This is particularly interesting because usually image and video face-swap models require training for each subject or pair of subjects and therefore require expensive subject-specific data [72] [73] [41] [74].

To further illustrate which facial concepts are captured by each of the three style groups, Figure 3.7 shows three heatmaps created by computing the Mean-squared error (MSE) between 50,000 randomly selected generated images and their edited counterparts. The edited counterpart for each image is generated by copying a specific scale of styles from another image. The heatmap in Figure 3.7(a) shows that non-facial parts such as background and hairstyle are affected most when changing the coarse styles. Meanwhile, Figure 3.7(b) shows that the facial parts related to identity such as eyes, nose and mouth are impacted most when middle styles are changed. Lastly, the heatmap resulted from changing fine styles does not display an emphasis on any particular parts of face images. That is because the fine styles capture the color themes throughout the entire image. In other words, the heatmap in Figure 3.7(c) shows that modifying fine styles results in almost equal changes in all parts of the image.

3.3.2 Reconstructions without Mixing Style Vectors

Figure 3.8 presents a qualitative comparison between reconstructions of CelebA-HQ samples by our deep autoencoder model and three other benchmark models; StyleALAE [66], Balanced PIONEER [65], and PIONEER [64] which were briefly explained in Chapter 2. These models attempt to achieve a faithful and high quality reconstruction of input images through a balanced combination of reconstruction loss optimization and GAN adversarial loss optimization. As it is displayed in the figure, PIONEER which uses an architecture inspired by PGGAN model [49] is able to generate reconstructions that display a descent level of both visual quality and resemblance to the original real images. The model is later improved by adding new normalization schemes leading to the introduction of Balanced PIONEER. Balanced PIONEER reconstructions display a significant improvement in terms of reconstructions looking close to the real input images. However, they look slightly blurry in comparison to PIONEER reconstructions. StyleALAE is the model that achieves both faithful reconstruction and high visual quality through alternating between reconstruction loss and adversarial loss and also by taking advantage of an architecure inspired by StyleGAN architecture. Similar to StyleALAE, our model utilises an architecture inspired by StyleGAN. Furthermore, our autoencoder model places its main focus on faithful reconstruction of real face images (using a reconstruction loss) and disentanglement of concepts (using the RA reconstruction loss). As a result, our autoencoder reconstructs the real input images more faithfully than the other three models (i.e. our reconstructions display the most resemblance to the real images). Additionally, our model is able to capture not only the facial information but also the information related to the background. These background details appear blurry in our reconstructions. However, they show clear resemblance to the backgrounds of the real images. This reconstruction of background-related information is completely

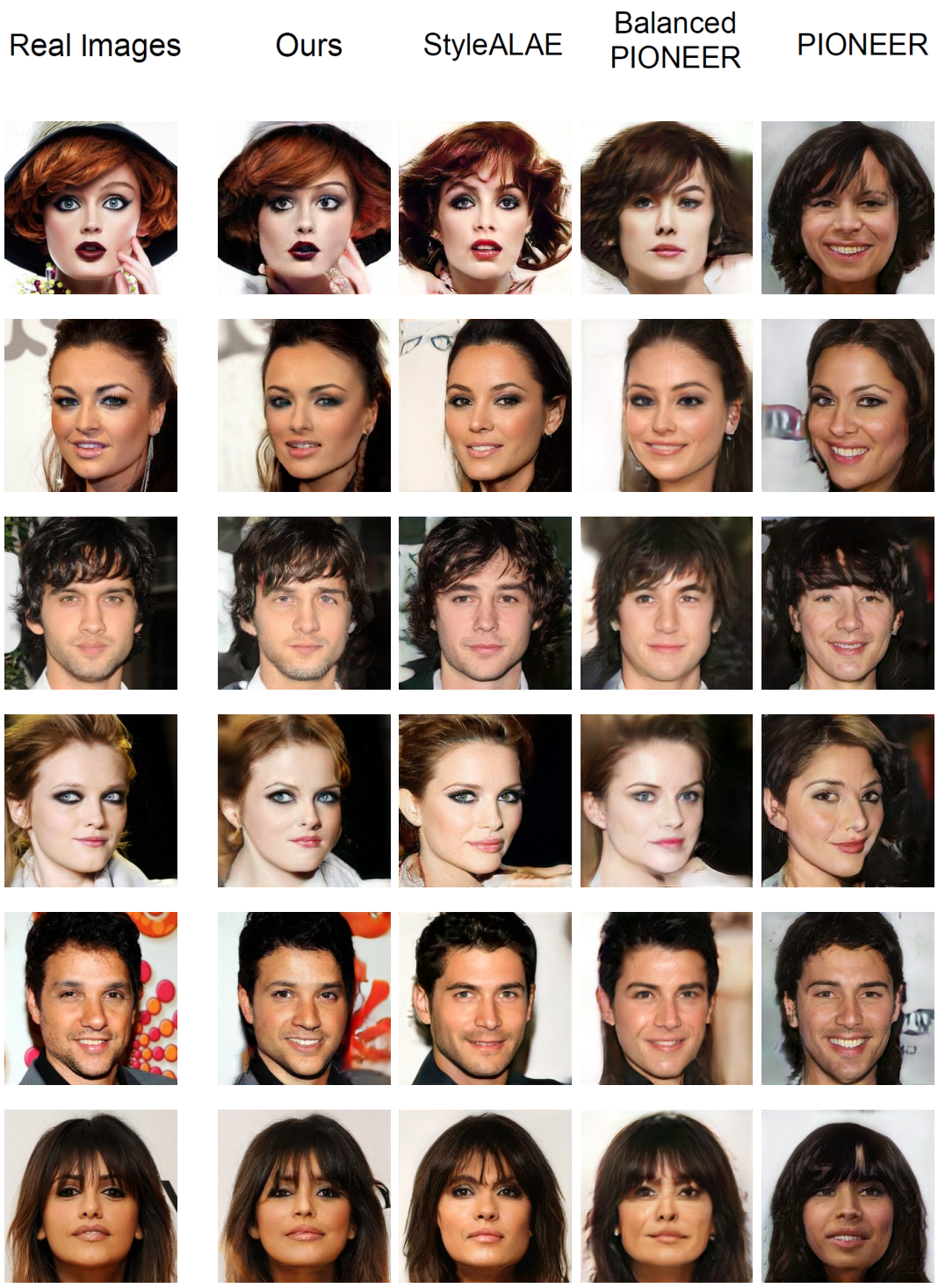


Figure 3.8: Reconstructions of CelebA-HQ [49] samples at resolution 256×256 done by different methods.

absent in the reconstructions performed by the other three models. To summarize, the faithful and high quality reconstructions achieved by our model are a result of StyleGAN inspired architecture and also allowing for the reconstruction loss to dominate the image generation process. This is in contrast to the other models which are trained using a balanced combination of reconstruction loss and adversarial loss.

In order to further display the quality of reconstructions achieved by our deep autoencoder model, Figures 3.9 and 3.10 displays several more examples of pairs of real images in 256×256 resolution and their reconstructions from CelebA-HQ and FFHQ datasets performed by our autoencoder model. Similarly, Figures 3.11 and 3.12 displays examples of pairs of real CelebA and UTKFace images in 128×128 resolution and their reconstructions performed by our autoencoder model. It can be seen from these images that the reconstructions of our autoencoder show a high resemblance to the original image while displaying high visual quality. Please note that usually the small details such as eyeglasses, earrings, or braiding of hair are not present in the reconstructed images. This is again due to our model primarily using reconstruction losses for generating these reconstructed images. Generally speaking, autoencoders are very good in capturing high level information from images. However, they are not able to capture and reconstruct small details.

3.4 Quantitative Analysis

3.4.1 Photorealism versus Disentanglement of Facial Concepts

Table 3.4 reports the Frechet inception distances (FID) scores and the perceptual path length (PPL) scores of our model trained using four different datasets and compares them with the available scores for six other benchmark methods; PGGAN [49],

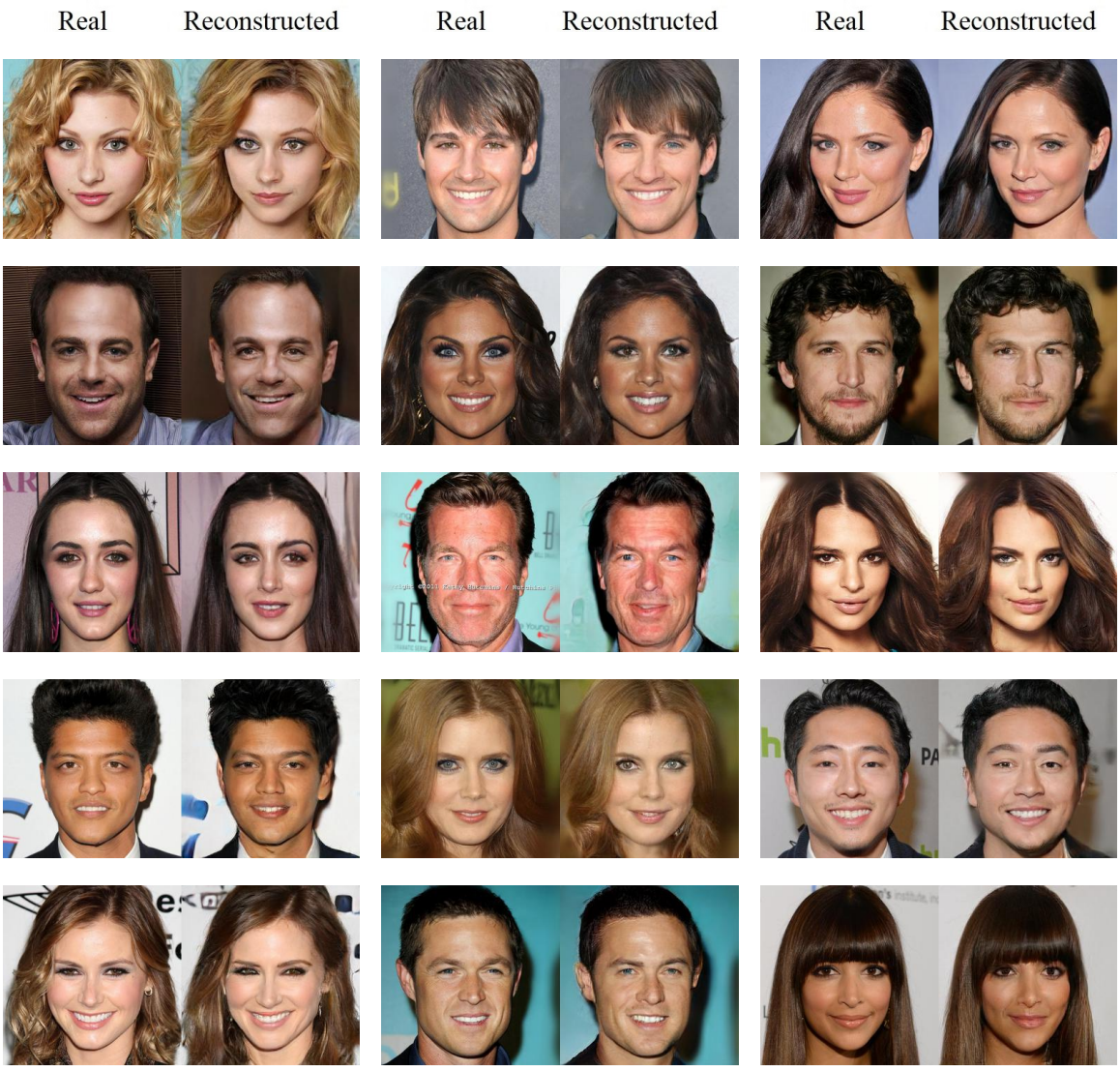


Figure 3.9: Examples of pairs of real CelebA-HQ [49] images at 256×256 resolution and reconstructions done by our deep autoencoder model.

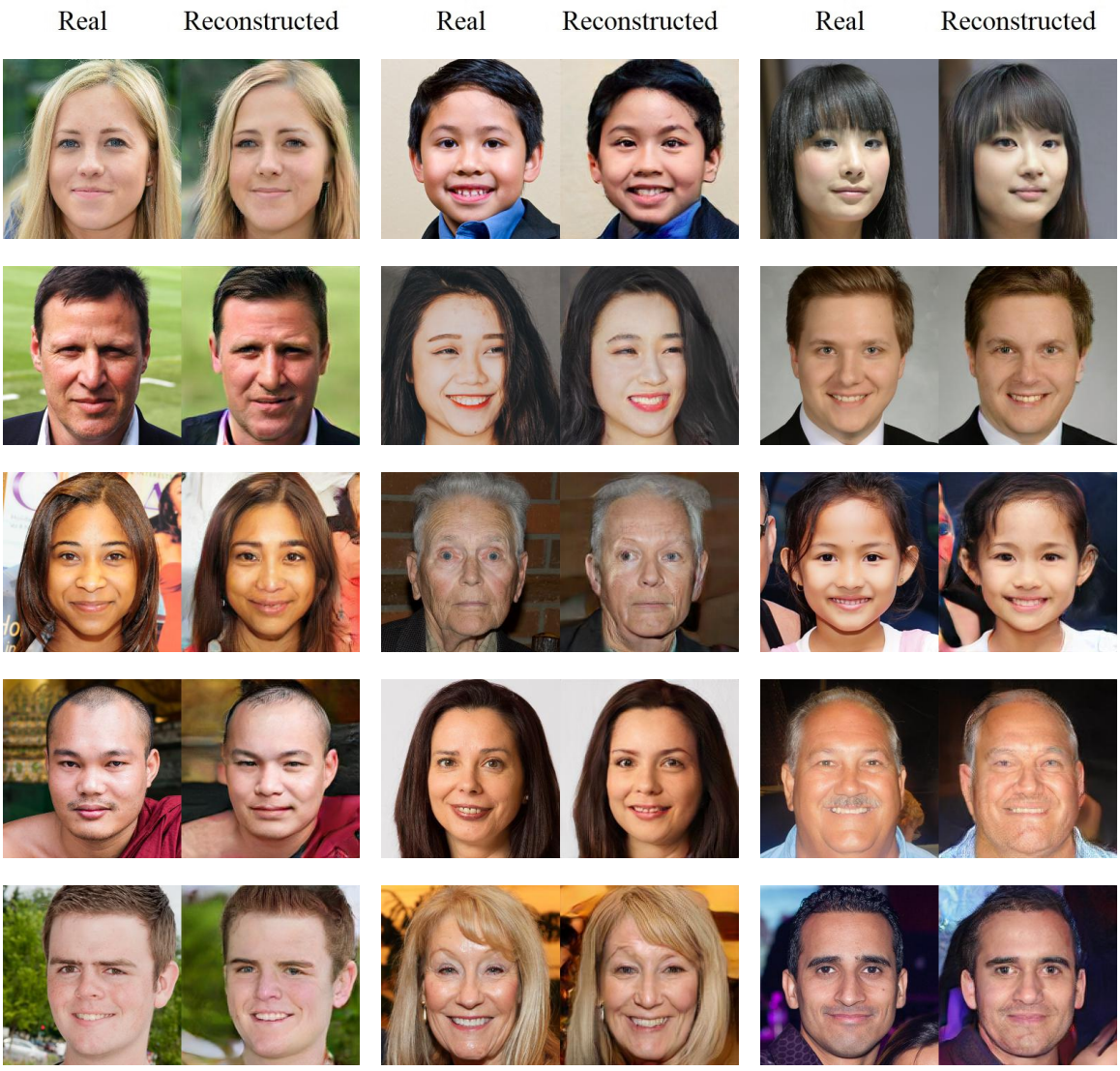


Figure 3.10: Examples of pairs of real FFHQ [11] images at 256 × 256 resolution and reconstructions done by our deep autoencoder model.

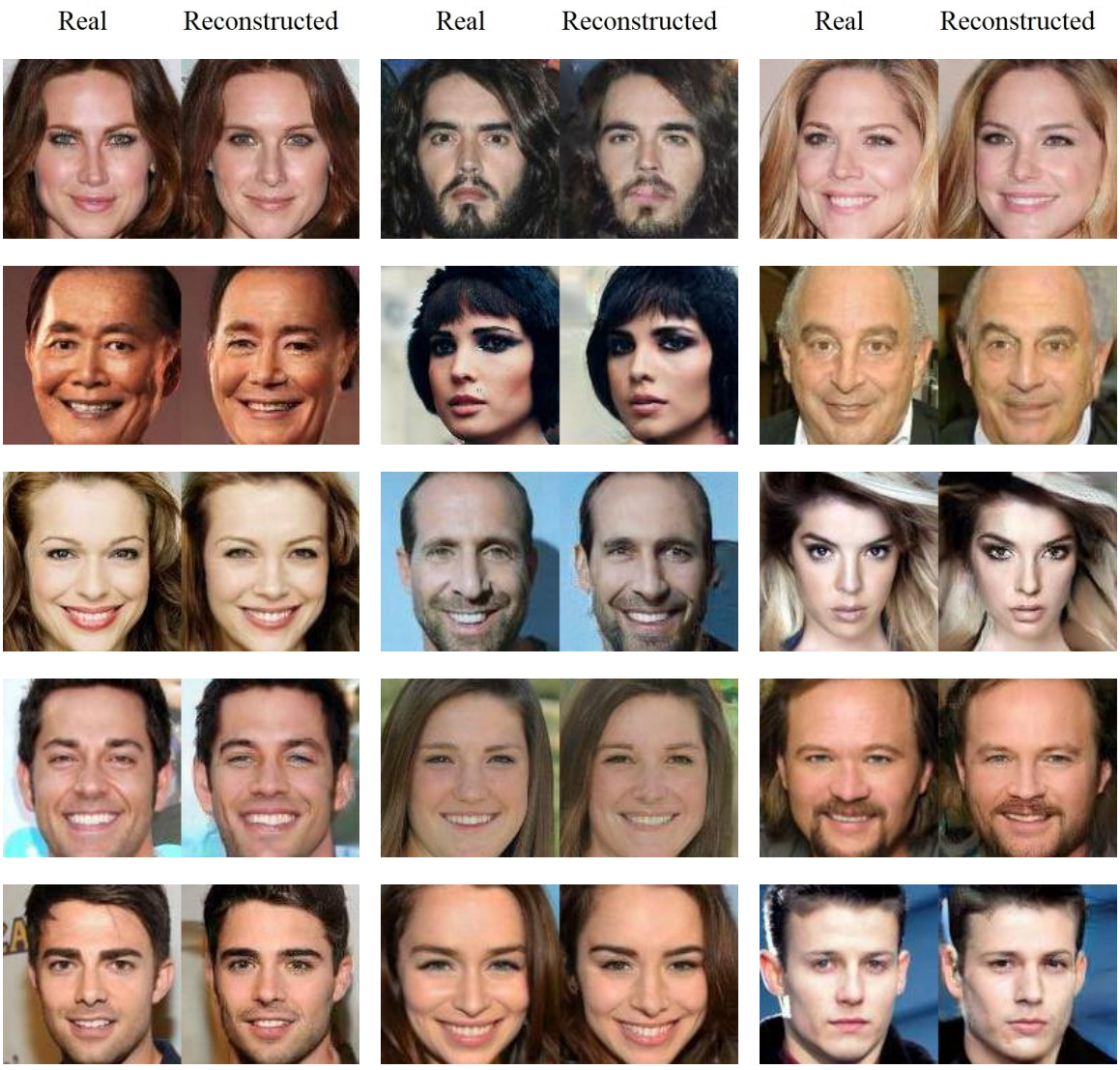


Figure 3.11: Examples of pairs of real CelebA [71] images at 128 × 128 resolution and reconstructionsdone by our deep autoencoder model.

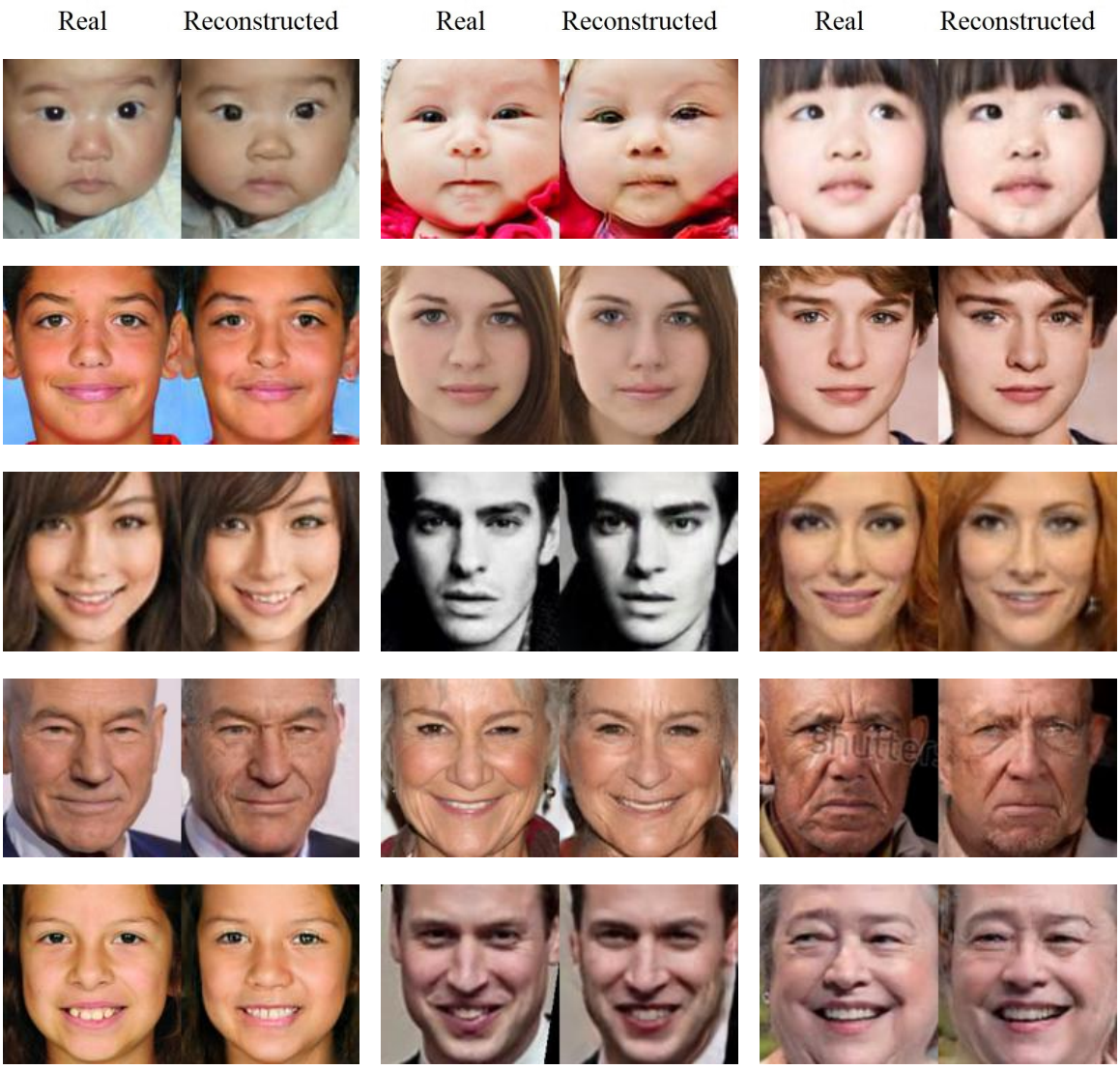


Figure 3.12: Examples of pairs of real UTKFace [70] images at 128×128 resolution and reconstructions done by our deep autoencoder model.

PIONEER [64], Balanced PIONEER [65], StyleALAE [66], AVAE [26], and SoftIntroVAE [68]. A brief review of these benchmark methods was provided in Chapter 2.

FID is a measure of the quality of generated images and is computed by feeding a set of real images and a set of images synthesised by the model to the Inception network and computing the difference between the inception representations of the two groups. PPL is a metric to measure whether the interpolation of latent-space vectors yields non-linear changes in the output image. In other words, if a facial feature is absent in two points in the representation space and it appears in the linear interpolation path between the two points, it is an indication that the facial causal factors are still entangled in the representation space. PPL measures how curved or flat the latent space is by measuring how drastic or smooth are the changes in generated images as we move along a linear interpolation path between two points. In short, a lower FID score indicates more realistic generated images, while a lower PPL score indicates better disentanglement of concepts in the latent representation space.

Generally, we expect that an autoencoder model will not perform as well as GAN-based models in terms of FID score since the data distribution estimation process by GANs is exact and autoencoders only estimate a lower band on the data distribution. However, by including an adversarial loss with a small weight ($\gamma = 0.5 \times 10^{-3}$) in our autoencoder model, it is able to achieve a FID score better than PIONEER, BalancedPIONEER, StyleALAE, and SoftIntroVAE models when trained using CelebA-HQ dataset at 256×256 resolution. Additionally, our model achieves a FID score close to AVAE method when trained using large-scale CelebA dataset at 128×128 resolution. In terms of PPL score, our model achieves a significant improvement in comparison to PGGAN, PIONEER, and BalancedPIONEER models when trained using CelebA-HQ 256×256 dataset. However, it does not achieve as good as of a

Method	CelebA-HQ		FFHQ		CelebA		UTKFace	
	FID	**PPL**	**FID**	**PPL**	**FID**	**PPL**	**FID**	**PPL**
PGGAN [49]	8.03	229.2	–	–	–	–	–	–
PIONEER [64]	39.17	155.2	–	–	23.15	–	–	–
Balanced PIONEER [65]	25.25	146.2	–	–	–	–	–	–
StyleALAE [66]	19.21	**33.29**	–	–	–	–	–	–
AVAE [26]	–	–	–	–	**15.46**	–	–	–
SoftIntroVAE [68]	18.63	–	17.55	–	–	–	–	–
Autoencoder with AR Loss (ours)	**16.29**	100.85	**15.97**	102.26	18.28	64.34	18.09	63.39

Table 3.4: FID scores (lower is better) and PPL scores (lower is better) for the proposed deep autoencoder trained on four benchmark datasets (CelebA-HQ 256×256, FFHQ 256×256, CelebA 128×128, and UTKFace 128×128) along with the available scores for six other benchmark models.

PPL score as StyleALAE model. This indicates that the latent space representation of our model is perceptually more linear and therefore less entangled than PGGAN, PIONEER, and BalancedPIONEER models but not as disentangled as StyleALAE model. This is in agreement with the qualitative results of Figures 3.3, 3.4, 3.5, and 3.6 for which the issue with transferring coarse styles by our deep autoencoder was discussed.

3.4.2 Ablation Studies

We continue using the FID score as a measure of the quality and realness of reconstructed images and and PPL score as a measure of disentanglement in the representation space for our ablation studies.

AR Reconstruction Loss

In order to understand the role that the AR reconstruction loss plays in the scale-based disentanglement of concepts by our autoencoder model, we train our autoencoder model without the AR reconstruction loss (i.e. using only standard reconstruction loss and weighted adversarial loss). The model trained without the AR reconstruction loss does not perform any style mixing and subsequently does not enforce any constraint on images generated by style mixing. This is in contrast to the model trained with AR reconstruction loss which performs mixing in styles in every step of training and enforces a constraint on images generated by mixing styles. We compute the FID score and the PPL score for this model and compare them with our autoencoder model with AR reconstruction loss in Table 3.5.

The significant increase in PPL score of the model resulted from removing the AR reconstruction loss provides evidence for the significant role that novel AR reconstruction loss plays in achieving the disentanglement of facial concepts in the latent representation space by our autoencoder model. However, the FID score does not change as drastically by removing the AR reconstruction loss indicating that the quality of reconstructed images does not highly depend on the inclusion of the AR reconstruction loss.

Method	FID	PPL full
Autoencoder Model		
with AR Loss (Style Mixing)		
GAN Loss $\gamma = 0$	105.28	**97.53**
GAN Loss $\gamma = 0.0001$	34.20	98.19
GAN Loss $\gamma = 0.0005$	**16.29**	100.85
GAN Loss $\gamma = 0.001$	16.61	100.15
Autoencoder Model		
with GAN Loss $\gamma = 0.0005$		
Without AR Loss (No Style Mixing)	20.69	174.45

Table 3.5: Comparison of FID scores (lower is better) and PPL scores (lower is better) for the proposed deep autoencoder given different values of weights for the advesarial loss and also after removing the AR reconstruction loss. The models are all trained using CelebA-HQ dataset at 256×256 resolution.

GAN Adversarial Loss

In order to understand the role of adversarial GAN loss in our results, Table 3.5 also compares the FID scores and PPL scores for images generated by our proposed deep autoencoder model using different values of weight γ for the adversarial loss. It is evident from the table, that the inclusion of the adversarial loss plays a significant role in improving the quality of reconstructed images by our model. This increase in image quality is shown through decreases in FID score. However, the PPL score does not change drastically for different weights of the adversarial loss indicating that the inclusion of the adversarial loss does not influence the disentanglement of concepts in the latent representation space. In other words, our autoencoder model achieves the disentanglement of concepts in the latent space by its architecture and the AR reconstruction loss only. Additionally, through experimentations with different values of weight for adversarial GAN loss the value $\gamma = 0.5 \times 10^{-3}$ is observed to result in

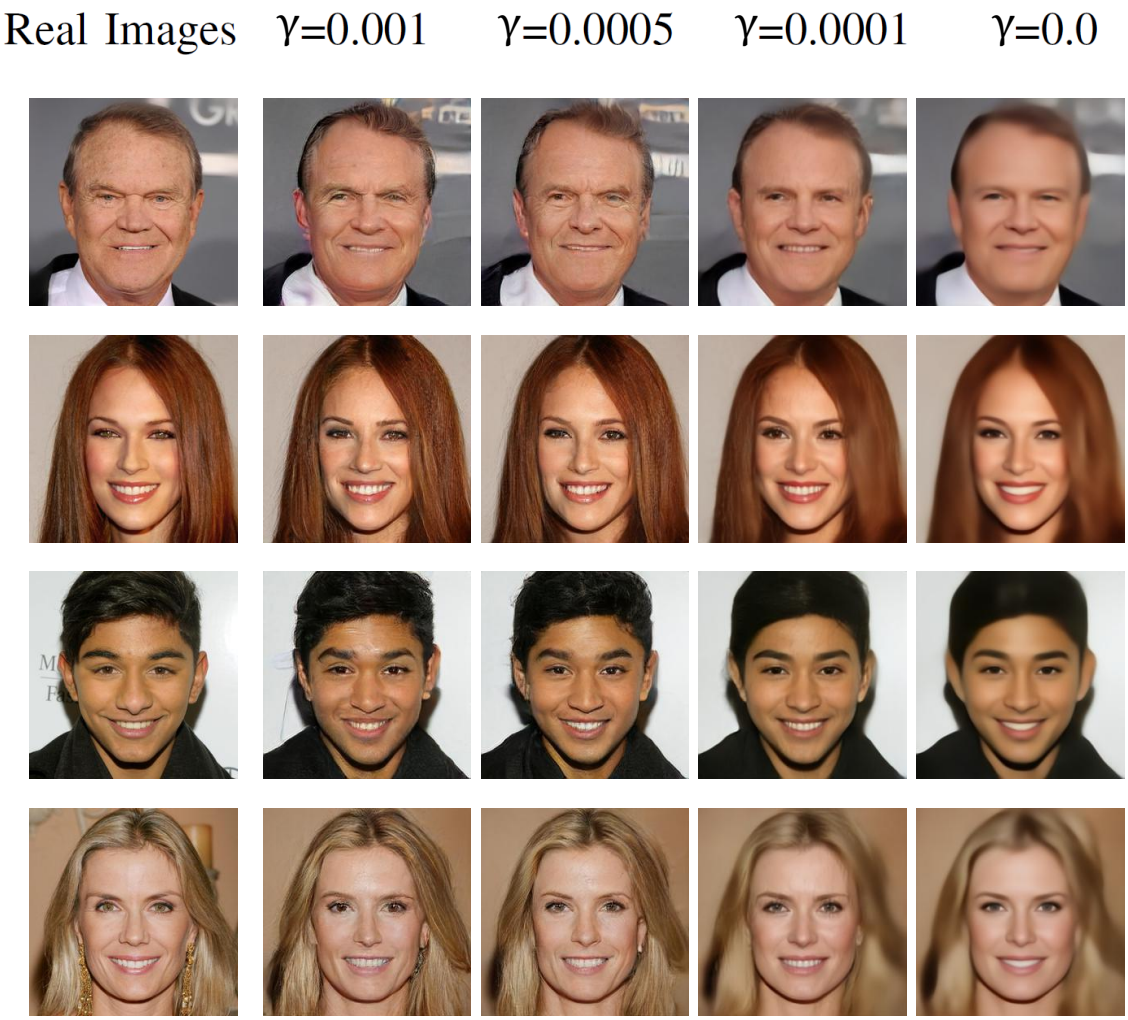


Figure 3.13: Example of reconstructions of CelebA-HQ [49] faces by the proposed deep autoencoder trained using different weights for the adversarial loss. The models are all trained using CelebA-HQ dataset at 256×256 resolution.

the best FID score and therefore is used for our final model. A visualization of the influence of the adversarial loss in improving the quality of reconstructed images is shown in Figure 3.13. It displays examples of reconstructions by our deep autoencoder model trained using different weights of the adversarial loss. It can be seen from this figure that the adversarial loss is responsible for reducing the blurriness of reconstructed face images and making them look more realistic.

3.4.3 Impact of Style Mixing on The Quality of Generated Images

In Table 3.6 we compare the FID scores of images generated without style mixing (i.e. the faithful reconstructions of real images) with FID scores of images generated with style mixing (i.e. modifying real images by transferring concepts from another image). It is evident from the table that FID scores for reconstructions without style mixing are always lower than the reconstructions with style mixing. This is in agreement with the qualitative results displayed in Figures 3.3, 3.4, 3.5, and 3.6 in which faces generated by combining styles from different source images are slightly blurry in comparison to images generated without style mixing. The difference between the quality of these two groups of images originates from the fact that the generation of images without style mixing is enforced by reconstruction loss on highest resolution of input images, while the generation of images with style mixing is enforced by reconstruction loss on lower resolutions of the input images (i.e. the blurry versions).

3.5 Summary

In this chapter we proposed a deep autoencoder model that takes advantage of an architecture inspired by StyleGAN model and a novel adaptive resolution reconstruction loss. The adaptive resolution reconstruction loss is introduced inspired by the

Dataset	FID without Mixing	FID with Mixing
CelebA-HQ 256×256	10.27	16.29
FFHQ 256×256	10.08	15.97
CelebA 128×128	13.72	18.28
UTKFace 128×128	12.44	18.09

Table 3.6: A comparison between FID scores of images generated without mixing styles and the images generated with mixing styles by the proposed deep autoencoder trained on four benchmark datasets; CelebA-HQ 256×256, FFHQ 256×256, CelebA 128×128, and UTKFace 128×128

fact that different categories of concepts are encoded in (and can be captured from) different resolutions of an image. We proposed that it is possible to control the coarser concepts in a generated face image by enforcing a reconstruction loss on only the lower-resolution versions of that image. This new type of reconstruction loss facilitates learning a latent representation for real face images in which facial concepts are disentangled based on scale. We demonstrated that the autoencoder trained using the adaptive resolution reconstruction loss achieves promising results in disentangling the facial concepts associated with specific scales and therefore transferring these scale-based concepts from one real face image to another. This is achieved without the help of labels or performing matching between the input images. Furthermore, we showed that by including a discriminator network along with an adversarial loss with a small weight we can reduce the blurriness associated with autoencoder generated images. As a result, the proposed autoencoder is able to outperform benchmark models in generating faithful and high quality reconstructions of real face images.

Chapter 4

A Compositional Generative Adversarial Networks

In this chapter we implement and integrate a notion of compositionality into the GAN framework for the purpose of learning a more flexible and better disentangled distribution of face images. We propose that since all face images display the same underlying structure, a sense of position for meaningful facial components within a face image can be achieved by dividing the images into parts with fixed positions and sizes each containing specific components only. We introduce a methodology for building and training GANs to learn the distribution of face images as compositions of distributions of such parts. As a result, the model is able to produce realistic high-quality face images by sampling from the learned distributions for parts and then generating the parts and piecing them together. Given that each part is defined to only contain specific components of a face, learning a separate distribution for each part is equivalent to disentangling these components in the representation space. Such implementation of compositionality makes it possible to construct a large number of whole representations from a finite set of parts representations.

4.1 Methodology

4.1.1 Background: Standard GAN

As it was explained in Chapter 2, GAN [13] is a deep generative model that focuses on generating samples from a dataset distribution, instead of explicitly modeling and solving this distribution. The main idea of GAN is to first sample from a known latent distribution $z \sim p(z)$ and then transform this sample into a sample of training distribution. Two networks are involved in a GANs model; discriminator network and generator network. Given a data distribution $x \sim p_{data}(x)$, $x \in \mathcal{X}$, the generator network $G(z; \theta_g)$ maps samples of a latent prior to samples of the training distribution $G : \mathcal{Z} \to \mathcal{X}$. Meanwhile, the discriminator network $D(x; \theta_d)$ receives the real data samples and also the samples synthesised by generator network and estimates the probability of whether they are real or fake via a score $D : \mathcal{X} \to \mathbb{R}$. The training process of GANs constantly alternates between training of the discriminator network and training of the generator network. This can be interpreted as the two networks playing a minimax game with the value function

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{data}(x)}[\log D(x; \theta_d)] + \mathbb{E}_{z \sim p(z)}[\log(1 - D(G(z; \theta_g); \theta_d))]. \quad (4.1)$$

This minimax game will eventually converge to an equilibrium state in which the samples generated by generator network are identical to real training data and discriminator network assigns probability of 0.5 to every input regardless of whether it is real or fake.

4.1.2 The Compositional GANs

To achieve a compositional GANs, we add an assumption to the standard GANs model that the latent prior distribution $p(z)$ is composed of K distinct distributions $\{p(z_i)\}_{i=1}^{K}$ and a syntactic method R_z. Assuming that there is a function $h()$ that pairs up each z_i with an image part x_i and also R_z with the way x_is are combined R_x:

$$x_1 = h(z_1), ..., x_k = h(z_k), R_x = h(R_z), \tag{4.2}$$

then the system is compositional if and only if every (syntactically) complex item x in the syntactic system is a function of syntactic parts $x_i = h(z_i)$ and the way they are combined $R_x = h(R_z)$ [75]

$$x = f(x_i, R_x) = f(h(z_i), h(R_z)). \tag{4.3}$$

We define $\{p(z_i)\}_{i=1}^{K}$ and their syntactic method R_z according to the structure present in face images. This process is explained in detail in the next section. The generator network of the compositional GAN learns the functions $h()$ and $f()$. More precisely, the generator network of our model performs the following mappings

$$G : \{\mathcal{Z}_i\}_{i=1}^{K}, R_z \xrightarrow{h} \{\mathcal{X}_i\}_{i=1}^{K}, R_x \xrightarrow{f} \mathcal{X}. \tag{4.4}$$

It means that the generator network learns the mapping $h()$ from the samples of parts priors z_i and the relationship among them R_z to the image parts x_i and the relationship among them R_x. It also learns the mapping $f()$ from image parts x_i and the relationship among them R_x to a whole image x. This is how the generator

implicitly defines a distribution over data:

$$G(z_i, \theta_h, \theta_f) = f(h(z_1, ..., z_K, R_z; \theta_h); \theta_f) = p_g(x). \tag{4.5}$$

This can be interpreted as a group of generators working in parallel to generate the image parts. However, the objective function used for training the compositional GAN model is similar to the one presented in Equation 4.1. More precisely, the adversarial loss is defined for the realistic whole images and not the individual image parts. This results in the compositional GAN model learning the relations among the priors/parts in order to make the whole pieced-together image comparable with real samples from the dataset.

4.1.3 Latent Priors and Their Syntactic Method

Typically in GANs architecture the first layer of the generator network is for generating a volume $w \times h \times c$ by passing the input latent vector z through a fully-connected (FC) layer followed by a reshaping operation as it is shown in Figure 4.1 (a). The purpose of this transition from a 1-D vector to a 3-D volume is to configure the layers input/output shapes compatible with the subsequent convolutional layers. Our modifications to the standard GANs model focuses on this component of the generator network. We replace the single input vector z with multiple input vectors $\{z_i\}_{i=1}^{K}$ with each passing through a distinct FC layer followed by a distinct reshaping operation. The results are multiple volumes $\{v_i\}_{i=1}^{K}$ that can be concatenated together in order to create the volume $v = w \times h \times c$ as input to the first convolutional layer. Figures 4.1 (b) and (c) respectively illustrate the way these architecture modifications are implemented for the two main experiments of this chapter; 1) Two-part compositional GAN in which face images are composed of two parts, one representing the face and the other representing hair and background. This is similar to the separation that was

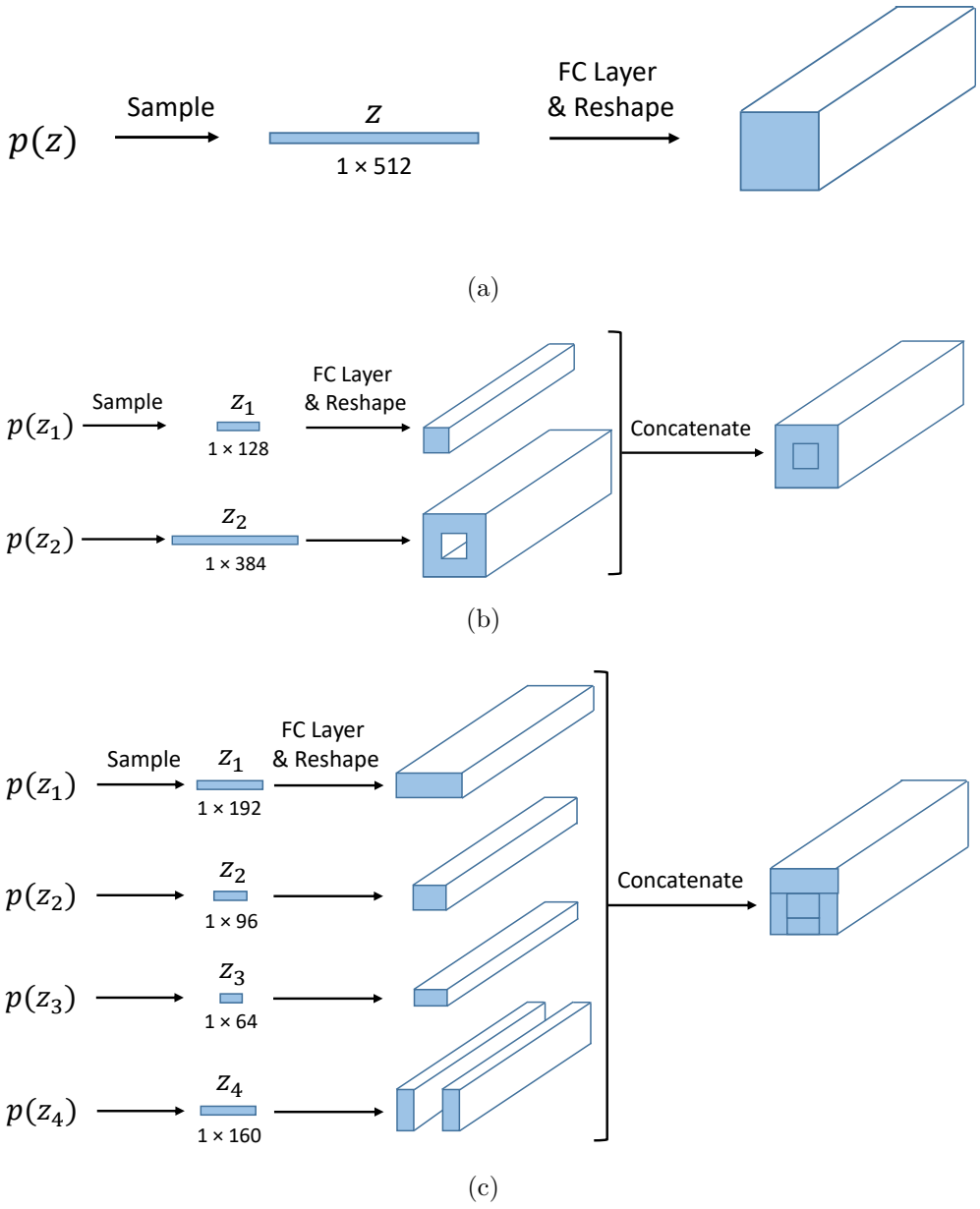


Figure 4.1: (a) The first layer of generator network in a standard GANs architecture. (b) The first layer of generator network in the two-part compositional GAN model which generates face images as a composition of two distinct parts. (c) The first layer of generator network in the four-part compositional GAN model which generates face images as a composition of four distinct facial components.

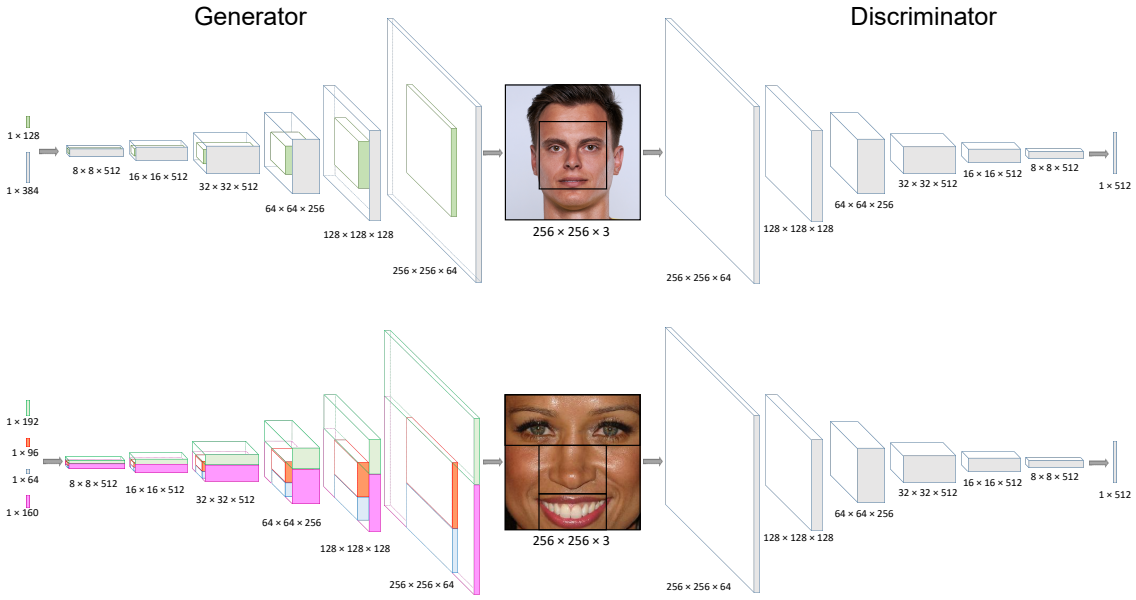


Figure 4.2: The schematic view of (top) the generator network and the discriminator network of the two-part compositional GAN model in which face images are composed of two parts; one for the face and one for hair & background, (bottom) the generator network and the discriminator network of the four-part compositional GAN model in which cropped faces are composed of four distinct facial components; 1) eyes, 2) nose, 3) mouth, and 4) Jaw & cheeks. The displayed faces are samples from FFHQ [11] dataset.

achieved by the deep autoencoder proposed in Chapter 3. 2) Four-part compositional GAN in which faces are composed of four distinct facial components; 1) eyes, 2) nose, 3) mouth, and 4) Jaw & cheeks.

Convolutional layers operate by scanning an input volume row by row starting from the top left corner. Therefore, the arrangement of the volumes $\{v_i\}_{i=1}^{K}$ is carried from the input to the output of a convolutional operation. Similarly, as it is illustrated in Figure 4.2, the arrangement of the volumes $\{v_i\}_{i=1}^{K}$ from the first layer carries throughout the entire network until the output image is generated without requiring any further modifications to the architecture. To summarize, the choice of shapes and the manner of arranging the volumes $\{v_i\}_{i=1}^{K}$ into the larger volume v in the first

layer of the generator network determines which part of the output images each latent variable $\{z_i\}_{i=1}^{K}$ is responsible to represent.

4.2 Experiments and Setup

In this section, we detail the building and training of compositional GANs for the purpose of learning the distribution of face images.

4.2.1 Datasets and Defining the Parts

The following benchmark face datasets are used in our experiments to train the compositional GAN models; FFHQ Dataset [11] and CelebA-HQ [49] dataset. FFHQ dataset consists of 70,000 high-quality face images displaying extensive variations in terms of age, ethnicity and image background. CelebA-HQ dataset is the high-quality version of the large-scale CelebA [71] dataset and consists of 30,000 high-resolution photographs of celebrity faces.

We define the image parts for training of the compositional GANs model inspired by the way humans perceive face images. When a human is asked to describe a face image they have the ability to divide it into parts and then describe each part. We would like our model to have a similar way of dividing face images into smaller components. In other words, we would like to divide training examples into predefined parts each containing specific components only and learn a representation for each part. This way we can assure that the representation learned for a part describes a specific component and the model succeeds in learning it. Clearly, for this to be implemented an alignment among the training samples is required. However, this is not a very strict notion of alignment. More precisely, the model requires the specific concepts to be located in approximately the same position within all training examples. This requirement is met when using the aligned and cropped versions of

FFHQ and CelebA-HQ datasets.

Here we first divide a face image using an 8×8 grid and then define the parts. Figure 4.3 (top) displays samples from FFHQ dataset along with the way parts are defined for the two-part compositional GAN model; one part contains the face and the other contains everything else. Similarly, Figure 4.3 (bottom) displays cropped samples of CelebA-HQ dataset along with the way parts are defined for the four-part compositional GAN model. We crop the CelebA-HQ samples to include the face part only for implementing and training the four-part compositional GAN model. This allows for a more elaborate division of face images into multiple facial component. As it can be seen from this figure, these four distinct parts are defined to contain 1) eyes, 2) nose, 3) mouth, and 4) Jaw & cheeks.

4.2.2 Architecture and Training

We utilize the PGGAN model [49] as the backbone for implementing our compositional GAN model. More precisely, the generator and discriminator of our compositional GAN model have a similar architecture to the generator and discriminator networks of PGGAN model except for two modifications. Firstly, the first convolutional block from the generator network and the last convolutional block from the discriminator network which have a size of $4 \times 4 \times 512$ are removed. Therefore, the initial block of our generator has a size of $8 \times 8 \times 512$. This modification allows for the parts in the first layer of the generator to be defined better and for each latent prior vector to have a more local effect on the output image. More precisely, we require a resolution of at least 8×8 for the input of the first convolutional layer in order to be able to place the parts in relatively exact positions. Secondly, the first layer of the generator network is modified in order to incorporate multiple input latent vectors instead of one latent vector. The sizes of latent vectors $\{z_i\}_{i=1}^{K}$ are selected relative to the area of the output image that they are responsible to represent and conditioned

Figure 4.3: An illustration of the parts defined for training the compositional GAN models. (top) FFHQ [11] samples divided into two parts one containing the face and the other containing hair & background, (bottom) Cropped CelebA-HQ [49] samples divided into four parts each containing a specific facial component; 1) eyes, 2) nose, 3) mouth, and 4) Jaw & cheeks.

to $\sum_{i=1}^{K} len(z_i) = 512$.

The details of the architectures used for the generator network and the discriminator network of our compositional GAN model are shown in Tables 4.1 and 4.2. The compositional GAN models are trained using the 256×256 resolution of CelebA-HQ or FFHQ images. Therefore, the generator network is composed of six convolutional blocks transforming the latent vectors into color images of size 256×256. Each Convolutional block of the generator network consists of an up-sampling layer followed by two convolutional layers. The activation function used for every layer is *Leaky-Relu* with $\alpha = 0.2$. Similarly, the discriminator network architecture is composed of six convolutional blocks. It receives a color image of size 256×256 as input and outputs a score. Each convolutional block of the discriminator network is made up of two convolutional layers followed by a down-sampling layer. The activation function used for every layer is *Leaky-Relu* with $\alpha = 0.2$ in this network as well. Lastly, similar to PGGAN model, we start the training of our model with lower resolution of input images and smaller network sizes and grow them progressively and synchronously during the training. The initial networks work with low-resolution 8×8 face images. As the networks grow larger in size, the resolution of the images increases gradually as well. This approach of gradual growing of the network sizes and input resolutions results in speeding up the training of the model and making it more stable. We use the WGAN-GP loss [76] and an ADAM optimizer with $\alpha = 0.001$, $\beta_1 = 0.0$, $\beta_2 = 0.99$, and $\epsilon = 10^{-8}$ for training our compositional GAN model. We train the models using each dataset for 10 million images which is equivalent of around 300 epochs for CelebAHQ dataset, 150 epochs for FFHQ dataset.

Generator	**Activation**	**Output Shape**
latent Vectors $\{z_i\}_{i=1}^{K}$	–	$\{1 \times \text{size}(z_i)\}_{i=1}^{K}$
FC Layers $\{l_i\}_{i=1}^{K}$	LeakyReLU	$\{1 \times \text{size}(l_i)\}_{i=1}^{K}$
$\text{Reshapes}_{i=1}^{K}$ and Concatenate	–	$8 \times 8 \times 512$
Conv 4×4	LeakyReLU	$8 \times 8 \times 512$
Conv 3×3	LeakyReLU	$8 \times 8 \times 512$
Upsample	–	$16 \times 16 \times 512$
Conv 3×3	LeakyReLU	$16 \times 16 \times 512$
Conv 3×3	LeakyReLU	$16 \times 16 \times 512$
Upsample	–	$32 \times 32 \times 512$
Conv 3×3	LeakyReLU	$32 \times 32 \times 512$
Conv 3×3	LeakyReLU	$32 \times 32 \times 512$
Upsample	–	$64 \times 64 \times 512$
Conv 3×3	LeakyReLU	$64 \times 64 \times 256$
Conv 3×3	LeakyReLU	$64 \times 64 \times 256$
Upsample	–	$128 \times 128 \times 256$
Conv 3×3	LeakyReLU	$128 \times 128 \times 128$
Conv 3×3	LeakyReLU	$128 \times 128 \times 128$
Upsample	–	$256 \times 256 \times 128$
Conv 3×3	LeakyReLU	$256 \times 256 \times 64$
Conv 3×3	LeakyReLU	$256 \times 256 \times 64$
Conv 1×1	LeakyReLU	$256 \times 256 \times 3$

Table 4.1: Details of the architectures used for the generator network of the proposed compositional GAN model.

Discriminator	**Activation**	**Output Shape**
Input Image	–	$256 \times 256 \times 3$
Conv 3×3	LeakyReLU	$256 \times 256 \times 64$
Conv 3×3	LeakyReLU	$256 \times 256 \times 128$
Downsample	–	$128 \times 128 \times 128$
Conv 3×3	LeakyReLU	$128 \times 128 \times 128$
Conv 3×3	LeakyReLU	$128 \times 128 \times 256$
Downsample	–	$64 \times 64 \times 256$
Conv 3×3	LeakyReLU	$64 \times 64 \times 256$
Conv 3×3	LeakyReLU	$64 \times 64 \times 512$
Downsample	–	$32 \times 32 \times 512$
Conv 3×3	LeakyReLU	$32 \times 32 \times 512$
Conv 3×3	LeakyReLU	$32 \times 32 \times 512$
Downsample	–	$16 \times 16 \times 512$
Conv 3×3	LeakyReLU	$16 \times 16 \times 512$
Conv 3×3	LeakyReLU	$16 \times 16 \times 512$
Downsample	–	$8 \times 8 \times 512$
Minibatch stddev	–	$8 \times 8 \times 513$
Conv 3×3	LeakyReLU	$8 \times 8 \times 512$
Conv 4×4	LeakyReLU	1×512
Fully-Connected	linear	1×1

Table 4.2: Details of the architectures used for the discriminator network of the proposed compositional GAN model.

4.3 Qualitative Results

4.3.1 Two-Part Compositional GAN

The two-part compositional GAN model is trained to generate face images as a composition of two parts; one containing the face and the other containing hair and background. As a result, it makes it possible to modify the generated faces in a controlled way. It means that it is possible to modify the face part without changing the hair and background part. Similarly it is possible to modify the hair and background part only. Figure 4.4 displays examples of faces generated by the two-part compositional GAN model. In each row of the figure, the *Source1* face is generated by a random sample of latent priors $\{z_{1,s1}, z_{2,s1}\}$ and the *Source2* face is generated by another random sample from latent priors $\{z_{1,s2}, z_{2,s2}\}$. The *Combination* displays an illustration of how parts representations from the two source images are combined to create the *Output* face in each row. More precisely, the *Output* face is synthesised by passing the latent representations $\{z_{1,s2}, z_{2,s1}\}$ to the generator of the two-part compositional GAN model.

As it can be seen from the figure, the *Output* faces display the identity and facial expressions similar to *Source2* and hair, background, and head pose similar to *Source1*. This indicates that the latent prior z_1 captures the identity and facial expressions while the latent prior z_2 captures information related to hair, background, and head pose. This is in agreement with the way latent priors are defined and arranged in Figure 4.2 (top). The shape of the face falls on the boundary/interlocking region between the two parts. This results in the *Output* face to have a shape in between the two sources' face shapes. Additionally, the color of skin in the *Output* face is more similar to the color of skin in *Source2*, while the colors of hair and background are more similar to the colors of hair and background in *Source1*. Finally, the general lighting of the image seems to be more influenced by *Source1* (i.e. the hair

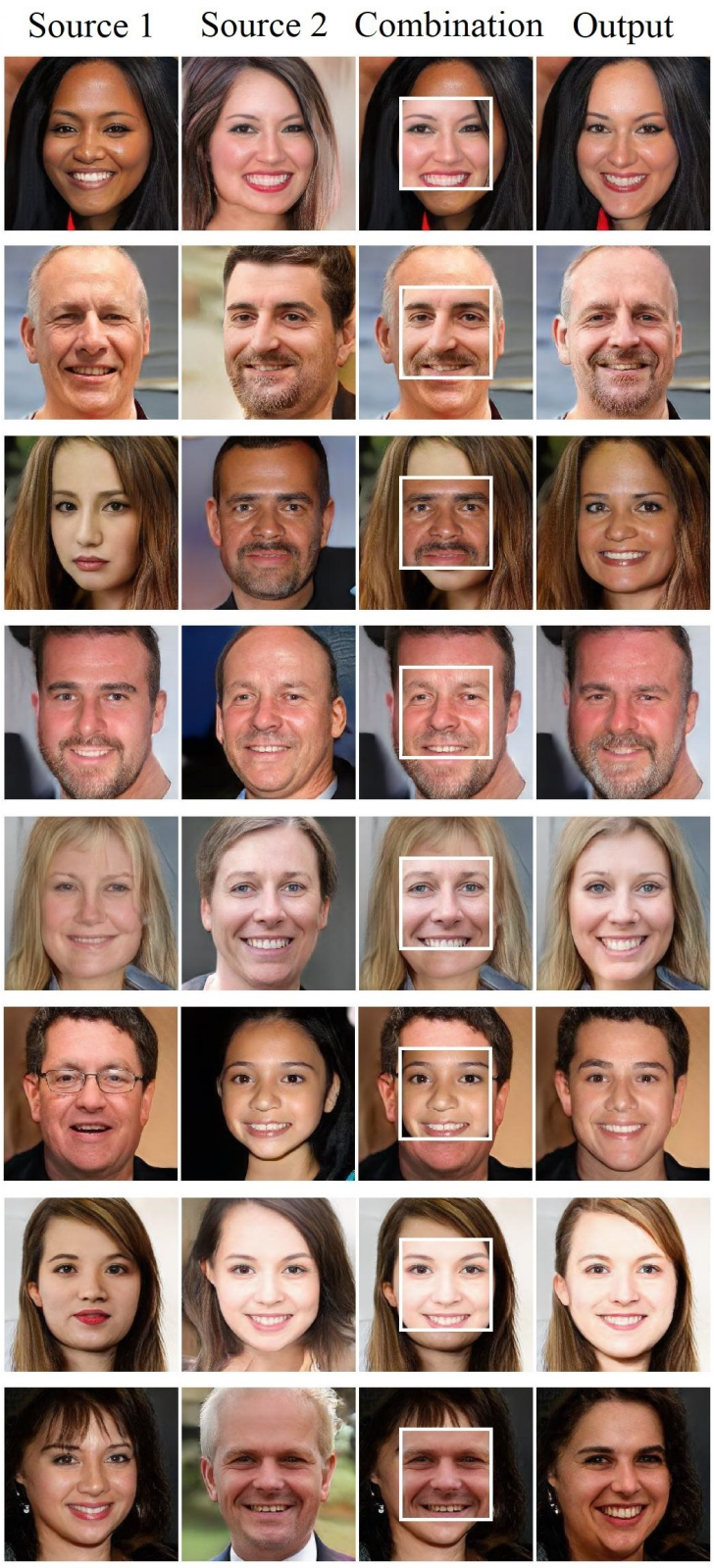


Figure 4.4: Examples of generated faces by the two-part compositional GAN model.

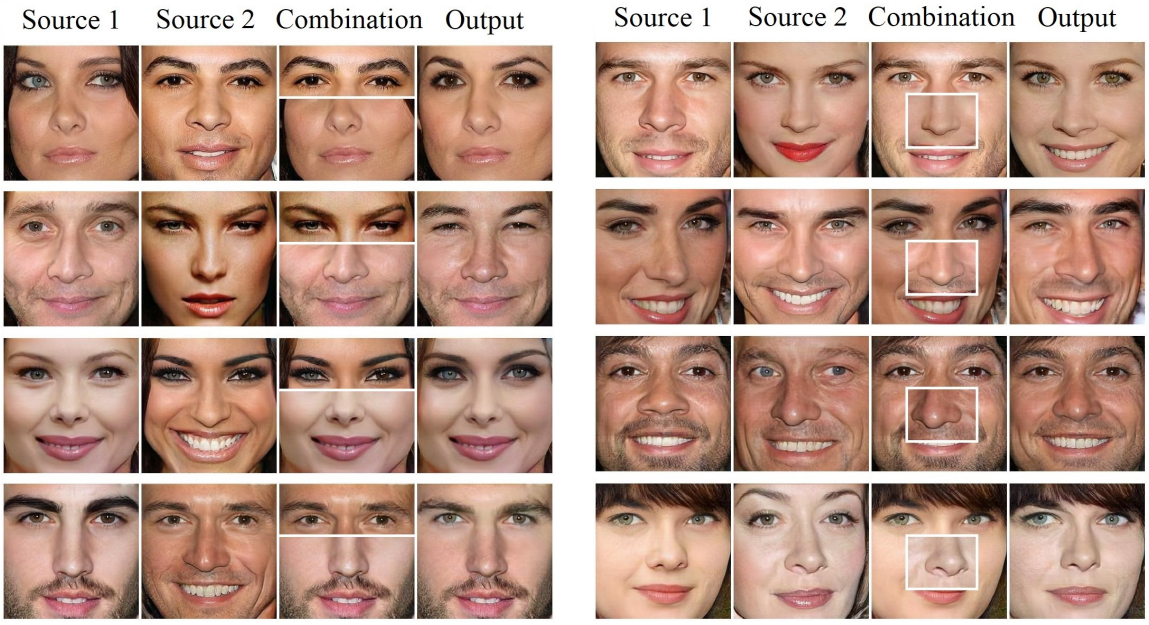
and background source). As mentioned previously, the disentanglement of concepts achieved by the proposed two-part compositional GAN is somewhat similar to the disentanglement achieved by the autoencoder proposed in Chapter 3. However, the disentanglement of color themes seems to be different between the two models; The two-part compositional GAN model disentangles the face part along with its colors from the hair and background part along with their colors. Meanwhile, the deep autoencoder introduced in Chapter 3 disentangles the color themes of the entire image into a single third group. Therefore, the autoencoder makes it possible to control the colors in the entire image. However, it is not possible to control the color of face only or the color of hair & background only.

4.3.2 Four-Part Compositional GAN

The four-part compositional GAN model is trained to generate face images as a composition of four parts; 1) eyes, 2) nose, 3) mouth, and 4) Jaw & cheeks. As a result, it makes it possible to modify the generated faces in a controlled way. It means that it is possible to modify one of these four components without changing the others. Figure 4.5 displays examples of faces generated by the four-part compositional GAN model. In each row of the figure, the *Source1* face is generated by a random sample from latents priors $\{z_{1,s1}, z_{2,s1}, z_{3,s1}, z_{4,s1}\}$ and the *Source2* face is generated by another random sample from latent priors $\{z_{1,s2}, z_{2,s2}, z_{3,s2}, z_{4,s2}\}$. The *Combination* image displays an illustration of how parts from the two source faces are combined to create the *Output* face. More precisely, the *Output* face in each row of Figure 4.5 (a) is synthesised by the generator of the four-part compositional GAN model given latent representation z_1 (eyes) from *Source2* and every other latent representation from *Source1*. Similarly, the *Output* face in each row of Figure 4.5 (b) is synthesised by the generator given latent representation z_2 (nose) from *Source2* and every other latent representation from *Source1*. Moreover, the *Output* face in each row of Figure 4.5 (c)

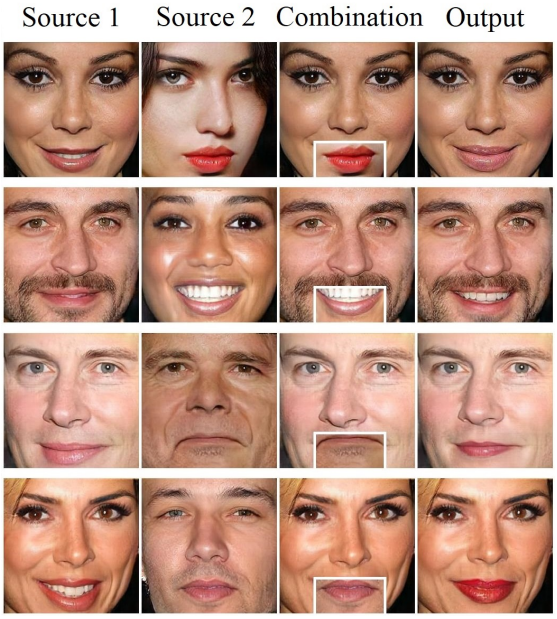
is synthesised by the generator given latent representation z_3 (mouth) from *Source2* and every other latent representation from *Source1*. Finally, the *Output* face in each row of Figure 4.5 (d) is synthesised by the generator given latent representation z_4 (jaw & cheeks) from *Source2* and every other latent representation from *Source1*.

As it can be seen from the figure, the latent representation z_1 of the four-part compositional GAN model captures the shape of the eyes and eyebrows. However, the color of skin and general lighting of the image remains unchanged after replacing z_1. The latent representation z_2 captures the shape of the nose. Additionally, since the part represented by z_2 is located at the center of the image, it borders with all the other important facial components. Therefore, it can be seen from the figures that it has a more global influence on the output image. For example, latent representation z_2 dominates the colors of skin, lips and eyes, as well as the gender in the *Output* face. The latent representation z_3 captures the shape of the mouth. Similar to z_1, the latent prior z_3 seems to have a more local effect on the output image. Additionally, it seems to influence the lower lip more than the upper lip in the generated faces. This is reasonable since the upper lip falls on the boundary region between the parts represented by z_2 and z_3. As a result, the latent representation z_2 also contributes to the synthesis of the upper lip. An extensive discussion of how border/interlocking regions between parts are influenced by more than one latent representation is provided in the following sections. Lastly, latent representation z_4 captures the shape of the jaw and the cheeks. Similar to z_1 and z_3, it has a more local influence on the image. Although, it seems to slightly influence the shape of eyes as it borders with the part represented by z_1.

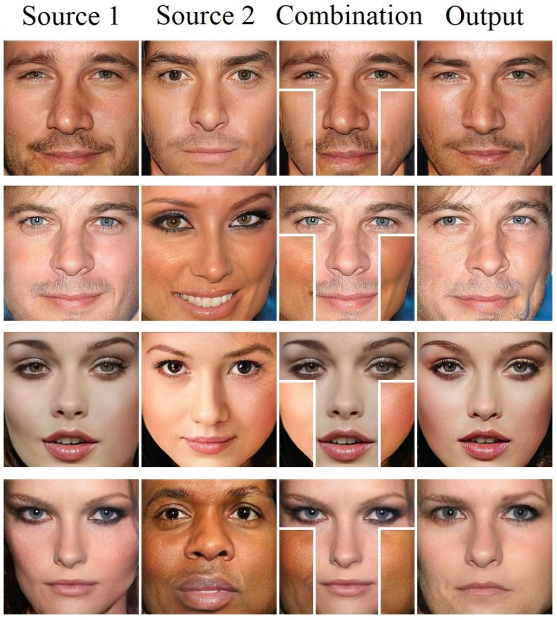


(a) z_1

(b) z_2



(c) z_3



(d) z_4

Figure 4.5: Examples of generated faces by the four-part compositional GAN model. In each row, the *Source1* face is generated given one set of latent representations, the *Source2* face is generated given another set of latent representations, and the *Output* faces are generated given one latent representation from the *Source2* and every other latent representation from the *Source1*.

4.4 Quantitative Analysis

4.4.1 Photorealism

An important criteria for evaluating a generative model is the measure of photorealism in the synthesised images. As it was discussed in previous chapters, FID is a measure of the quality of generated images. Briefly, a lower FID score indicates synthesised images that are more realistic and therefore closer to real images. Table 4.3 reports the FID scores for our compositional GAN models and compares them with the benchmark methods; PGGAN [49], StarGAN [38], ELEGANT [37], Pix2PixHD [36], SPADE [39], MaskGAN [40], and Latent Regression [55]. A brief review of these benchmark methods was provided in Chapter 2.

It is evidenced in Table 4.3 that PGGAN generated images achieve the highest level of realism (lowest FID scores). However, PGGAN is a standard GAN and neither achieves any type of disentanglement in its latent representation space nor has the ability to locally control the generated images. Our compositional GAN models achieve better FID scores compared to the other benchmark methods with the exception of PGGAN. As it was explained in Chapter 2, StarGAN, ELEGANT and Pix2PixHD are conditional GANs trained for domain translation. Meanwhile, SPADE and MaskGAN are conditional GANs trained for converting a semantic mask into a realistic image. It can be concluded that the extra constraint that is placed on the image synthesis process of conditional GANs results in the decrease in the realism of generated images. Contrary to the conditional GAN models, the compositional GAN model proposed here does not generate images conditioned on a domain or a semantic segmentation mask and therefore is able to achieve a FID score comparable to the standard PGGAN model. The Latent Regression method utilises the fixed pre-trained PGGAN model for implementing its compositions and for locally manipulating the synthesised faces. Since our compositional GAN models also use PGGAN as the

Method	FFHQ Original	CelebA-HQ Original	CelebA-HQ Cropped
PGGAN [49]	**13.32**	**9.61**	**8.51**
StarGAN [38]	–	40.61	–
ELEGANT [37]	–	55.43	–
Pix2PixHD [36]	–	54.68	–
SPADE [39]	–	46.17	–
MaskGAN [40]	–	37.14	–
Latent Regression [55]	24.09	15.35	–
Compositional GAN (ours)			
Two-Part Composition	14.25	10.48	–
Four-Part Composition	–	9.76	9.08

Table 4.3: FID scores (lower is better) for generated images by the proposed Compositional GAN models and the benchmark methods; StarGAN [38], ELEGANT [37], Pix2PixHD [36], SPADE [39], MaskGAN [40] and Latent Regression [55]. Models are trained on either FFHQ dataset or CelebA-HQ dataset and using images of 256×256 resolution.

backbone for their architecture and training process, it is interesting to compare the difference in FID scores between PGGAN and the Latent Regression method and also between PGGAN and our Compositional GAN models. As it is shown in the table, the difference in FID score between PGGAN and the proposed compositional GAN models is not significant. This means that the modifications of a standard GAN model done by our compositional GAN model do not result in significant decrease in quality and realism of the generated images. However, the decrease in the realism of generated images by the Latent Regression method is more significant.

4.4.2 Locality and Parts Independence

Figures 4.6 (a) shows an illustration of the first block of the generator network of our four-part compositional GAN model. The block size is $8 \times 8 \times n_{channels}$ and is being scanned by a convolutional filter of size 3×3. Three example positions of the filter are illustrated in the figure. As evidenced, there are regions that are only affected by one latent prior. For example the pixel labeled 1 in the figure is generated influenced by z_1 only. In contrast, some regions are generated with more than one latent prior influencing them. For example, the pixel labeled 2 is generated influenced by z_2 and z_4, and the pixel labeled 3 is generated influenced by z_2, z_3, and z_4. The pixels that are influenced by more than one latent prior create the boundary regions connecting the parts. Such boundary regions are in fact responsible for learning the interlocking of different parts. In Figure 4.6 (b) and (c) three distinct regions are displayed for each latent prior of the compostional GAN models; The area highlighted in green is the area where that latent prior is responsible to represent. The area highlighted in yellow is the interlocking area between that latent prior and the other latent priors. The area highlighted in blue is the area completely outside the influence of that latent prior. Furthermore, Figure 4.7 (a) and (b) display the Mean squared-error (MSE) boxplots for each latent representation computed in these three distinct regions. MSEs are computed between 50,000 generated images and their edited counterparts by replacing that latent prior only. These plots clearly exhibit that for all latent priors the inside area has the largest MSE, the outside area has the lowest MSE, and the interlocking area's MSE is in between. In other words, the plots display a decrease in MSE from the inside area, to the interlocking area, and a decrease in MSE from the interlocking area to the outside area.

To further evaluate the influence of each latent representation on the generated

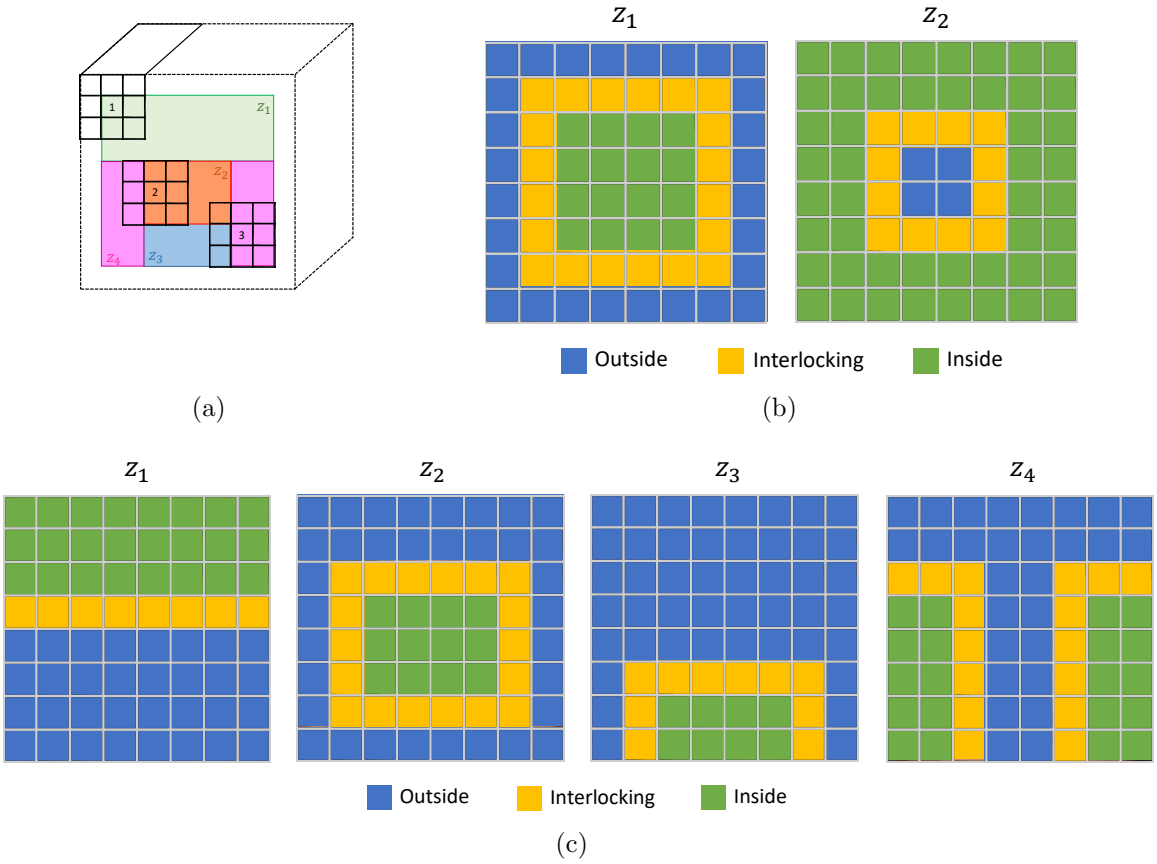
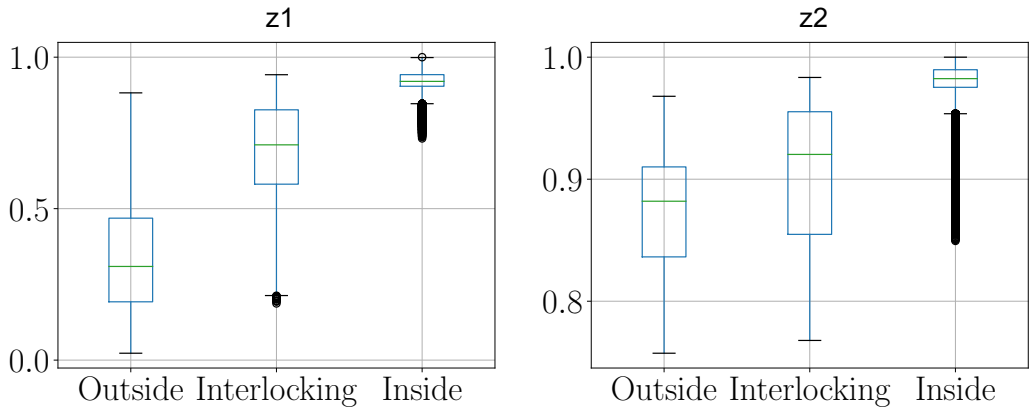
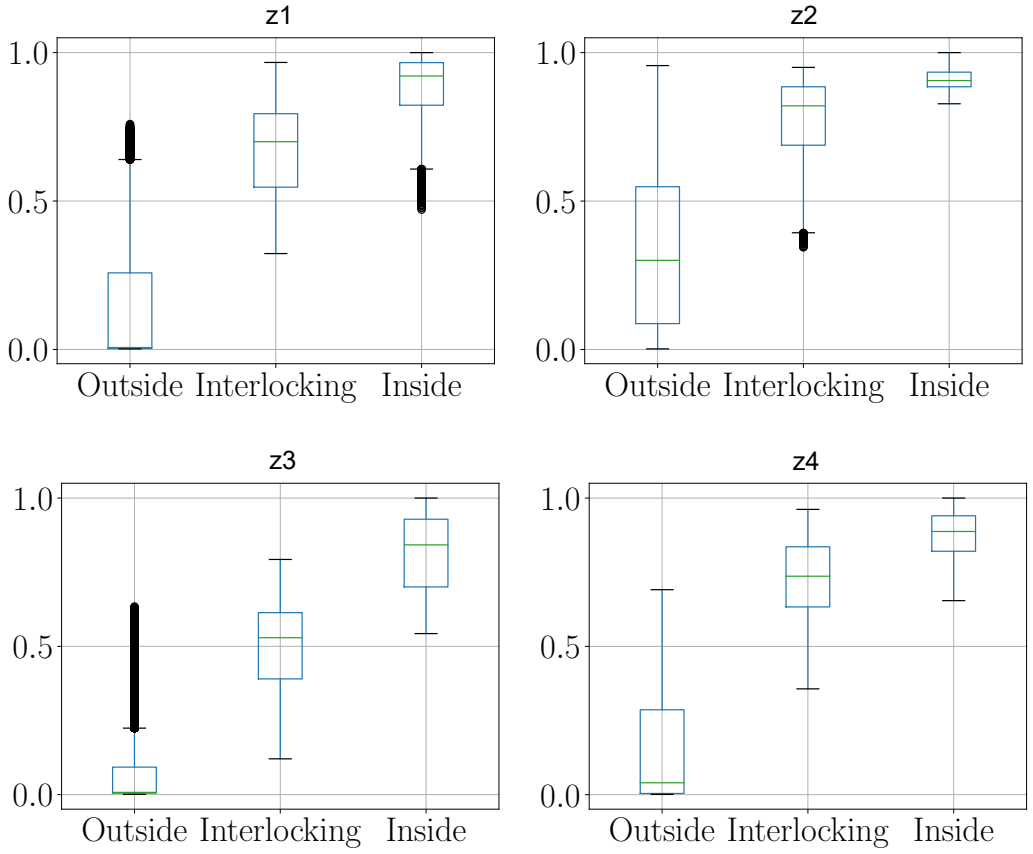


Figure 4.6: (a) Illustration of a convolution operation with filter size of 3×3 and SAME padding scanning an $8 \times 8 \times n_{channels}$ block with four distinct parts. 1, 2, and 3 respectively display examples of pixels influenced by only one latent prior, two different latent priors, and three different latent priors. (b) Three distinct regions for each latent representation of the two-part compositional GAN model. (c) Three distinct regions for each latent representation of the four-part compositional GAN model. The displayed faces in (b) and (c) are samples from FFHQ [11] and CelebA-HQ [49] datasets, respectively.



(a)



(b)

Figure 4.7: (a) The boxplots of MSEs in three distinct regions for each latent representation of the two-part compositional GAN model. (b) The boxplots of MSEs in three distinct regions for each latent representation of the four-part compositional GAN model.

images, we quantify and evaluate the parts independence [55]. For each latent representation z_t, we generate N images x_{n,z_t} with every latent representation fixed except for z_t. We then measure how much changing this latent influences each pixel location of the image by computing the normalized pixel-wise standard deviation among these N images as $v_{z_t} = \sigma_{z_t} / \sum \sigma_{z_t}$ where $\sigma_{z_t} = \sqrt{\mathbb{E}_n[(x_{n,z_t} - \mathbb{E}_n[x_{n,z_t}])^2]}$. Lastly, for each latent representation z_t, we measure independence as the average variation in the outside and interlocking regions of z_t that results from changing z_t. We repeat this experiment 100 times using $N = 20$ samples.

Method	**Parts Independence**			
			Face	**Hair&Background**
Two-Part Compositional GAN			0.115	0.309
	Eyes	**Nose**	**Mouth**	**Cheeks&Jaw**
Four-Part Compositional GAN	0.091	0.097	0.0359	0.088

Table 4.4: A measure of parts independence achieved by the two-part compositional GAN model and the four-part compositional GAN model (a lower value means more independent).

Method	**Overall Independence**
Two-Part Compositional GAN	0.212
Four-Part Compositional GAN	**0.078**
Latent Regression [55]	
based on PGGAN	0.093
based on StyleGAN	0.105

Table 4.5: The overall independence value achieved by the two-part compositional GAN model, the four-part compositional GAN model, and the Latent Regression method [55] (a lower value means more independent). The overall independence value for a model is computed by averaging the parts independence values.

Table 4.4 reports these computed independence values for individual parts/representations for our compositional GAN models. Additionally, Table 4.5 shows the overall independence values achieved by the two-part compositional GAN model, the four-part compositional GAN model, and the Latent Regression method [55]. It is evident from the table that the four-part compositional GAN model achieves better parts independence in comparison to the other methods. In contrast, the two-part compositinal GAN model has the most leakage among the two parts. Specifically changing the latent prior representing hair and background seems to have a more global effect on the image. This is in agreement with the qualitative results of Figure 4.4 and the boxplots shown in Figure 4.7 (a). One reason for this global effect is that the general lighting of the image is influenced by this latent representation. Additionally, this latent represents a large portion of the synthesised image and therefore its influence on the entire image is more dominant. In comparison, in the four-part compositional GAN model each latent representation represents a relatively small area of the output image and therefore the model is able to achieve a better parts independence.

4.4.3 Transferring Smile vs Preserving Identity

It was shown in the previous section that our four-part compositional GAN model achieves a high level of parts independence. As a further evaluation, in this section we perform an experiment of transferring *Smiling* from one image to another using the four-part compositional GAN model. The reason for selecting the *Smiling* attribute for this experiment is that it is recognized by the benchmark conditional GAN methods that *Smiling* is one of the most challenging attributes to transfer from one face image to another while preserving the identity-related features [40].

In order to label the generated images by our compositional GAN model, we use an auxiliary classification network trained for the binary attribute *Smiling* using the

Method	**Smile Transfer Accuracy (%)**	**Identity Preserving Accuracy (%)**
StarGAN [38]	**92.5**	–
ELEGANT [37]	72.8	–
Pix2PixHD [36]	78.5	58.46
SPADE [39]	73.8	70.77
MaskGAN [40]	77.3	76.41
Four-Part Compositional GAN (ours)	76.72	**85.08**

Table 4.6: A comparison between the proposed four-part compositional GAN model and the benchmark methods; StarGAN [38], ELEGANT [37], Pix2PixHD [36], SPADE [39], and MaskGAN [40], in terms of their ability to modify an image by adding a smile while preserving the identity in the modified face.

CELEBA-HQ dataset [11]. More specifically, we generate thousands of images using the generator of our four-part compositional GAN model and classify them using the auxiliary classifier. From these classified images, we select 400 pairs of images and store their latent priors. In each pair one latent is labeled as not-smiling and one latent is labeled as smiling by the classifier with a high confidence. Unlike MaskGAN, we do not require the images in each pair to have similar head poses. We then modify the latent labeled as not-smiling in each pair by replacing its z_3 with the z_3 of the smiling latent in that pair (the latent z_3 represents the mouth part in our four-part compositional GAN model). We then pass these modified latents trough the generator network, synthesise the modified faces, and classify them using the auxiliary classifier. The percentage of the modified faces for which the classifier's prediction is changed from not-smiling to smiling after replacing z_3 is reported in Table 4.6 as smile transfer accuracy. Additionally, we conduct a face verification experiment using ArcFace [77] model which has an accuracy of %99.52 in face verification on LFW dataset. For this experiment we use the 400 pairs of unmodified not-smiling faces and modified

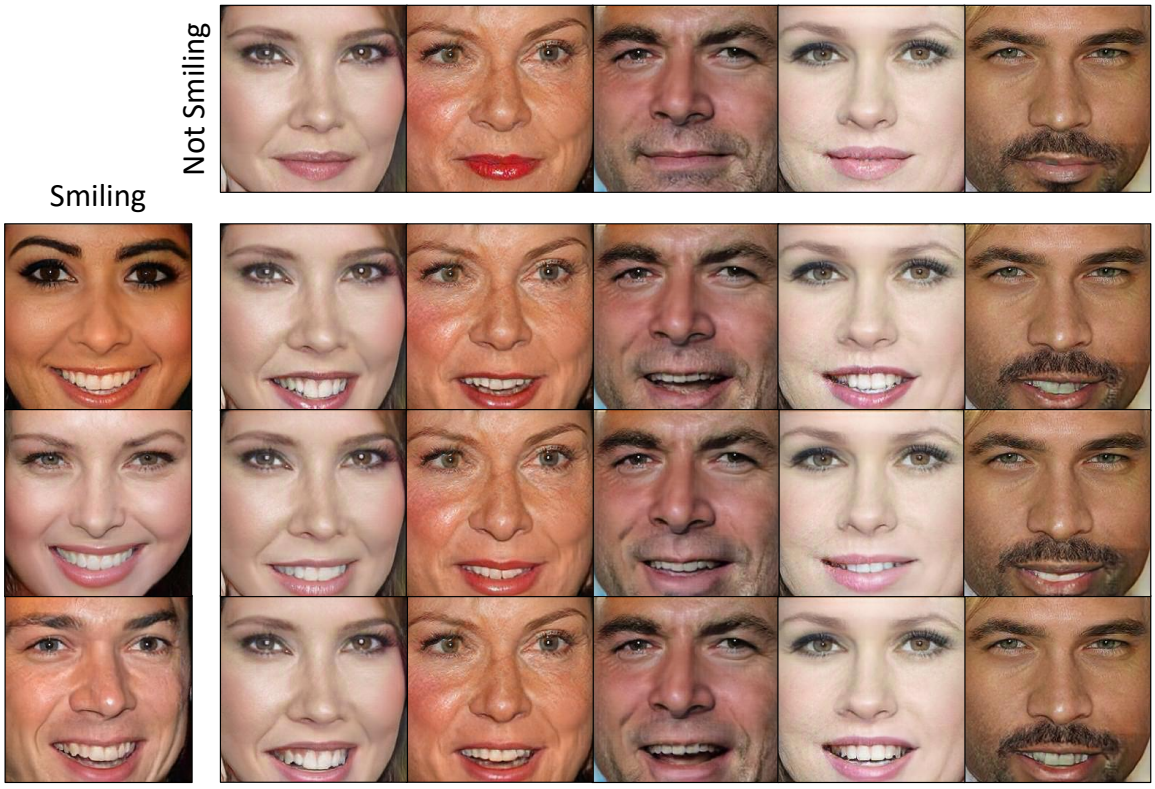


Figure 4.8: Examples of transferring smile from one image to another by the four-part compositional GAN model. The smile transfer is achieved by replacing the latent representation z_3 of a *not smiling* target face with the z_3 of a *smiling* source face.

faces after replacing z_3. The percentage of these face pairs for which ArcFace predicts matching identity is reported in Table 4.6 as identity preserving accuracy.

As it can be seen from the table, StarGAN achieves the highest accuracy in transferring the *Smiling* attribute. However, unlike the other methods presented in the table, StarGAN cannot generate images by exemplars. It means it is able to translate an image from the *Not Smiling* domain to the *Smiling* domain, but it cannot generate different versions of the smiling face using different *Smiling* source images. Our compositional GAN model achieves an accuracy close to MaskGAN and Pix2PixHD methods and higher than ELEGANT and SPADE methods in transferring *Smiling*

attribute from one face to another. Moreover, our model achieves a superior accuracy in maintaining the identity in the modified faces in comparison to the other benchmark models for which this metric is available.

Figure 4.8 displays examples of transferring smile from *Smiling* source images to *Not Smiling* target images. The edited faces in the figure get their z_3 from a *Smiling* face image and every other latent vector from a *Not Smiling* face image. It is evidenced in the figure that our four-part compositional GAN is able to edit a face image from *Not Smiling* to *Smiling* while maintaining the identity of the face. Additionally, the model is able to generate slightly different smiling versions for the same target face depending on the faces used as the source of smile.

4.4.4 The Relationship Between Parts Latent Representations and High-level Facial Concepts

In this section we continue with our experiments using the four-part compositional GAN model which was shown to achieve a high level of parts independence. Since the four-part compositional GAN learns distinct representations for distinct facial components, it would be interesting to look deeper into the relationship between the representations of these smaller components and the more abstract and high-level facial concepts such as expression, gender, age, and identity. The experiments done in this section follow the same steps as the experiment in the previous section. However, instead of considering only one attribute, here we repeat the experiment for three attributes. These attributes include *Smiling*, *Male*, and *Young* and are representing the following high-level facial concepts: expression, gender, and age. Additionally, instead of measuring the number of identities preserved after replacing a latent representation, here we measure the number of identities that are transformed. In other words, we are interested in learning which parts latent representations capture the identity

	Concept Transfer Accuracy(%)			
Representation	**Smiling**	**Male**	**Young**	**Identity**
z_1	3.54	7.14	53.23	68.18
z_2	94.32	92.86	66.67	81.82
z_3	76.72	3.76	7.52	14.92
z_4	15.0	21.43	83.34	30.91

Table 4.7: The impact of parts representations in determining smiling, gender, age, and identity in the generated faces measured for each latent representation of the four-part compositional GAN model.

related information the most (and therefore replacing that latent representation will result in changing identity). Finally, we repeat the experiment for all the parts latent representations and not only z_3. The results of this experiment are reported in Table 4.7.

Looking at the row of the table, it can be concluded that replacing latent z_1 influences identity and age more than it influences expression and gender. This means that identity recognition is highly influenced by the information embedded in eyes. Additionally, one important indicator of old age is the appearance of wrinkles on the skin around the eyes. Therefore, it seems reasonable that replacing z_1 has a significant effect on transforming age in the target face. Lastly, replacing z_1 does not seem to play a significant role in determining expression and gender of the face. The latent representation z_2 seems to highly influence all the high-level facial concepts. This is in agreement with the results of Figure 4.5 (b) where we discussed that the part represented by z_2 is located at the center of the face and therefore borders with all the other important facial components. As a results, changing z_2 has a more global influence on the output face. Furthermore, it was shown in Table 4.4 and the boxplots of Figure 4.7 (b) that z_2 has the least parts independence and the highest MSE in outside and interlocking regions among the parts of four-part compositional

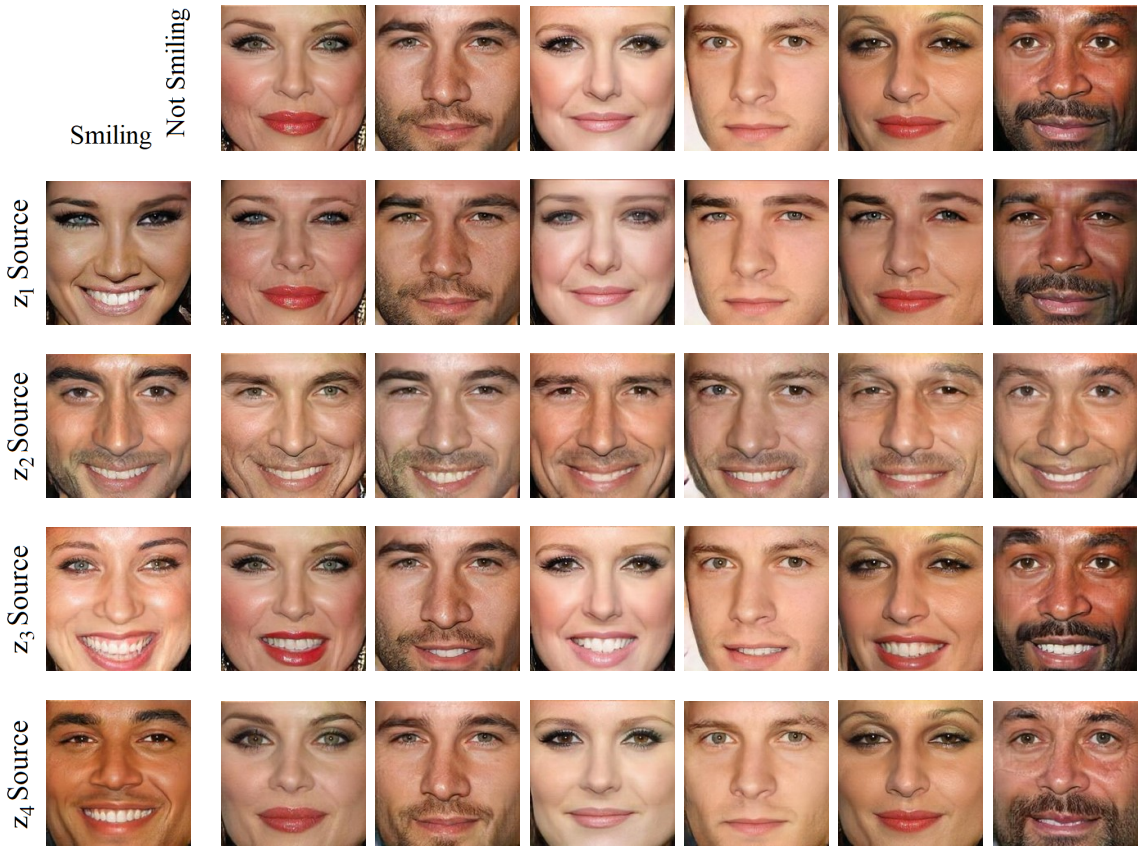


Figure 4.9: Examples of faces generated by copying a single latent representation from a smiling source image to a not-smiling target image.

GAN. The latent representation z_3 influences smiling/expression the most. This is reasonable since z_3 is responsible to represent the mouth area. Additionally, as it was discussed before, latent representation z_3 has the most local effect on the generated face among the parts of four-part compositional GAN. As a result, it is understandable that it does not significantly influence the other high-level facial concepts. The latent representation z_4 seems to influence age more than the other high-level facial concepts. This is again due to the fact that a sign of old age is the appearance of wrinkles and losing firmness in the skin of the cheeks and jaw area. As a result, replacing z_4 can influence the skin in these areas and therefore transform the age in the target face.

It is also possible to discuss the columns of Table 4.7. It is shown in the first column

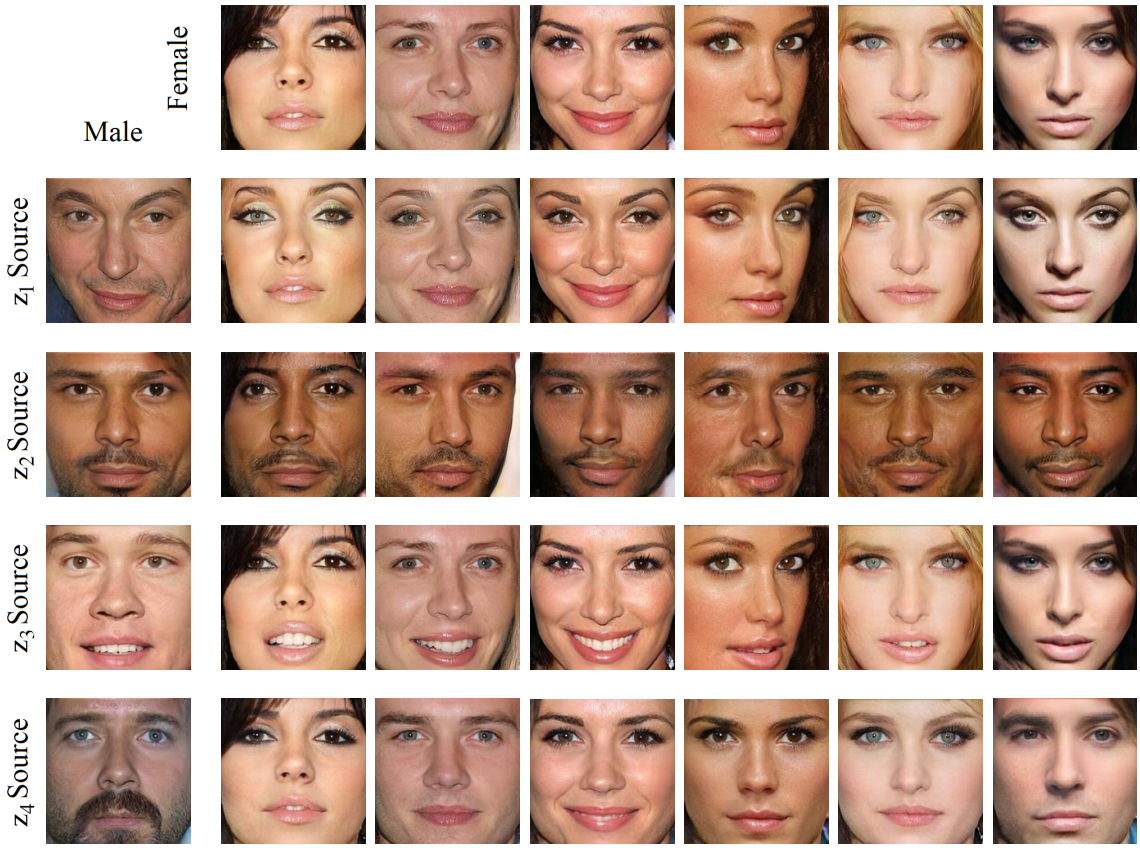


Figure 4.10: Examples of faces generated by copying a single latent representation from a male source image to a female target image.

that the concept/attribute of smiling is captured by z_2 and z_3 the most. Meanwhile, replacing z_1 has the least effect on smiling. The impact of replacing z_4 on smiling is also low but higher than the impact of z_1, since z_4 borders with the corners of the lips. Examples of faces generated by copying different parts latent representations from a smiling source image to a not smiling target image are displayed in Figure 4.9. The second column of Table 4.7 shows that the concept of gender is impacted most by replacing z_2. Meanwhile, changing z_1 or z_3 has the lowest impact on the gender in the generated face. The impact of z_4 on determining gender in the generated face is higher than z_1 and z_3. This could be due to the fact that the shape of jaw is usually different between males and females. Examples of faces generated by copying

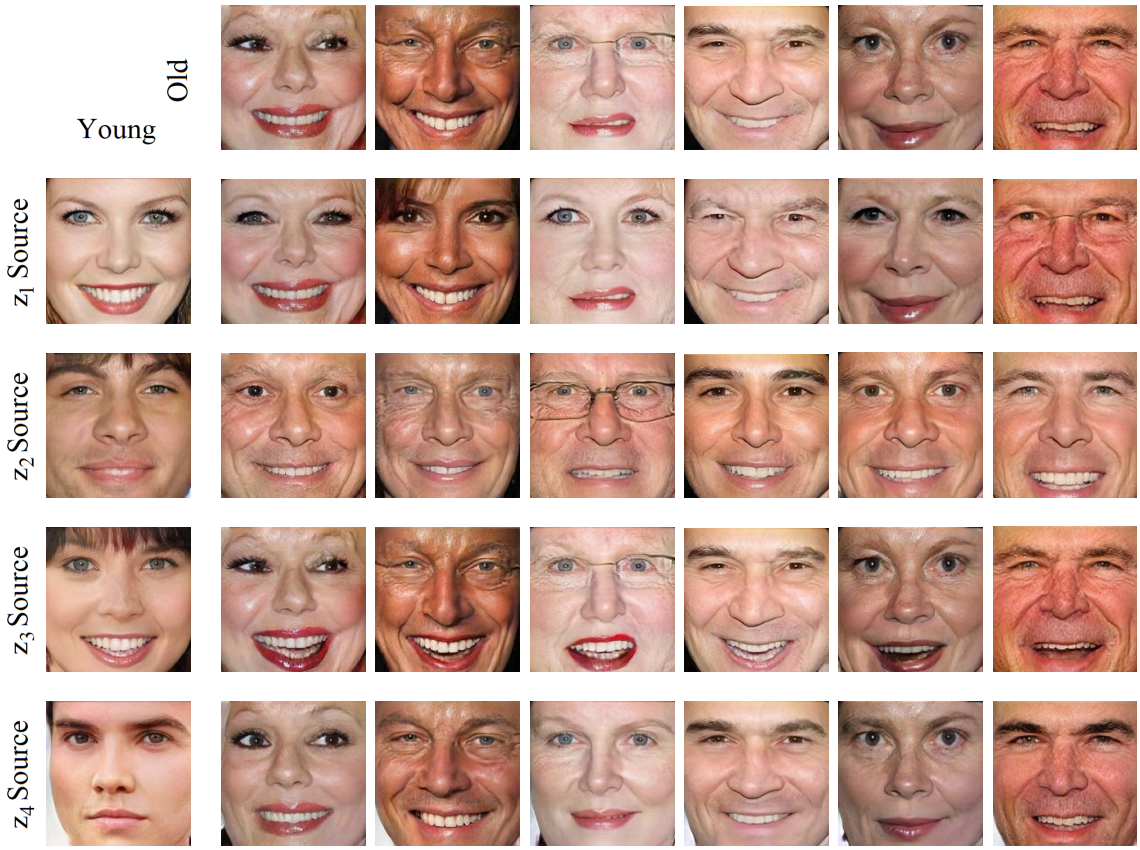


Figure 4.11: Examples of faces generated by copying a single latent representation from a young source image to an old target image.

different parts representations from a male source image to a female target image are displayed in Figure 4.10. Furthermore, the third column of Table 4.7 indicates that the concept of age is captured highly by z_1, z_2, and z_4. Since old age shows itself mainly through appearance of wrinkles in different parts of skin, it is not possible to capture the old age from one position only. More precisely, replacing different parts representations can influence the age of the generated face depending on where on the skin wrinkles are appearing the most. Examples of faces generated by copying different parts representations from a young source image to an old target image are displayed in Figure 4.11. Finally, the last column of the table shows that the concept of identity is captured most by z_1 and z_2. Therefore, changing z_1 and z_2

results in generating new identities. However, replacing z_3 or z_4 has a lower impact on determining the identity in the generated face.

4.5 Summary

In this chapter we presented a straightforward method to improve a standard GAN architecture such that the modified architecture is able to learn the distribution of face images as compositions of multiple smaller parts distributions. Each part is defined to only contain specific components of a face. Therefore, learning a separate distribution for each part is equivalent to disentangling these components in the representation space. We showed that the compositional GAN models are able to produce realistic high-quality face images by sampling from the learned distributions for parts and then generating the parts and piecing them together. Additionally, we demonstrated that the proposed compositional GAN models not only learn the representations for facial components but also the relations between them. Therefore, they can generate realistic whole faces given any combination of samples from components/parts prior distributions. This results in a more flexible and less entangled machine face perception and leads to easier generalization to examples outside training data. Furthermore, we showed through extensive qualitative and quantitative evaluations that the proposed compositional GANs are able to outperform benchmark methods in generating realistic face images while performing compositions in image domain and allowing for local control over generated faces.

Chapter 5

Conclusions and Future Work

In this thesis, we conducted two studies in order to examine the applicability of two unsupervised clues for learning a representation of face images in which different groups of facial concepts are disentangled. The first unsupervised clue is the scale at which a facial concept appears. The idea of scale-based disentanglement of facial concepts is inspired by the fact that different categories of concepts are encoded in different resolutions of a face image. The second unsupervised clue is the position within the face structure at which a facial component/concept is located. The idea of disentangling facial concepts based on position is inspired by the natural underlying structure that is present in all face images.

5.1 Conclusions

In the first study of this thesis:

- We proposed a deep autoencoder model that takes advantage of an architecture inspired by StyleGAN model and an adaptive resolution reconstruction loss. The adaptive resolution reconstruction loss is motivated by the fact that different resolutions of an image display facial concepts of different scales. Therefore,

it is possible to control the coarser concepts in a generated face image by enforcing a reconstruction loss on only the lower resolutions of that image. This new type of reconstruction loss facilitates learning a latent representation for real face images in which facial concepts are disentangled based on scale.

- We demonstrated that the proposed deep autoencoder moves from a latent representation for autoencoders in which all facial concepts are captured in a single large group towards learning a latent representation in which facial concepts are separated into three distinct groups. These three groups were referred to as coarse scale, middle-scale, and fine-scale. We showed that the coarse-scale group captures the information related to head pose, shape of hair, background, shape of face (i.e. hairline, ears, chin, and jaws), neck, and collar area. The middle-scale group captures all the identity related features including the shape and size of eyes, eyebrows, nose, lips, and cheeks. Additionally, the texture of hair, the texture of skin, whether a person is wearing makeup or not, and facial expressions such as smiling are captured by this group as well. Finally, the fine-scale group captures all the colors from the face image including background colors, color of hair, color of skin, color of eyes, color of eyebrows, color of lips, and makeup colors (if a person is wearing makeup).

- We demonstrated, through extensive qualitative and quantitative evaluations, that the proposed deep autoencoder is successful in disentangling the facial concepts associated with specific scales and in transferring these scale-based concepts from one face image to another. Moreover, it was shown that by including a discriminator network along with an adversarial loss with a small weight, we can reduce the blurriness associated with autoencoder generated images. As a result, the proposed deep autoencoder is able to outperform benchmark methods in generating faithful and high quality reconstructions of

real face images.

In the second study of this thesis:

- We proposed a straightforward method to improve a standard GAN architecture such that the modified architecture is able to learn the distribution of face images as compositions of multiple smaller parts distributions. This method was referred to as compositional GAN. The proposed compositional GAN defines each part to only contain specific components of a face. Therefore, learning a separate distribution for each part is equivalent to disentangling these components in the representation space. Such compositional GAN model is able to produce realistic high-quality face images by sampling from the learned distributions for facial components and then generating the parts and piecing them together.

- Two compositional GAN models were developed; 1) A two-part compositional GAN for learning the representation of face images composed of two parts; one representing the face and the other representing hair & background. 2) A four-part compositional GAN for learning the representation of cropped faces composed of four facial components; eyes, nose, mouth, and jaw & cheeks. We showed, through extensive qualitative and quantitative evaluations, that the proposed compositional GAN moves from a latent representation for GANs in which all facial concepts are captured in a single large group towards learning multiple latent representations for different groups of facial concepts/components.

- For the four-part compositional GAN, we studied the impact of parts representations in determining high-level facial concepts such as facial expression, gender, age, and identity. More precisely, we showed that replacing latent z_1

(eyes part) influences identity and age more than it influences expression and gender. Meanwhile, replacing latent representation z_2 (nose part) has a more global influence on the output face and influences all the high-level facial concepts. In other words, this part has the least disentangled representation as it is located at the center of the face and borders with all the other important facial components. Moreover, it was shown that replacing the latent representation z_3 (mouth part) influences smiling/expression the most. Finally, replacing the latent representation z_4 (jaw & cheeks part) influences age more than the other high-level facial concepts.

- With the help of various qualitative results and quantitative analysis, we showed that the proposed compositional GAN models are able to outperform benchmark methods in generating realistic face images while performing compositions in image domain and allowing for local control over generated faces. We demonstrated that the compositional GAN models not only learn the representations for facial components but also the relations between them. Therefore, it can generate realistic whole faces given any combination of samples from components/parts distributions. This results in a more flexible and less entangled machine face perception and leads to easier generalization to examples outside training data.

5.2 Future Work

One general future direction for improving autoencoders and GANs is to find new ways of defining reconstruction loss for autoencoders and new ways of adding hierarchy and structure into GANs. This can result in different ways of extracting and disentangling concepts from natural images. Furthermore, a possible future direction to improve the proposed deep autoencoder model in this thesis it to move

from traditional autoencoders to VAEs. This would make it possible to incorporate structure into the learned latent representations and to achieve within-scale disentanglement/categorization of facial concepts. Similarly, it is possible to improve the proposed compositional GAN model by adding structure in the latent representations priors. This structure can be selected based on the content each latent representation is responsible to represent.

As discussed in Chapter 2, a powerful trend in deep generative models is combining autoencoders with GANs in one framework. The combination makes it possible to take advantage of the benefits of the two methods while avoiding their drawbacks. The deep autoencoder proposed in the first study of this thesis takes advantage of such combination. A GAN is integrated into its framework which helps reduce the blurriness and increases the quality of autoencoder generated images. However, the compositional GAN proposed in the second study of this thesis is purely a GAN model. As a result, the proposed model is not able to reconstruct real face images and is not able to compute the latent representations for parts of real face images. One future direction to pursue would be to integrate an autoencoder into the compositional GAN model. Such model can be trained using a balanced combination of reconstruction loss and adversarial loss, or higher weights may be assigned to either of the two losses. This goal is to build and train a model that is able to modify real existing faces locally and in controlled ways. Additionally, the introduction of an autoencoder into the framework might result in enriching the disentanglement as autoencoders are helpful in inferring interpretable latent representations and capturing structures. However, the inclusion of an autoencoder may reduce the quality of generated images slightly.

Lastly, the two studies of this thesis can be combined together into one framework. More precisely, we can develop a representation learner of face images that achieves

both scale-based and position-based disentanglement of facial concepts in its representation space. This may be achieved by introducing a framework that samples from parts latent priors and then generates the parts and pieces them together. Meanwhile, with the help of a StyleGAN architecture and an AR reconstruction loss the model learns to generate the image parts at different scales and learns to disentangle the parts information based on scale.

Similar to other deep generative model, the methods proposed in this thesis can be used for numerous applications such as super resolution, colorization, and image completion for face images. In short, the methods proposed in this thesis can be considered as part of the broad research efforts for enabling computers to learn a more comprehensive and less entangled representations for face images which has numerous potential applications. For example, computers can become much better in person re-identification through different appearance changes. This has great potential for security purposes such as disguise detection and disguise-invariant face recognition. Additionally, computers will be able to understand and notice changes in appearance, emotions and expressions and respond/behave accordingly. This can dramatically improve the field of Human Machine Interactions (HCI).

List of References

[1] R. VanRullen, "Perception science in the age of deep neural networks," *Frontiers in psychology*, vol. 8, p. 142, 2017.

[2] J. V. Haxby, E. A. Hoffman, and M. I. Gobbini, "The distributed human neural system for face perception," *Trends in cognitive sciences*, vol. 4, no. 6, pp. 223–233, 2000.

[3] J. V. Haxby, E. A. Hoffman, and M. I. Gobbini, "Human neural systems for face recognition and social communication," *Biological psychiatry*, vol. 51, no. 1, pp. 59–67, 2002.

[4] V. Bruce and A. Young, "Understanding face recognition," *British journal of psychology*, vol. 77, no. 3, pp. 305–327, 1986.

[5] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 815–823, 2015.

[6] D. Y. Tsao and M. S. Livingstone, "Mechanisms of face perception," *Annu. Rev. Neurosci.*, vol. 31, pp. 411–437, 2008.

[7] B. Rossion, "Understanding face perception by means of human electrophysiology," *Trends in cognitive sciences*, vol. 18, no. 6, pp. 310–318, 2014.

[8] Z. Yan and X. S. Zhou, "How intelligent are convolutional neural networks?," *arXiv preprint arXiv:1709.06126*, 2017.

[9] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[10] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[11] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4401–4410, 2019.

[12] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.

[13] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, pp. 2672–2680, 2014.

[14] M. Abdolahnejad and P. X. Liu, "Deep learning for face image synthesis and semantic manipulations: a review and future perspectives," *Artificial Intelligence Review*, vol. 53, no. 8, pp. 5847–5880, 2020.

[15] M. Abdolahnejad and P. X. Liu, "A deep autoencoder with novel adaptive resolution reconstruction loss for disentanglement of concepts in face images," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–13, 2022.

[16] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[17] D. J. Rezende, S. Mohamed, and D. Wierstra, "Stochastic backpropagation and approximate inference in deep generative models," *arXiv preprint arXiv:1401.4082*, 2014.

[18] B. Cheung, J. A. Livezey, A. K. Bansal, and B. A. Olshausen, "Discovering hidden factors of variation in deep networks," *arXiv preprint arXiv:1412.6583*, 2014.

[19] J. M. Susskind, A. K. Anderson, and G. E. Hinton, "The toronto face database," *Department of Computer Science, University of Toronto, Toronto, ON, Canada, Tech. Rep*, vol. 3, 2010.

[20] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker, "Multi-pie," *Image and Vision Computing*, vol. 28, no. 5, pp. 807–813, 2010.

[21] H. Edwards and A. Storkey, "Towards a neural statistician," *arXiv preprint arXiv:1606.02185*, 2016.

[22] J. Klys, J. Snell, and R. Zemel, "Learning latent subspaces in variational autoencoders," in *Advances in Neural Information Processing Systems*, pp. 6444–6454, 2018.

[23] D. Bouchacourt, R. Tomioka, and S. Nowozin, "Multi-level variational autoencoder: Learning disentangled representations from grouped observations," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[24] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, "beta-vae: Learning basic visual concepts with a constrained variational framework.," *Iclr*, vol. 2, no. 5, p. 6, 2017.

[25] T. Q. Chen, X. Li, R. B. Grosse, and D. K. Duvenaud, "Isolating sources of disentanglement in variational autoencoders," in *Advances in Neural Information Processing Systems*, pp. 2610–2620, 2018.

[26] T. Cemgil, S. Ghaisas, K. Dvijotham, S. Gowal, and P. Kohli, "The autoencoding variational autoencoder," *Advances in Neural Information Processing Systems*, vol. 33, pp. 15077–15087, 2020.

[27] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.

[28] J. Gauthier, "Conditional generative adversarial nets for convolutional face generation," *Class Project for Stanford CS231N: Convolutional Neural Networks for Visual Recognition, Winter semester*, vol. 2014, no. 5, p. 2, 2014.

[29] G. Antipov, M. Baccouche, and J.-L. Dugelay, "Face aging with conditional generative adversarial networks," in *2017 IEEE International Conference on Image Processing (ICIP)*, pp. 2089–2093, IEEE, 2017.

[30] M.-Y. Liu and O. Tuzel, "Coupled generative adversarial networks," in *Advances in neural information processing systems*, pp. 469–477, 2016.

[31] T. Kim, M. Cha, H. Kim, J. K. Lee, and J. Kim, "Learning to discover cross-domain relations with generative adversarial networks," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 1857–1865, JMLR. org, 2017.

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

[33] Z. Yi, H. Zhang, P. Tan, and M. Gong, "Dualgan: Unsupervised dual learning for image-to-image translation," in *Proceedings of the IEEE international conference on computer vision*, pp. 2849–2857, 2017.

[34] A. Royer, K. Bousmalis, S. Gouws, F. Bertsch, I. Mosseri, F. Cole, and K. Murphy, "Xgan: Unsupervised image-to-image translation for many-to-many mappings," *arXiv preprint arXiv:1711.05139*, 2017.

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

[36] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro, "High-resolution image synthesis and semantic manipulation with conditional gans," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8798–8807, 2018.

[37] T. Xiao, J. Hong, and J. Ma, "Elegant: Exchanging latent encodings with gan for transferring multiple face attributes," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 168–184, 2018.

[38] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "Stargan: Unified generative adversarial networks for multi-domain image-to-image translation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8789–8797, 2018.

[39] T. Park, M.-Y. Liu, T.-C. Wang, and J.-Y. Zhu, "Semantic image synthesis with spatially-adaptive normalization," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2337–2346, 2019.

[40] C.-H. Lee, Z. Liu, L. Wu, and P. Luo, "Maskgan: Towards diverse and interactive facial image manipulation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5549–5558, 2020.

[41] A. Pumarola, A. Agudo, A. M. Martinez, A. Sanfeliu, and F. Moreno-Noguer, "Ganimation: Anatomically-aware facial animation from a single image," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 818–833, 2018.

[42] E. Sanchez and M. Valstar, "Triple consistency loss for pairing distributions in gan-based face synthesis," *arXiv preprint arXiv:1811.03492*, 2018.

[43] S. Suwajanakorn, S. M. Seitz, and I. Kemelmacher-Shlizerman, "Synthesizing obama: learning lip sync from audio," *ACM Transactions on Graphics (TOG)*, vol. 36, no. 4, p. 95, 2017.

[44] H. Kim, P. Carrido, A. Tewari, W. Xu, J. Thies, M. Niessner, P. Pérez, C. Richardt, M. Zollhöfer, and C. Theobalt, "Deep video portraits," *ACM Transactions on Graphics (TOG)*, vol. 37, no. 4, p. 163, 2018.

[45] R. Natsume, T. Yatagawa, and S. Morishima, "Rsgan: face swapping and editing using face and hair representation in latent spaces," *arXiv preprint arXiv:1804.03447*, 2018.

[46] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, "Infogan: Interpretable representation learning by information maximizing generative adversarial nets," in *Advances in neural information processing systems*, pp. 2172–2180, 2016.

[47] E. L. Denton, S. Chintala, R. Fergus, *et al.*, "Deep generative image models using pyramid of adversarial networks," in *Advances in neural information processing systems*, pp. 1486–1494, 2015.

[48] X. Huang, Y. Li, O. Poursaeed, J. Hopcroft, and S. Belongie, "Stacked generative adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5077–5086, 2017.

[49] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," in *International Conference on Learning Representations*, 2018.

[50] tkarras, "tkarras/progressive growing of gans for improved quality, stability, and variation." `https://github.com/tkarras/progressive_growing_of_gans`, 2018. (Accessed on 01/14/2020).

[51] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2414–2423, 2016.

[52] NVlabs, "Nvlabs/stylegan: Stylegan - official tensorflow implementation." `https://github.com/NVlabs/stylegan`, 2019. (Accessed on 01/14/2020).

[53] D. Bau, J.-Y. Zhu, H. Strobelt, B. Zhou, J. B. Tenenbaum, W. T. Freeman, and A. Torralba, "Gan dissection: Visualizing and understanding generative adversarial networks," *arXiv preprint arXiv:1811.10597*, 2018.

[54] E. Collins, R. Bala, B. Price, and S. Susstrunk, "Editing in style: Uncovering the local semantics of gans," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5771–5780, 2020.

[55] L. Chai, J. Wulff, and P. Isola, "Using latent space regression to analyze and leverage compositionality in gans," in *International Conference on Learning Representations*, 2021.

[56] Goodfellow, "Ian goodfellow on twitter: "4.5 years of gan progress on face generation. https://t.co/kiqkuyulmc https://t.co/s4absu536b https://t.co/8di6k6bxvc https://t.co/uefhewds2m https://t.co/s6hkqz9glz... https://t.co/bqyv6zgftb"." `https://twitter.com/goodfellow_ian/status/1084973596236144640?lang=en`, 2019. (Accessed on 01/06/2020).

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

[58] D. Berthelot, T. Schumm, and L. Metz, "Began: Boundary equilibrium generative adversarial networks," *arXiv preprint arXiv:1703.10717*, 2017.

[59] V. Dumoulin, I. Belghazi, B. Poole, O. Mastropietro, A. Lamb, M. Arjovsky, and A. Courville, "Adversarially learned inference," *arXiv preprint arXiv:1606.00704*, 2016.

[60] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther, "Autoencoding beyond pixels using a learned similarity metric," *arXiv preprint arXiv:1512.09300*, 2015.

[61] A. Brock, T. Lim, J. M. Ritchie, and N. Weston, "Neural photo editing with introspective adversarial networks," *arXiv preprint arXiv:1609.07093*, 2016.

[62] M.-Y. Liu, T. Breuel, and J. Kautz, "Unsupervised image-to-image translation networks," in *Advances in Neural Information Processing Systems*, pp. 700–708, 2017.

[63] G. Perarnau, J. Van De Weijer, B. Raducanu, and J. M. Álvarez, "Invertible conditional gans for image editing," *arXiv preprint arXiv:1611.06355*, 2016.

[64] A. Heljakka, A. Solin, and J. Kannala, "Pioneer networks: Progressively growing generative autoencoder," in *Asian Conference on Computer Vision*, pp. 22–38, Springer, 2018.

[65] A. Heljakka, A. Solin, and J. Kannala, "Towards photographic image manipulation with balanced growing of generative autoencoders," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 3120–3129, 2020.

[66] S. Pidhorskyi, D. A. Adjeroh, and G. Doretto, "Adversarial latent autoencoders," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14104–14113, 2020.

[67] H. Huang, R. He, Z. Sun, T. Tan, *et al.*, "Introvae: Introspective variational autoencoders for photographic image synthesis," *Advances in neural information processing systems*, vol. 31, 2018.

[68] T. Daniel and A. Tamar, "Soft-introvae: Analyzing and improving the introspective variational autoencoder," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4391–4400, 2021.

[69] X. Huang and S. Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," in *Proceedings of the IEEE international conference on computer vision*, pp. 1501–1510, 2017.

[70] Z. Zhang, Y. Song, and H. Qi, "Age progression/regression by conditional adversarial autoencoder," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5810–5818, 2017.

[71] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[72] I. Korshunova, W. Shi, J. Dambre, and L. Theis, "Fast face-swap using convolutional neural networks," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3677–3685, 2017.

[73] W. Wu, Y. Zhang, C. Li, C. Qian, and C. Change Loy, "Reenactgan: Learning to reenact faces via boundary transfer," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 603–619, 2018.

[74] E. Sanchez and M. Valstar, "A recurrent cycle consistency loss for progressive face-to-face synthesis," in *2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020)*, pp. 53–60, IEEE, 2020.

[75] F. J. Pelletier, "Compositionality and concepts—a perspective from formal semantics and philosophy of language," in *Compositionality and concepts in linguistics and psychology*, pp. 31–94, Springer, Cham, 2017.

[76] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," in *Advances in neural information processing systems*, pp. 5767–5777, 2017.

[77] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4690–4699, 2019.