Advancing Controllability and Explainability in Generative 3D Face Models

Yajie Gu

PhD

University of York
Computer Science

September 2024

To my parents, Guohua Gu and Weiping Huang, and my family,

for their love, support and encouragement.

Yajie

Abstract

Three-dimensional face modelling, whether employing linear or non-linear approaches, involves mapping 3D face scans into a latent space for reconstructing 3D face shapes using this model. To enhance the interpretability of this mapped latent space for humans, a critical task emerges in computer vision with a focus on developing specific latent spaces for individual facial components rather than a single global latent space for the entire face. This thesis presents pipelines based on deep learning algorithms for the explainable and controllable non-linear modelling of 3D faces within latent spaces. Firstly, our method introduces a 3D face model that learns to map face identity and expression into two independent latent spaces, achieving face identity and expression disentanglement. This is particularly aimed at addressing limitations in scenarios lacking facial identity ground truths, which differs from other approaches. Secondly, beyond learning identity and expression latent spaces, our work further subdivides entire faces into multiple semantic regions, including the nose, eyes, mouth and others, and learns the separate latent variables for these regions through our novel framework. Additionally, we apply a Laplacian blending technique to the key facial feature swapping strategy, enhancing data augmentation and seamlessly reconstructing face shapes. Both methods are evaluated on public datasets and achieve state-of-the-art performance, demonstrating their effectiveness in reconstructing face shapes and disentangling latent variables for different facial features. The learnt latent variables are proven to be applicable to many applications, *e.g.* face recognition, face expression transfer and face editing. Moreover, to investigate the impact of different representations on the reconstructed face shapes, our two models employ different representations for 3D face shapes, one using explicit representations and the other employing implicit representations. Comprehensive comparative analyses are conducted to evaluate the effectiveness of our methods in 3D face modelling based on different representations and architectures.

Contents

List of Tables

List of Figures

Acknowledgments

I would like to express my deepest and sincerest gratitude to my supervisor, Prof. Nick Pears, for his continuous support throughout my research, his patience and his incredible knowledge. My journey towards completing this thesis and finding the right path would not have been possible without his invaluable guidance and encouragement. He was the first person to introduce me to the field of computer vision, providing countless suggestions and insights when I was almost unfamiliar with it. Thank you, Nick!

I would like to especially thank my Thesis Advisory Panel members, Dr. Patrik Huber and Prof. Will Smith, for their valuable feedback at every progression meeting during my PhD and enthusiastic greetings whenever we met.

I extend my genuine thanks to my research group, Vision, Graphics and Learning at Department of Computer Science for offering the necessary resources and environment for my research. In particular, I am grateful to my fellow PhD students within the same research group and to my colleagues in office CSE/115 for their companionship and the sparkling ideas they shared.

I would like to extend my sincere thanks to my MEng advisor, Prof. Jing Zhou, for her care and attention to my life. She provided me with unwavering help and kindness whenever I needed it, for which I am profoundly grateful.

My appreciation to my soulmate and travel pal Danyang Wang, for all the times we laughed together. I am deeply grateful to Haiyi Chen, Mengfan Zhou, Xueer Bai and Yaju Liu, for their cherished friendship. I would like to thank the friends I met in York, including Dr. Peiyun Hu, Dr. Yao Chen, Sue Anthony, Simon Anthony, my flatmates and neighbours, and friends from our film night group, for the memorable moments we shared.

Finally, I wish to express my heartfelt thanks and love to my parents, Guohua Gu and Weiping Huang, my beloved grandparents, Lumao Gu, Langjin Shuai, and Ruijin Huang, to my dear aunt and elder cousin, and to every member of my family. Without them, I would not have made it to today.

Author's declaration

I declare that this thesis is a presentation of original work and I am the sole author. The research described in this thesis, except where otherwise stated, is based on my own research carried out under the supervision of Prof. Nick Pears at the University of York and has not previously been presented for an award at this, or any other, University. All sources are acknowledged as References. Some chapters of the thesis have been published in following papers:

Chapter 3 includes:

- Gu, Y., Pears, N. and Sun, H., 2023, January. Adversarial 3D Face Disentanglement of Identity and Expression. In 2023 IEEE 17th International Conference on Automatic Face and Gesture Recognition (FG) (pp. 1-7). IEEE.

Chapter 4 includes:

- Gu, Y. and Pears, N. (2024). Parts-Based Implicit 3D Face Modeling. In Proceedings of the 19th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 2: VISAPP, ISBN 978-989-758-679-8, ISSN 2184-4321, pages 201-212.

Signed, Yajie Gu *YG*

— 1 —

Introduction

Three-dimensional shape reconstruction has become an increasingly active topic in the field of computer vision over the last two decades. In this thesis, we particularly focus on how to model 3D facial shapes using various types of representation. This has wide-ranging applications such as animation and gaming design, where parameters can be manipulated to create unique avatars, and medical imaging, such as plastic surgery, where there is a need for precise control over generated facial features. Through 3D face models, which are often referred to as 3D Morphable Models (3DMMs), raw face scans can be accurately reconstructed and understood.

Traditional 3D face statistical modelling employs linear space methods, which primarily involve the use of Principal Components Analysis (PCA) and linear combinations of shape deformations to construct 3D faces. Here, a large dataset of 3D human face scans is collected from individuals of diverse ages, ethnicities and genders. These face shapes are typically represented by 3D coordinates that share the same topology. Subsequently, techniques are utilised to analyse common facial features and variations, including geometric, global and local variations, expression variations and appearance and illumination variations [30]. As mentioned, PCA has been traditionally employed to analyse face shapes, identifying the principal components that capture the most significant shape variance across the dataset. This facilitates dimensionality reduction resulting in shapes that that are projected to a low-dimensional space. New face shapes can be generated by combining the mean shape with the a weighted sum of the retained principal components. However, these linear methods lack the flexibility to accurately capture variations of finer levels of facial information. Therefore, with the advancement of deep learning techniques, new algorithms that explore non-linear deep latent representations have been proposed.

These algorithms often employ an encoder-decoder architecture to achieve non-linear 3D face modelling. Among them, the AutoEncoder (AE), Variational AutoEncoder (VAE), Generative Adversarial Network (GAN) and diffusion models are the most commonly used deep architectures.

In the meantime, deep geometry learning in 3D computer vision has attracted more attention in recent years [109]. The ability of a model to represent 3D shapes accurately is crucial for the quality of reconstructed shapes. In this context, deep learning-based generative models have been explored for their potential in processing different types of representations, *i.e.*, explicit representations and implicit representations. Among explicit representations, voxels, point clouds, and meshes are the most commonly utilised in deep geometry learning. For example, due to the nature of voxels, graph convolutional networks (GCNs) are usually employed to process them effectively. PointNet [86] and PointNet++ [87] were proposed to process point clouds for 3D shapes, with these methods being widely applied in various pipelines.

Another important avenue in 3D shape reconstruction and modelling involves implicit representations, such as signed distance functions (SDFs) and occupancy probability functions. Deep generative models, leveraging either explicit or implicit representations, provide different advantages. For the former, the data are easier to obtain, and the models are more training time and memory efficient; the latter has the ability to synthesise shapes at flexible resolutions, which preserves the finer details of complex 3D objects.

Within this context, this thesis aims to employ deep learning-based architectures to learn latent spaces for non-linear 3D face shape models. Initially, we use explicit representations, specifically point clouds, to model 3D faces and decompose them into independent global identity and expression latent spaces, capturing their respective variations. Notably, this method addresses a common challenge in this field, the frequent absence of neutral expression ground truths, which is different from other approaches. However, while global explicit representations can model 3D faces, they may not always effectively capture finer details of facial regions. To overcome this limitation, we introduce a novel method that employs implicit representations, *i.e.*, SDFs, to learn segmented latent representations of specific facial areas. To our knowledge, the approach we present, which is based on implicit representations, has not been previously applied. To further evaluate the efficacy of these approaches, we conduct a comparative analysis of models utilising different representations and

architectures, focussing on their capability to reconstruct unseen inputs, generate novel face shapes and explore applications.

In the following sections, we outline our research objectives, which include 3D face disentanglement of identity and expression using explicit points-based shape representation, and parts-based implicit 3D face modelling. Additionally, we conduct a comprehensive comparative analysis to evaluate the effectiveness of these two methods in 3D face modelling. Each of our following research objectives is divided into two main subsections: the background and motivation behind our work, and the specific contributions of our study.

1.1 EXPLICIT 3D FACE DISENTANGLEMENT

Our first objective focuses on disentangling the 3D shapes associated with identity and facial expression from an input shape that is represented as an unstructured point cloud. All 3D shape representation is in an explicit form.

1.1.1 Background and Motivation

Point clouds, as a form of explicit representations of 3D shapes, have seen widespread use in recent years due to their ease of acquisition and relative simplicity compared to other explicit representations. Moreover, architectures including PointNet [86], PointNet++ [87] and Point Cloud Transformer [44] were proposed for processing 3D shapes using point clouds representation, providing the groundwork for the architectural design of our network.

Since the introduction of linear 3DMMs in 1999 [5], which presented a PCA-based 3D face model to represent variations in identity and expressions, the task of 3D facial shape modelling for identity and expression components has been a popular research topic. Independent identity and expression representations are utilised in many applications, including facial identity and expression interpolation, facial expression transfer and facial neutralisation for recognition purpose. These applications heavily rely on the effective disentanglement of identity and expression.

Concurrently, in the light of developments in deep learning and the transition towards non-linear modelling of 3D faces, recent studies have proposed innovative methods that use a single encoder and two decoders. This common architecture encodes the entire face and outputs latent embeddings separately for identity and

expression from the encoder. Following this, two decoders are employed to reproduce the facial identity and expression based on their respective embeddings. However, a notable challenge of this architecture is its dependency on the availability of a corresponding facial identity for the input face shape, which is used as a reference for the identity decoder. The distance between the predicted face identity from the identity decoder and the facial identity ground truth can be employed within a loss function in the supervised learning process for face identity reconstruction. This loss function is critical not only for ensuring the accuracy of identity shape reconstruction but also for keeping the learnt identity representations as separate from the expressive ones as possible.

For example, a framework using two VAE models that share the same encoder but employ two different decoders: one for facial identity shapes and the other for the full face was proposed in [94]. As previously mentioned, an identity reconstruction loss function, comparing the generated output to the original input, is implemented. The strategy of utilising the identity loss function and ground truths was also adopted in other research work, such as [46, 116, 48].

Although existing studies demonstrate strong performance in 3D face reconstruction and disentanglement through the use of identity and expression latent representations, a significant challenge arises in real-world applications such as facial expression neutralisation: expressive faces are often available without their corresponding neutral expression ground truths, *i.e.*, facial identity.

This leads us to propose research question 1 (RQ1):

- *RQ1: In the absence of neutral expression ground truths, how can we effectively achieve the disentanglement of facial identity and expression, and accurately reconstruct the neutral facial identity and the full expressive face using the learnt decoupled latent representations?*

1.1.2 CONTRIBUTIONS OF OUR WORK

To address RQ1, we propose a novel approach that integrates VAE and GAN frameworks, specifically for processing 3D face shapes represented as point clouds. This architecture consists of an encoder and two decoders, adhering to a structure similar to those previously discussed. Distinctively, we incorporate an identity discriminator, drawing inspiration from the innovative work presented in [40, 41].

Their studies identified the 'point of apathy' within the expression space as the state in which facial muscles are most relaxed.

Building upon this finding, we hypothesise that the origin of the expression manifold, representing the connection between faces sharing the same identity, is the apathy expression. To leverage this concept, we introduce an identity discriminator into the adversarial learning process of our architecture. This discriminator is designed to encourage the preservation of shared information within face shape pairs that share the same identity, thereby ensuring the model effectively retains identity-specific features. Consequently, in scenarios where neutral expression ground truths are unknown, our strategy utilises the invariance of identities from the same individuals, adopting the 'apathy expression' as the pivotal point in the expression space for model training. Meanwhile, we also consider scenarios where facial identity ground truths *are* present in the dataset, ensuring our settings align with other architectures for easier comparison under both scenarios.

Our hypothesis and innovative design have enabled our network to demonstrate superior performance not only in disentanglement and reconstruction tasks but also in facilitating practical exploration into diverse scenarios of face identity and expression modelling. Empirical validation across multiple datasets has confirmed the robustness of our method and its ability to outperform existing models. Furthermore, effective disentanglement of facial identity and expression has inspired our further research, especially in the representation of finer segmented facial regions based on facial neutral expression, *i.e.*, facial identity shapes.

In summary, at the time of its initial proposal, to the best of our knowledge, our proposed method is the first to consider the challenge of unknown facial identity ground truth, introducing an end-to-end architecture designed to address RQ1.

1.2 PARTS-BASED IMPLICIT 3D FACE MODELLING

Our second objective also aims to disentangle identity and expression, with the additional aim of disentangling latent codes for facial parts (*e.g.* eyes, nose, mouth, etc), whilst exploring the benefits of employing implicit shape representation.

1.2.1 Background and Motivation

In 2019, the introduction of deep implicit representations marked a milestone in the modelling of 3D shapes, including SDFs [79] and occupancy functions [72, 18, 66] that model 3D shapes as continuous functions over the 3D space. Such representations have become highly appealing for many methods, as they can better represent complex topologies of 3D shapes and demonstrate robustness against noisy and incomplete data. Moreover, the inherent continuity of these representations ensures their compatibility with deep generative models and 3D shape reconstruction is achieved by gradient-based optimisation.

Following the discussion and achievement of 3D face modelling to disentangle identity and expression, our research interests expanded to include modelling of the 3D face into smaller facial regions. This aims to improve both the interpretability and controllability of the 3D face model. Notably, methods like those proposed in [34, 2] employ a VAE to learn decoupled latent variables for different facial parts.

The study conducted in [2] employs seven different encoders, each with a varying number of dimensions, to separately learn the latent codes for corresponding predefined facial parts. These latent codes are then concatenated into a unified vector as the input for a single decoder to reconstruct full faces. Furthermore, the work introduces a specialised loss function designed to reinforce latent space disentanglement and ensure that each part of the latent vector affects only the assigned part of the face.

In another approach presented in [34], a network utilising a single encoder and decoder is employed to learn concatenated latent representations for the entire face. Additionally, this method divides the latent code into fixed dimensions allocated for specific facial parts and introduces a loss function to enforce part-wise similarities within the latent codes and differences across the remaining parts.

Compared to existing work, our research uniquely concentrates on learning independent latent vectors for individual facial parts, as opposed to concatenating them into a single latent vector for the entire face, including its decoupled components within the vector. Furthermore, we place particular emphasis on the use of implicit representations for 3D face shapes, leveraging their advantages over explicit representations, as introduced previously.

Aligned with our second research objective, we propose the second research question:

- *RQ2: How can we effectively learn a 3D face model with independent facial part latent representation and control, while still dealing with facial expression, and additionally gaining the benefit of implicit representations for arbitrary resolution 3D face shape reconstruction?*

1.2.2 Contributions of Our Work

In our second study, we introduce a sequential network designed for facial parts deformation, inspired by the work presented in [117]. This method is aimed at answering RQ2 and filling the research gap in 3D parts-based facial modelling using implicit representations, which learns independent latent representations for multiple facial regions in an end-to-end manner.

In order to learn separate latent spaces for specific facial components, we predefined three key semantic facial regions: the nose, eyes and mouth. For the rest of the face, we consider it as a single part, referred to as the 'remainder'. While it is possible to further subdivide the remaining parts, such as the forehead and cheeks, we opted to group them together since these regions are less semantically meaningful and their deformations are less obvious in qualitative observation compared to the other predefined areas.

Unlike existing VAE-based architectures that learn a unified, disentangled latent representation for the face shape, our method employs a deformation network comprising a set of neural networks. This end-to-end network is designed to transform an input face shape to align with a predefined template face by sequentially executing deformations tailored to designated facial regions. In detail, we develop five specialised neural nets for each facial region: NoseNet for the nose, EyesNet for the eyes, MouthNet for the mouth, RemNet for the remaining parts of the face; and TemplateNet (also known as SDFNet), which serves as a reference framework to output the SDFs for the overall facial template. Each network is designed to learn latent embeddings specific to its corresponding facial part, enabling targeted and precise deformation processes.

In the architecture, we introduce an additional component, *i.e.*, ExpNet, specifically designed for learning separate expression latent variables. This innovation significantly contributes to the disentanglement of facial expression from identity, further improving the robustness of our method. Although the disentanglement of identity and expression in 3D facial modelling has been a focus of our previous re-

search objective, the key difference is that ExpNet is based on implicit representations, *i.e.*, SDFs of 3D faces, rather than point clouds.

To enhance the effectiveness of our representations and improve their generative capabilities, our method adopts an innovative data augmentation strategy: swapping facial features, *i.e.*, the nose, eyes, and mouth, among different instances in the dataset. Inspired by [34], this technique involves exchanging these features with those from other randomly selected face parts within our training dataset. Since we use affine transformation that optimally (least squares) matches the facial feature peripheral vertices into the graft site vertices of the face, the preprocessed face is different from the original face.

However, a challenge arising from our data augmentation strategy is the incoherence between the swapped features and the original face, resulting in a visible seam on the face. To overcome this issue, we explore the use of a Laplacian blending technique during the feature swapping process. Finally, the swapped faces demonstrate that employing Laplacian blending effectively eliminates the seams at the junctions of swapped features, significantly improving the quality of the face reconstruction.

This end-to-end deformation network, using implicit representations, not only disentangles expression and identity latent representations but also learns independent latent embeddings for specific facial regions. There are many applications based on these learnt latent embeddings, including face part editing and facial region interpolation.

In summary, the contributions of this work are as follows:

- A novel strategy that integrates facial feature swapping with Laplacian blending to enhance data augmentation, ensuring seamless generation of face or head shapes.

- The development of an end-to-end deformation network that effectively disentangles latent representations of expression and facial parts-based identity on both 3D face and head shapes, using implicit representations.

- Comprehensive evaluation on publicly available datasets that demonstrates strong performance in face reconstruction and disentanglement of 3D face or head shapes.

1.3 COMPARISON OF EXPLICIT AND IMPLICT APPROACHES

As outlined in the research objectives 1 and 2 above, we aim to model 3D face shapes using two different representation forms, *i.e.*, point clouds (explicit shape representation) and SDFs (implicit shape representation), each with its own advantages. The networks developed for these objectives are designed to facilitate a thorough learning process, achieving decoupled latent representations for facial expressions, identity and individual facial parts.

Given that both of our proposed methods successfully disentangle expression and identity latent vectors, our third research objective is to conduct a comparative analysis between these two methods. This comparison evaluates their performance from five perspectives: the quality of 3D face reconstruction, the effectiveness in disentanglement of facial features (specifically identity and expression), the ability to generalise to new faces, the practical applications based on their learnt latent codes and computational resource requirements.

For the first three perspectives, our comparisons focus only on the learnt latent representations for facial identity and expressions using the same dataset, which is easier for comparison. Moreover, in the visualisation of various applications of our disentangled latent codes, we also consider the additional capability of our implicit deep learning to disentangle facial parts. This further explores the effectiveness of our overall 3D face modelling architectures.

In this thesis, we not only present comprehensive comparisons between our methods and other state-of-the-art face modelling methods proposed in recent years but also conduct an intra-comparison of our own methods, with a particular focus on 3D face modelling.

1.4 STRUCTURE OF THE THESIS

The following chapter presents a comprehensive review of existing studies relevant to the topics explored in our study. This includes a thorough review focussing on 3DMMs, with their fundamental components, *i.e.*, representations of 3D shapes, and the deep generative models that support the construction of 3DMMs, and studies most directly connected to our research.

Chapter 3 focuses the research on 3D face disentanglement of facial identity and expressions, across a variety of scenarios. Through comprehensive validations on multiple datasets, we demonstrate the robustness and superior performance of our proposed architecture. Moreover, we explore various applications based on this method.

Chapter 4 provides a detailed explanation of our research topic on implicit 3D face modelling on identity, expression and facial regions. In this chapter, we present an in-depth description of our architecture, the data augmentation strategy and the improvements implemented for 3D face reconstruction. Additionally, we provide comprehensive evaluations on various datasets, comparing our results against state-of-the-art approaches. This chapter further discuss the practical applications of our research.

Chapter 5 describes a thorough comparative analysis of our proposed methods for 3D face modelling. We not only present new evaluation results that highlight the strengths and limitations of each method in this chapter, but also introduce different applications based on our studies.

A final chapter concludes the thesis by summarising the key contributions in 3D face modelling, critically discussing the potential limitations of our methods and exploring future work.

— 2 —

Literature Review

In this chapter, a comprehensive review of the literature relevant to the proposed research topic is presented. It provides not only the broader context within which our research is situated, but also the research gaps this work is intended to fill. The literature review is structured as illustrated in Figure 2.1 and the various sections will examine the following essential components of our work:

i. **Representations of 3D shapes**, establishing the fundamental knowledge necessary for understanding models of 3D shapes. This includes both explicit and implicit representational forms, which will be described in Section 2.1.

ii. **3D Morphable Models (3DMMs)**, as the core of our literature review and introduced in Section 2.2, 3DMMs are generative models originally designed by Blanz and Vetter [5] for linear representations of facial shape (identity and expression) and texture.

iii. **Deep generative models**, focussing on Variational Autoencoders (VAEs) and Generative Adversarial Networks (GANs) in Section 2.3, represent the deep network technical foundation commonly adopted for constructing 3DMMs.

iv. **Literature that is most closely related to the thesis contributions**, including learning of 3D face latent representations of identity and expression, and separate facial regions, which aim to achieve disentanglement, will be discussed in Section 2.4.

Finally, we close the literature review in Section 2.5, by analysing key findings and highlighting their relevance to our studies.

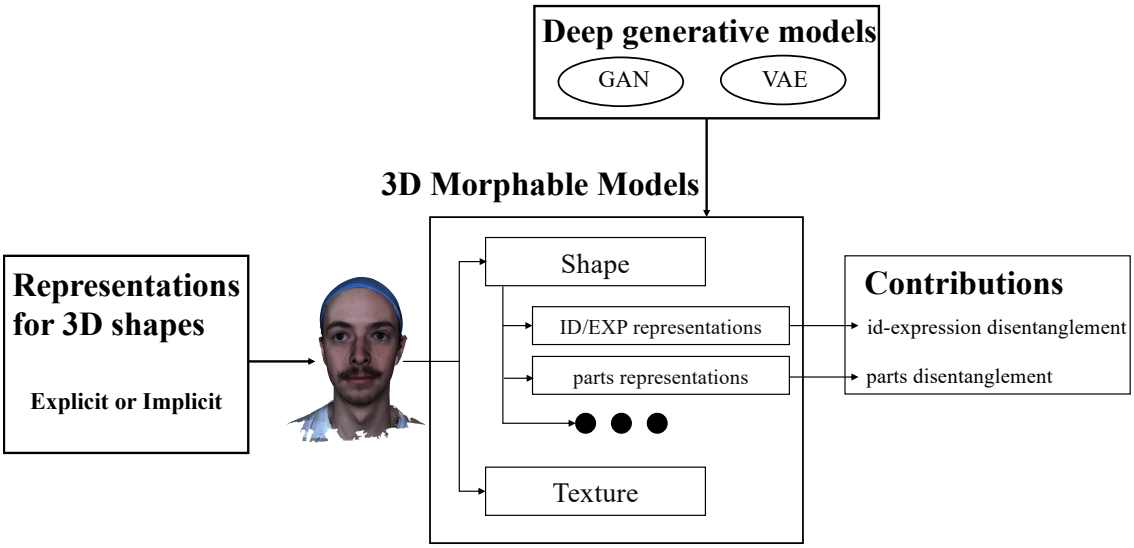

Figure 2.1: Conceptual framework for the relationship between the various literature review sections.

2.1 3D SHAPE REPRESENTATIONS

This work focuses on the analysis of 3D face shapes, aiming to develop sophisticated deep 3DMMs, the details of which are extensively discussed in Section 2.2. In this section, we introduce representations of 3D shapes, a fundamental component for understanding 3D face modelling.

2.1.1 EXPLICIT REPRESENTATIONS OF 3D SHAPES

3D shapes can be described using various explicit representations including voxels, point clouds and meshes. Figure 2.2 shows the Stanford bunny 3D shape using these representations. Typical explicit representations are:

- **Depth Maps**: 2D images that encode depth information, enabling 3D shapes to be represented by RGB-D (color and depth) images captured from different viewpoints and projected into a 2D plane [108].
- **Voxels**: analogous to pixels in 2D images, represent 3D shapes through a regular grid structure in 3D space and are commonly used in 3D reconstruction but may necessitate significant resource consumption as resolutions increase.

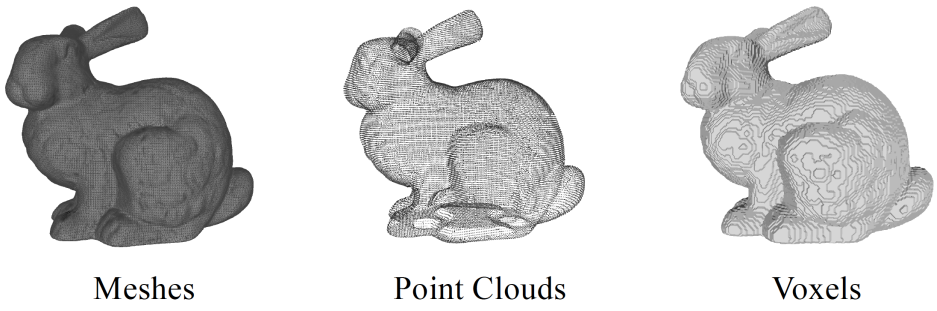

Figure 2.2: Three types of explicit representations for the Stanford Bunny object: meshes, point clouds and voxels.

- **Meshes**: composed of vertices, edges and faces, offer lower memory and computational costs compared to voxels. Mesh vertices fully contain all local surface information, including vertex connectivity, and the surface normal at any point on the shape surface can often be approximated by a nearest neighbor or a local linear combination of vertex normals. Their inherent connectivity facilitates the application of graph-based Convolutional Neural Networks (CNNs), leading to high-quality reconstructions of 3D objects.
- **Point Clouds** consist of a set of sampled 3D coordinates that represent the surfaces of 3D shapes. They are effective in capturing fine details, relatively easy to obtain and also convenient to process.

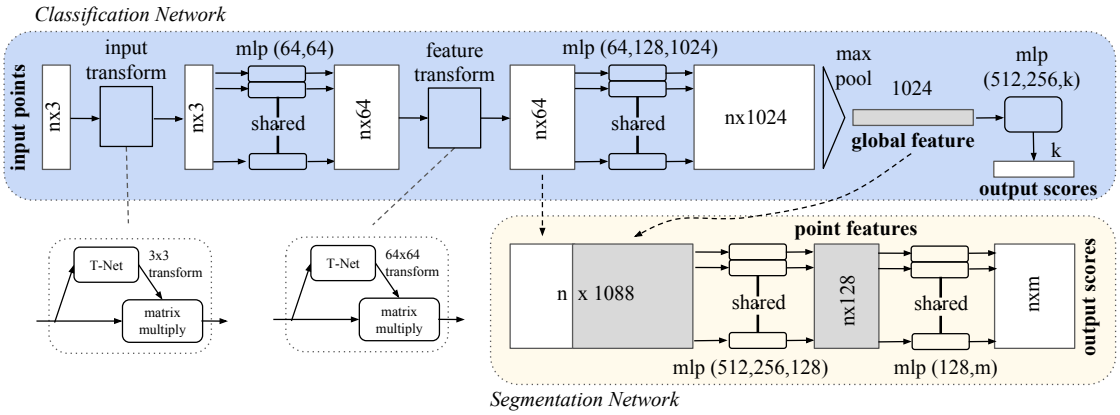

Figure 2.3: PointNet architecture (figure cited from [86]).

The introduction of PointNet [86] and its extension, PointNet++ [87], marked a significant milestone in the processing and popularisation of point clouds. We

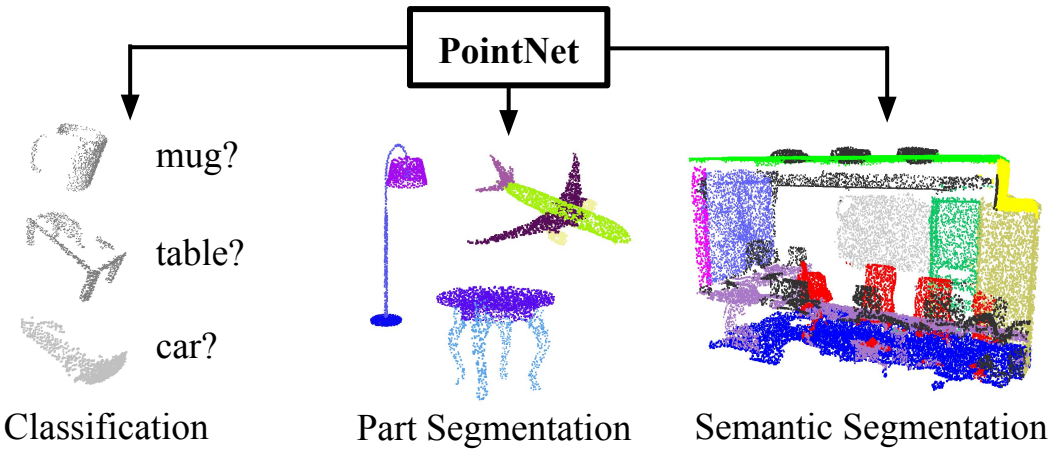

Figure 2.4: PointNet applications (figure cited from [86]).

will provide detailed introductions to these models since we employed PointNet as the base network for training our explicit face model. PointNet [86], a unified deep network architecture, directly uses point clouds due to their permutation invariance, for many applications, including part segmentation, object classification and scene semantic segmentation, as detailed in Figures 2.3 and 2.4. Given the traditional CNN's requirement for a more regular grid structure and the irregular format of point clouds that are less complex than meshes, a straightforward and widely adopted deep network was proposed that processes unordered point clouds directly as both input and output. As depicted in Figure 2.3, for the task of object classification, the classification network takes n sampled points represented as 3D coordinates (x, y, z) through input and feature transformations, and employs max pooling to aggregate the features of all sampled points into global features, yielding k predicted scores for the corresponding classes. For a more complex segmentation task, an additional network resolves it by leveraging not only the global features but also the local features of each point. These concatenated global and local features are then used to predict scores for each point.

To address the limitations of PointNet and improve its ability to recognise fine details, capture local structures and improve generalisability for complex scene segmentation tasks, Qi *et al.* proposed an enhanced version named PointNet++ [87], a hierarchical feature learning network to combine features at multiple scales. This approach significantly improves upon the original model's performance in scene segmentation, as illustrated in Figure 2.5, by addressing several drawbacks of

PointNet:

1. PointNet employs only a single max pooling layer for all sampled points, leading to information loss. In contrast, PointNet++ introduces a hierarchy of abstraction layers that progressively process points to abstract larger local regions across different scales.
2. Unlike PointNet, which applies point-wise Multilayer Perceptrons (MLPs) to extract features for each point, PointNet++ introduces two key layers, *i.e.*, sampling and grouping, which enables the selection of certain points from the input as centroids for local regions and the construction of these local regions by identifying neighbours of these selected points.
3. In the segmentation network of PointNet, global features are directly concatenated with local features. PointNet++, however, subsamples points and interpolates features on these points before concatenating them with skip-linked local features from the corresponding scale to achieve more distinctive segmentations.

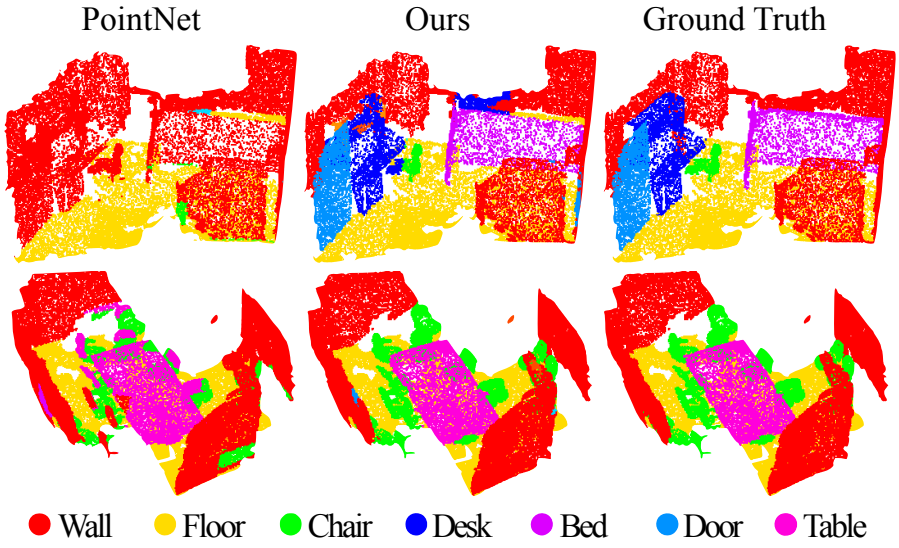

Figure 2.5: The comparative performance between PointNet and PointNet++ in room scene segmentation, 'Ours' refers to PointNet++ method (figure cited from [87]).

In addition to PointNet and PointNet++, which are based on point cloud representations, several other methods utilise different explicit representations of 3D shapes. For example, Yan *et al.* employed voxel representation in an innovative way

to train a single view AE model for 3D volumetric reconstruction in an unsupervised manner, combining an image encoder, volume decoder and perspective transformer [111]. Chen *et al.* proposed a novel deep GAN for 3D shape detailisation, which refines low-resolution coarse voxel shapes through voxel upsampling to achieve higher-resolution shapes with geometric details [17]. Zhou *et al.* introduced a Point-Voxel diffusion network, employing denoising diffusion models with hybrid point-voxel representations of 3D shapes, pioneering a diffusion process that effectively captures the underlying structure and patterns by mutually transforming point clouds and voxel grids [119]. For meshes, MeshGAN [19] built non-linear 3DMMs using mesh representations. Liu *et al.* chose meshes for their 3D surface representations, enabling easy optimisation based on many graphics techniques and arbitrary shape manipulation for applications like relighting and simulation [69]. They leverage the graph structure of 3D meshes and represent these meshes with deformable tetrahedral grids, marking the first application of a diffusion model for the unconditional generation of high-quality 3D meshes. Tarasiou *et al.* described the Locally Adaptive Morphable Model (LAMM) [101], which encodes a source mesh into a latent code and applies additional displacements at specific controlled points within each facial region to generate semantically partially changed new faces.

2.1.2 Implicit Representations of 3D Shapes

As one of the many 3D shape representations, deep implicit functions are gaining increasing attention. Unlike traditional explicit representations (*e.g.* point clouds, meshes and voxels), deep implicit functions describe shapes within a continuous volumetric field, defining the spatial relationship between points and surfaces. Such representations are capable of representing shapes with flexible topologies and continuously increasing resolution under reasonable memory consumption.

In 2019, Park *et al.* introduced a learnt continuous **Signed Distance Function (SDF)** [79] that facilitates high-quality representation, interpolation and completion from partial or noisy 3D data. SDF represents shape surfaces within a continuous field, where the sign determines whether points are inside (negative) or outside (positive) the shape, and the magnitude indicates the distance to the surface boundary with the boundary itself as the zero level-set of the SDF. Concurrently, the introduction of occupancy probability has become another option that can be used to achieve flexible resolutions and is more robust to complicated topologies [72,

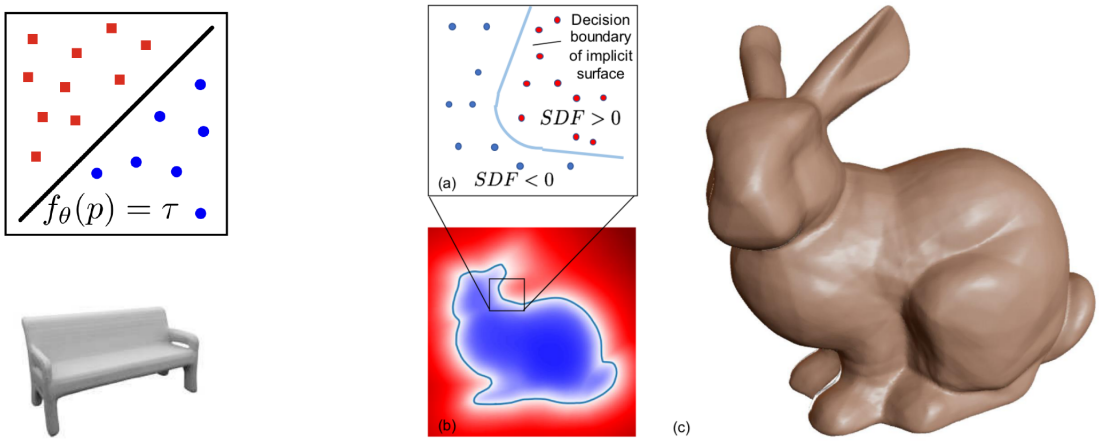

Figure 2.6: Two types of implicit representations: left: occupancy functions (the chair from [72]); right: SDFs (the Stanford bunny from [79]).

18, 66]. Mescheder *et al.* implicitly represented the 3D surface as a continuous decision boundary, approximating a 3D function that assigns every possible point an occupancy probability between 0 and 1 [72]. Both SDF and occupancy function representations are depicted in Figure 2.6.

Since the early developments in deep implicit representations, exemplified by DeepSDF [79] that leveraged the auto-decoder model, there have been improvements in coarse reconstruction of shapes such as chairs and airplanes, yet challenges remain at finer levels of detail. Subsequent advancements have been presented to enhance the 3D shape reconstruction performance in these implicit representations, especially in detailisation, as highlighted by [29, 98, 63]. Duan *et al.*, inspired by the human learning process that begins with simpler tasks and moves to more complicated ones, proposed a shape curriculum learning approach, organising tasks in an ascending order of difficulty based on surface accuracy [29]. It outperformed DeepSDF on reconstruction under the same architecture as well as the same set of training data and number of epochs. Unlike DeepSDF that persistently focuses on the same objective, such an approach started from the learning of smooth shapes with a high tolerance parameter. Such a method allows errors smaller than parameters during SDFs estimation, enabling control over surface accuracy through the adjustment of the tolerance parameter. Lipman designed a new loss function for training implicit neural representation directly from input raw geometry, where the learnt density function converges to an accurate occupancy function, and its logarithmic transformation

converges to a distance function [63]. Takikawa *et al.* combined a small surface extraction neural network with a sparse-octree data structure, achieving state-of-the-art geometry reconstruction quality and enabling real-time rendering in 2021 [98].

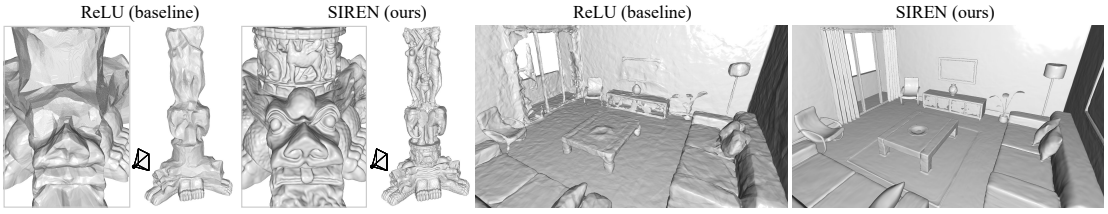

Figure 2.7: Comparison of shape representation between SIRENs and ReLU implicit representations (figure cited from [91]).

Sitzmann *et al.* [91] proposed a general pipeline, Sinusoidal Representation Networks (SIRENs), that is capable of being applied on multiple scenarios like representing images, wavefields, video and sound. SIRENs leverage periodic activation functions with MLPs for implicit neural representations, enabling the robust fitting of complicated 3D shapes and their derivatives, as well as addressing challenging boundary value problems, outperforming traditional MLP networks with ReLU activation in SDFs-based reconstruction, as illustrated in Figure 2.7. Such an approach significantly improves the capacity of neural networks that employ implicit functions to reconstruct fine details of objects and the complexity of scenes. Moreover, one of the most challenging issues for a network with sine activation functions is its initialisation, which impacts the network's final performance and convergence speed during training. To mitigate this issue, Sitzmann *et al.* presented a novel initialisation scheme for training SIRENs that preserves a normal distribution with a stardard deviation of 1 for each input to the sine activation function through the network, ensuring the output derived from the initialisation remains independent of the layers of networks.

In our work on 3D face modelling, we utilised SIRENs and their initialisation scheme to learn priors for implicit representations, *i.e.*, SDFs, effectively fitting differentiable SDFs to parameterise 3D human faces, especially focusing on details of facial regions. To generate unseen new faces in our research, the learnt latent codes are essential. Adopting the idea from Sitzmann *et al.* , where each SIREN is defined by its parameters, we modelled latent codes onto these parameters in order

to learn a latent space for implicit representations, different from previous methods that typically learn latent codes for general 3D objects or scenes.

On the other hand, deformation networks have been specifically designed for use with implicit representations. The exploration of deformed implicit networks for objects with complex geometric variations was investigated in [118, 27, 95, 47]. Deng *et al.* focussed on leveraging a template implicit field across object categories, representing 3D shapes through their combination with the template, 3D deformations and corrections [27]. Zheng *et al.* learnt a plausible template and employed Long Short-Term Memory (LSTM) as a spatial warping module to achieve point-wise transformations in an unsupervised manner [118]. Furthermore, Sundararaman *et al.* [95] and Jung *et al.* [47] developed auto-decoder-based networks to reconstruct a 3D deformation field between a fixed template and a target shape.

Concurrently, the application of implicit representations in 3D face, body and hand modelling has emerged, such as [96, 113, 89, 21]. For instance, Yenamandra *et al.* introduced i3DMM [113], the pioneering deep implicit 3D morpable model of full heads, and created a new dataset consisting of 64 subjects, each with different expressions and hairstyles. This method established a 3D model through the use of a reference network that encodes a single reference shape, allowing all individual shapes to be deformed towards this reference without learning a latent code for the reference shape itself. Additionally, a shape deformation network was developed to learn the latent codes of the displacement aligning the corresponding shape with the reference. Furthermore, a colour network was proposed to accurately capture the colour for the given shape and hairstyles. Giebenhain *et al.* proposed a novel 3DMM for complete human heads [38], innovatively embedding the human identity within a canonical signed distance field and the expressions within a neural deformation field. Additionally, they released a newly captured dataset comprising over 5200 head scans, including 29% female, from 255 different identities. A Pixel-aligned Implicit Function (PIFu) [89], introduced by Saito *et al.*, enables textured surface inference of clothed 3D humans from a single or multiple input images. PIFu aligns individual local features at the pixel level to the global context of the entire object. Chibane *et al.* utilised occupancy functions in their approach [20]. Instead of using a single latent code to encode a 3D shape, they constructed a rich encoder for the input data by subsequently convolving it with learnt 3D convolutions to create multi-scale deep features. For the decoder, they extracted deep features from the grid at continuous point locations to determine their occupancy probability. This approach successfully

reconstructs articulated structures, *i.e.*, human body, while preserving input details.

2.1.3 Analysis

The use of both explicit and implicit representations is widespread in 3D modelling, each bringing its unique strengths, as introduced in Sections 2.1.1 and 2.1.2. While significant advancements have been made and many innovative ideas have been proposed, challenges remain in learning latent spaces for specific facial regions and generalising to unseen facial parts, especially when using implicit representations. Although explicit representations, such as point clouds and meshes, have shown progress in this area, they come with limitations, particularly in achieving flexible resolutions, an inherent advantage of its implicit counterpart.

On the other hand, the use of unsupervised networks for modelling 3D facial identity and expression has attracted considerable research interest for both explicit and implicit representations. The lack of corresponding identity and expression ground truths in real-world scenarios, however, adds complexity to this issue.

2.2 3D MORPHABLE MODELS (3DMMS)

In this section, we focus on the technical core of our research, *i.e.*, 3DMMs. We begin with an introduction to the foundational 3DMM [5], which was first proposed in the late 1990s, and outline its subsequent development.

Much work focuses on analysing 2D and 3D images of the human face in terms of their physically-meaningful components, *i.e.*, subject identity, facial expression, surface reflectance, illumination and camera parameters. The introduction of 3DMM is an early notable milestone, which was proposed by Blanz and Vetter in 1999 and specifically designed to model textured 3D faces. This innovative model, represented by a multi-dimensional 3D morphing function, accurately maps shapes and textures with dense correspondence into a vector space, rather than relying on facial feature points alignment, and generates new faces through linear combinations of a base 'prototype' face, as mathematically represented in Equation (2.1) for shape, and Equation (2.2) for texture.

$$\mathbf{S} = \overline{\mathbf{S}} + \sum_{i=1}^{m} \alpha_i \mathbf{s}_i, \tag{2.1}$$

$$\mathbf{T} = \overline{\mathbf{T}} + \sum_{i=1}^{m} \beta_i \mathbf{t}_i, \tag{2.2}$$

where $\mathbf{S} \in \mathbb{R}^{3n}$ (x_1, y_1, z_1, ..., x_n, y_n, z_n) represents the facial shape vector and $\mathbf{T} \in \mathbb{R}^{3n}$ (r_1, g_1, b_1, ..., r_n, g_n, b_n) represents the facial texture vector, consisting of n points. $\overline{\mathbf{S}}$ and $\overline{\mathbf{T}}$ denote the mean shape and texture, respectively. α and β are the parameters for the descending order eigenvectors $\mathbf{s}$ and $\mathbf{t}$, respectively. Additionally, m typically represents the number of principal components in the model. This model enables the automatic alignment of the 3D face morphable model with 2D images and facilitates the registration of new 3D face shapes. The model was built from the first 100 shape principal components and the first 100 texture principal components, both derived from 200 exemplar faces, where new characters can be generated from modifying the facial attributes by varying the model coefficients.

2.2.1 Developments of 3DMMs

Further improvements by Blanz *et al.* [4] refined the model by applying PCA not only to the shape space for geometric representation but also to the incorporation of expressive principal components, which is a common practice. From then on, work of others concentrated on identity and expressions analysis. Bouaziz *et al.* proposed a dynamic 3D expression model that combined an identity PCA model, a dynamic expression template, and a parameterised deformation model in a low-dimension space [9]. This model was able to transform the neutral shape to generate user-specific blendshapes without requiring manual assistance. A statistical and multilinear model [10] was employed to analyse facial identity and expression, exploring their variations. This model decomposed a high-dimensional global shape space into many localised and decorrelated low-dimensional shape spaces, enabling it to avoid overfitting during training, learn local fine details, and fit 3D faces from noisy and occluded shapes from various sources.

In subsequent years, 3D face models were developed that use more sophisticated shape morphing techniques (exemplified by the widely used Basel Face Model (BFM) [81]) or leverage a larger body of 3D training samples [8, 6, 7]. The BFM was constructed based on 200 face scans from the training set, consisting of 100 male and 100 female faces. These face scans were captured using a high-quality scanning device that was intended to improve the precision of shape and texture, and correspondence between scans was established through the Optimal Step Non-rigid

Iterative Closest Points (NICP). To facilitate the synthesis of new faces, the shape and albedo PCA models are learnt, enabling detailed and accurate facial modelling. Booth *et al.* developed a statistical feature-based texture model from "in-the-wild" facial images, which fully corresponded to a statistical shape in both identity and expression variations [6]. Furthermore, Booth *et al.* [8, 7] were the first to propose a large-scale 3DMM in a neutral expression, constructed automatically from 9,663 different face identities, and tailored for specific gender, age and ethnicity groups. Prior to their work, 3DMMs were constructed using small datasets with manual preprocessing work for face meshes. Constructing a 3DMM typically involves two critical stages: establishing dense correspondence between face meshes to ensure a shared topology, and performing similarity alignment and statistical modelling. In their methodology, Booth *et al.* automatically localised the landmarks and then employed NICP based on these automatic landmarks to align all meshes to a template. Finally, they constructed a global PCA from the face meshes. The automatic dense correspondence proposed in [7] not only uses NICP but also compares with other two 'UV' based interpolation techniques, UVTPS and UV-Optical Flow (UV-OF). A 2D 'UV' space was defined for each face mesh and associated with its corresponding 3D surface through a bijective mapping. Thus, the correspondence between two 'UV' images represented the corresponding between two mapped 3D meshes. The comparison proved that, compared with UVTPS and UV-OF, NICP is a much better candidate for building an anatomically accurate and relevant statistical model.

Recent developments in 3D face modelling have expanded the scope of models to cover the full cranium as well as the face [25, 24]. The model proposed by Dai *et al.* was the first public shape-and-texture craniofacial 3DMM of the full head, which could be used in the clinical applications for several types of surgical intervention. This innovative work not only built a global craniofacial 3DMM but also developed demographic-specific sub-population 3DMMs (male, female, aged less than 15, aged between 15 to 30, aged between 31 to 50, aged over 50, using data from 1,212 subjects, 606 female and 606 male) within the Headspace dataset. It also introduced a high-quality texture map for statistical texture modelling. Ploumpis *et al.* extended this innovation by presenting a comprehensive fused 3DMM of head shapes, including the face, cranium, ears, eyes, teeth and tongue [84]. To achieve this nearly complete model, they blended the facial detail from an existing face model, *i.e.*, the Large-Scale Face Model (LSFM) [8], with the existing full head model, *i.e.*, the Liverpool-York Head Model (LYHM) [25] and they called this the Universal

Head Model (UHM). They proposed an interesting idea to find a universal covariance matrix that combines the covariance matrix of the highly detailed facial attributes from LSFM and the covariance matrix of the head distribution from LYHM, using a Gaussian processing framework.

FLAME (Faces Learned with an Articulated Model and Expressions) [61], as proposed by Li *et al.*, is a highly influential 3D morphable model in the domain, which is a statistical expressive head model that explicitly separates the representations of identity, expression and pose. Incorporating neck, jaw and eyeballs into the identity latent space, FLAME is based on the analysis of 3,800 head shapes for identity, pose parameters from 8,000 registered heads, and expressions from 21,000 registered frames of 3D motion sequences. DECA (Detailed Expression Capture and Animation) [31] regresses a parameterised face model with geometric details and generic expression parameters. This process enables reconstruction of a detailed 3D head model with detailed face geometry from a single face image.

While many of these models operate within a linear subspace following a Gaussian distribution, recent innovations have seen the introduction of models that incorporate articulated components with non-linear transformation [106, 19], significantly augmenting the representation power of 3DMMs. Tran and Liu proposed a non-linear 3DMM that utilised a CNN encoder to estimate the parameters for shape, texture and projection, as well as two decoders for mapping these parameters back to their corresponding 3D shapes and textures [106]. Remarkably, this model learnt the non-linear 3DMM directly from unconstrained 2D faces images without collecting 3D scans. Cheng *et al.* employed ChebNet [26] as the core network architecture to build a generator and discriminator to learn the non-linear identity and expression representations [19]. Moreover, it is a significant common practice among many non-linear 3DMMs to decouple the identity and expression features from a 3D face shape during the model's construction, a technique exemplified by the work of Cheng *et al.* [19]. Further contributions include the work of Tewari *et al.*, who demonstrated multi-frame video-based, self-supervised training methodology for a deep network that disentangles facial shape, appearance, expression, and illumination [102]. Additionally, Liu *et al.* proposed a framework for learning a non-linear face model by treating 3D scans as unorganized point clouds, thereby transforming them into shape and expression latent representations before reconstructing the 3D shapes [64]. Similarly, Liu *et al.* explored the use of an encoder-decoder network to regress 3D face shapes from 2D face images, effectively disentangling the identity from

non-identity components of 3D face shapes [65].

Recent studies have introduced novel 3DMMs [70, 59, 103, 32, 113, 117, 38, 101] provide unique contributions to 3D face or head modelling. These developments highlight the continuous innovation in the field, offering diverse applications and improved modelling techniques. Our research, while aligning with these innovative algorithms, specifically focuses on constructing shape-based non-linear 3DMMs without incorporating texture. By exploring more semantically segmented representations for 3D expressive faces – disentangling identity and expression, and further segmenting the facial shape into several semantic regions for modelling, our approach distinctively enhances the understanding in the structural aspects of 3D modelling.

2.3 DEEP GENERATIVE NETWORKS

Following recent progress in deep generative networks, the study of 3D shape reconstruction has seen a gradual rise in attention. A desirable generative model is able to synthesise highly realistic and varied 3D shapes. Given that we plan to use the VAE architecture and GAN in the proposed work, we primarily introduce these two generative models and elaborate on their application in 3D shape modelling, with a particular focus on 3D face modelling, as explained in Section 2.3.1 and Section 2.3.2, respectively.

In addition to these two models, diffusion models have emerged as a significant contender. Known for their excellent ability to generate high-resolution 2D images of diverse quality with high fidelity [45], diffusion models have recently attracted considerable attention. Among others, DreamFusion [85] represents a pioneering application of diffusion models to text-to-3D shape synthesis.

2.3.1 VARIATIONAL AUTO-ENCODERS (VAES)

VAE has emerged as one of the most popular and foundational networks across various architectures in the domain of generative models. The general VAE framework is shown in Figure 2.8.

To explore the mathematical details of VAE, we begin with an explanation of the simpler auto-encoder (AE) architecture. The core principle of AE architecture involves employing neural networks as the encoder and decoder component. The encoder reduces the dimensionality of the input data, while the decoder reconstructs

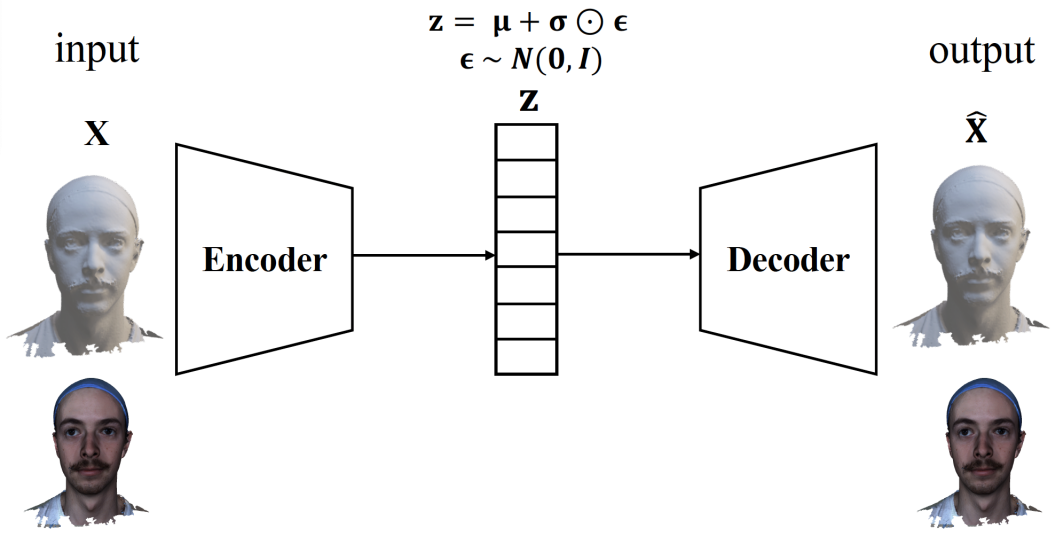

Figure 2.8: General framework of VAE.

the original data from the learnt, comparatively low-dimensional space. A significant challenge in reconstructing data from an AE is to ensure the regularity of the new reconstructed data, which heavily depends on the distribution of the original dataset. In response to this issue, Kingma and Welling proposed VAE [54], which adopted a probabilistic approach to addressing the limitations of traditional AEs by modelling the distribution of input data. Although VAE shares a similar architecture (encoder-decoder) with AE, its encoder represents a distribution over the latent space, rather than encoding a single latent code for each input.

To provide a comprehensive understanding of VAE, we present an overview from a mathematical perspective [53]. We assume the input observed data is denoted by $\mathbf{X}$, and its corresponding latent variable, which is learnt by the encoder, is denoted by $\mathbf{z}$. VAE aims to estimate the parameters θ to model the probability distribution $p(\mathbf{X})$ of the observed data. To facilitate the learning of the latent variable $\mathbf{z}$ in the encoder, a distribution $p(\mathbf{z}|\mathbf{X})$ is approximated, which represents the posterior distribution of the latent variables given the observed data $\mathbf{X}$, as denoted by Equation (2.3). Subsequently, the decoder models the conditional likelihood distribution $p(\mathbf{X}|\mathbf{z})$ to maximise the likelihood of the observed data $p(\mathbf{X})$, which is represented in

Equation (2.4).

$$p(\mathbf{X}) = \int p(\mathbf{z})\, p(\mathbf{X}|\mathbf{z})\, d\mathbf{z}, \tag{2.3}$$

$$\theta^* = \arg\max_{\theta} \int p(\mathbf{z})\, p_\theta(\mathbf{X}|\mathbf{z})\, d\mathbf{z}, \tag{2.4}$$

where the prior distribution $p(\mathbf{z})$ over the latent variables is defined, from which $\mathbf{z}$ is sampled.

Due to the intractable posterior distribution of $p(\mathbf{z}|\mathbf{X})$, the distribution $q_\phi(\mathbf{z}|\mathbf{X})$ is employed to approximate it. We use the Kullback-Leibler (KL) divergence term $D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{X})||p(\mathbf{z}|\mathbf{X}))$ to minimise their difference, as expressed in Equation (2.5).

$$D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{X})||p(\mathbf{z}|\mathbf{X})) = \mathbb{E}_{\mathbf{z}}\left[\log \frac{q_\phi(\mathbf{z}|\mathbf{X})}{p(\mathbf{z}|\mathbf{X})}\right]. \tag{2.5}$$

While the KL term $D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{X})||p(\mathbf{z}|\mathbf{X}))$ is not computable due to the intractable nature of $p(\mathbf{z}|\mathbf{X})$, another KL term $D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{X})||p(\mathbf{z}))$ is applied to measure the similarity between the approximated posterior distribution $q_\phi(\mathbf{z}|\mathbf{X})$ and the prior distribution $p(\mathbf{z})$, as formulated in Equation (2.6),

$$D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{X})||p(\mathbf{z})) = \mathbb{E}_{\mathbf{z}}\left[\log \frac{q_\phi(\mathbf{z}|\mathbf{X})}{p(\mathbf{z})}\right], \tag{2.6}$$

where the prior $p(\mathbf{z})$ is assumed to be a unit Gaussian distribution $\mathcal{N}(0, \mathbf{I})$.

Under the assumption we explain above, the latent variables $\mathbf{z}$ are approximately distributed with $q_\phi(\mathbf{z}|\mathbf{X})$. Therefore, the corresponding log-likelihood of the observed data is rephrased as follows,

$$\begin{aligned}
\log p(\mathbf{X}) &= \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{X})}\left[\log p(\mathbf{X})\right] \\
&= \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{X})}\left[\log \frac{p(\mathbf{z})\, p(\mathbf{X}|\mathbf{z})}{p_\theta(\mathbf{z}|\mathbf{X})}\right] \\
&= \mathbb{E}_{\mathbf{z}}\left[\log p(\mathbf{X}|\mathbf{z})\right] - \mathbb{E}_{\mathbf{z}}\left[\log \frac{q_\phi(\mathbf{z}|\mathbf{X})}{p(\mathbf{z})}\right] + \mathbb{E}_{\mathbf{z}}\left[\log \frac{q_\phi(\mathbf{z}|\mathbf{X})}{p_\theta(\mathbf{z}|\mathbf{X})}\right].
\end{aligned} \tag{2.7}$$

Combining Equations (2.5), (2.6) and Equation (2.7) one can further derive Equation (2.8):

$$\begin{aligned}
\log p(\mathbf{X}) &= \mathbb{E}_{\mathbf{z}}\left[\log p(\mathbf{X}|\mathbf{z})\right] - D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{X})||p(\mathbf{z})) \\
&\quad + D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{X})||p_\theta(\mathbf{z}|\mathbf{X})) \\
&\geq \mathbb{E}_{\mathbf{z}}\left[\log p(\mathbf{X}|\mathbf{z})\right] - D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{X})||p(\mathbf{z})),
\end{aligned} \tag{2.8}$$

where $\mathbb{E}_{\mathbf{z}}\left[\log p\left(\mathbf{X}|\mathbf{z}\right)\right]$ is for recovering the input data from its latent representation by the decoder. We minimize $D_{\mathrm{KL}}\left(q_{\phi}\left(\mathbf{z}|\mathbf{X}\right)||p\left(\mathbf{z}\right)\right)$ to make the posterior distribution $D_{\mathrm{KL}}\left(q_{\phi}\left(\mathbf{z}|\mathbf{X}\right)\right)$ close to the prior distribution $p\left(\mathbf{z}\right)$.

Thus, VAE aims to maximise the Evidence Lower Bound term (ELBO) on the log-likelihood of the observed data X, which is represented in a concise manner, as shown in Equation (2.9):

$$\log p\left(\mathbf{X}\right) = \mathbb{E}_{\mathbf{z}\sim q_{\phi}(\mathbf{z}|\mathbf{X})}\left[\log p(\mathbf{X})\right] \geq \mathrm{ELBO}, \tag{2.9}$$

where the ELBO is defined as the following expectation:

$$\mathrm{ELBO} = \mathbb{E}_{\mathbf{z}}\left[\log p\left(\mathbf{X}|\mathbf{z}\right)\right] - D_{\mathrm{KL}}\left(q_{\phi}\left(\mathbf{z}|\mathbf{X}\right)||p\left(\mathbf{z}\right)\right). \tag{2.10}$$

In other words, VAE optimises parameters θ for the decoder and ϕ for the encoder to maximise the log-likelihood of the observed data $\mathbf{X}$, as expressed as follows,

$$\theta^{*}, \phi^{*} = \arg\max_{\theta,\phi} \log p\left(\mathbf{X}\right). \tag{2.11}$$

Simply put, the negative ELBO is considered as the loss function of VAEs.

As the essential mechanism of VAE has been discussed, it is imperative to explore their deployment across diverse architectures to address a variety of problems, particularly within the context of 3D generative models.

Tan *et al.* designed a mesh VAE network to navigate the probabilistic latent space of 3D surfaces, effectively capturing the most important mesh-based rotation-invariant features [99]. It enabled the learning of reasonable representations for sets of deformation shapes, such as human bodies in different poses, expressive faces and hands with diverse gestures, and facilitated the generation of new shapes not present in the original dataset.

While VAEs excel in reconstructing the complex shapes, they often produce blurry outcomes. To address this, Liu *et al.* proposed an innovative approach that initially learns global latent variables and then integrates them with local latent codes representing a single level of feature abstraction, aiming to reconstruct objects that are realistic rather than blurry [67]. Bagautdinov *et al.* used a VAE for 3D face modelling with multiple levels of latent variables, where lower levels capture global information and high levels focus on local deformations [3]. They performed interpolations of only the higher-level latent variables with fixed lower ones to transition high-frequency details from beardless to bearded faces. If interpolations were done on lower-level variables, the entire face geometry would be changed.

Both Kim *et al.* [51] and Li *et al.* [60] achieved more sophistication of 3D shape reconstruction, conducting VAE-based models using the ShapeNet dataset [14], which includes models of various categories, such as bags, airplanes and lamps. Kim *et al.* integrated an attention-based transformer with VAE architecture, proposing a novel hierarchical VAE that learns latent representations at multiple scales [51]. It enables the capture of coarse-to-fine dependencies among categorical elements, generating high-quality, diverse objects across various categories. Li *et al.* employed the shape primitive-based point-cloud representations and designed a part-aware VAE framework to semantically disentangle the object parts, *e.g.* chairs, into latent spaces [60]. This framework includes 3D point clouds, 3D shape primitives and pose transformations to a canonical coordinate system.

VAE-based models are extensively used in the complex reconstruction of 3D shapes using different representations, *i.e.*, point clouds, voxels, meshes and implicit representations, requiring learning a suitable distribution in the latent space and facilitating applications in shape completion and shape arithmetic. In our architecture, we employ VAEs to learn separate probabilistic latent variables for identity and expressions, as well as facial parts, rather than a global representation for the entire face, in order to capture finer details within each part.

2.3.2 Generative Adversarial Networks (GANs)

In addition to VAEs in the domain of generative models, our work also leverages the GAN [39], a significant model introduced by Goodfellow *et al.* in 2014. The general GAN framework is shown in Figure 2.9.

Comprising a generator and a discriminator, GAN was the first to refer to adversarial examples within the context of generative models, a concept primarily used as the input to classification networks. The core idea of GAN is that the generator learns to capture the distribution of input data and generates new data samples, while the discriminator estimates a probability, indicating to which extent the given sample is built up from the actual data distribution or produced by the generator. Therefore, to produce convincingly realistic samples, the generator aims to maximise the confidence that the discriminator will classify its generated samples as real data, thereby enforcing the generated distribution to closely approximate the data distribution.

Similar to the work in Section 2.3.1, a mathematical explanation to ensure a

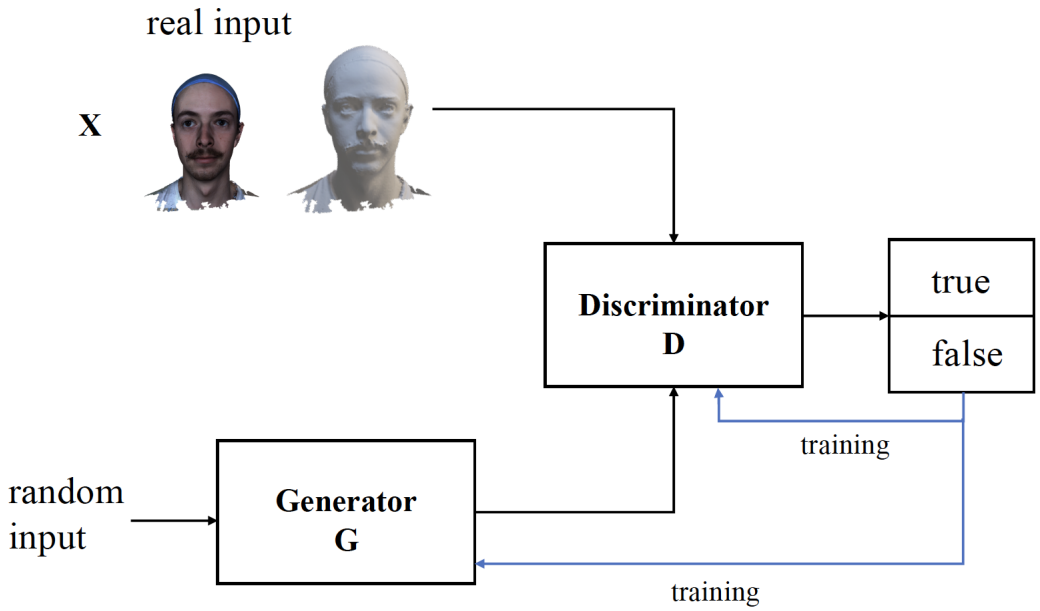

Figure 2.9: General framework of GAN.

thorough and sound introduction will be provided. A prior distribution for the random latent variables in the generator is defined as $p_{\mathcal{G}}(\mathbf{z})$, and the generated samples are denoted by Equation (2.12) as follows,

$$\mathbf{X}_g = \mathcal{G}(\mathbf{z}, \theta), \tag{2.12}$$

where θ denotes the parameters to be learnt in the generator. The discriminator takes the samples as input, which could either be real or generated, and the probability estimated by the discriminator is expressed as follows,

$$p_{\mathcal{D}} = \mathcal{D}(\mathbf{X}, \theta'), \tag{2.13}$$

where θ' denotes the parameters to be learnt in the discriminator. The goal of GAN is for the generator to fool the discriminator into failing to distinguish between true samples (from the real input data) and false ones (generated by the generator). Thus, when expressed mathematically, an effective discriminator for the generator implies:

$$\max_{\mathcal{G}} \min_{\mathcal{D}} \left[\mathbb{E}_{\mathbf{z} \sim p_{\mathcal{G}}(\mathbf{z})} \left[\log \left(\mathcal{D}(\mathcal{G}(\mathbf{z}, \theta), \theta') \right) \right] + \mathbb{E}_{\mathbf{X} \sim p_{\text{data}}} \left[\log \left(1 - \mathcal{D}(\mathbf{X}, \theta') \right) \right] \right], \tag{2.14}$$

where $\mathbf{X} \sim p_{\text{data}}$ denotes true samples following the distribution of input data.

During the initial stages of training, the discriminator can easily classify the two distributions with high confidence, as they may distinctly separated. However, as the iterative 'two-player' competition progresses between the generator and discriminator, it becomes increasingly challenging for the discriminator to accurately identify the generated distribution, enhancing the capabilities of both the generator and discriminator.

GAN-based models have emerged in recent year as pivotal techniques in 3D shape reconstructions, including 3D face reconstruction. Employing the GAN architecture in 3D shape modelling offers several advantages. First, novel 3D objects can be sampled from a latent space distribution, similar to VAEs, with both approaches learning a probability space. Second, the capability of the discriminator to classify makes it well-suited for 3D shape recognition. Below, we introduce some of these methods [107, 121]. Wu *et al.* combined volumetric convolutional networks with generative adversarial nets, encoding 2D images into a probabilistic latent space and reconstructing corresponding 3D objects [107]. A 3D GAN model was trained for each object category, successfully generating high-resolution 3D objects with detailed geometries. In 2018, their research group proposed an adversarial framework for modelling 3D shapes rendered to 2D images, as detailed by Zhu *et al.* [121]. They developed a category-specific model that disentangles latent spaces for shapes, viewpoints and textures separately, employing a GAN for mapping latent representations to 3D voxels, using latent codes for viewpoints to jointly create 2.5 sketches, and leveraging a texture network with texture embeddings to synthesise 2D images.

With the rapid development of GAN and its numerous applications, the challenge of limited supervision during training process has garnered attention. Tang *et al.* proposed a GAN that, instead of mapping latent representations directly to 3D objects, learns local warping functions [100]. For their discriminator, Tang *et al.* utilised point-wise loss functions to ensure tight fitting among local regions for complete object reconstruction, classifying global latent features predicted from both the reconstructed point clouds by the generator and the real sample point clouds. In their generator, they predefined 3D prior distributions for local areas on 3D grids, concatenating these with a globally learnt latent code that was split into local latent codes to predict local warping functions for corresponding point clouds. This innovative lightweight network efficiently produces uniformly distributed 3D point clouds with various resolutions.

For 3D face reconstruction, Moschoglou *et al.* introduced 3DFaceGAN [75], the first GAN specifically designed to model the distribution of 3D facial shapes, capturing high-frequency details. This approach involves using 3D face shapes and corresponding 2D facial UV maps as input to the generator (an AE), with the output from the decoder, as well as the ground truth 3D faces and 2D face UV maps, being fed into a pretrained discriminator. This joint architecture of AE and GAN facilitates the capture and reconstruction of non-linear facial details, such as lips and eyelids, addressing challenges inherent in some linear PCA models. Additionally, Otroshi Shahreza and Marcel proposed a GAN-based framework that focuses on face recognition and learns the mapping from facial templates to the latent space of a pretrained face generation network, thus generating high-resolution face images [78].

All these studies used GAN architecture, with a focus primarily on the reconstruction of 3D objects, including faces. Models were trained to achieve a balance between the generator and the discriminator, enhancing the reconstruction's capability to achieve a level resistant to discriminator classification. Featuring GANs, our work differs from others in the design of the discriminator that identifies pairs of shapes rather than individual ones, enforcing similarity within each pair and being used to disentangle 3D face identity and expressions.

2.4 CLOSELY RELATED LITERATURE

In this section we examine the literature that we believe is most closely related to the contributions of this thesis. These works relate to applications such as 3D facial shape editing and controllable shape deformation, which have played a significant role in driving the development of 3D face modelling. For example, in non-linear 3DMMs, a global latent representation may limit human understanding and control over local or fine-level attributes. Therefore, numerous studies focus on human facial expression analysis that requires an identity-agnostic expression representation. Moreover, there is a growing interest in employing smaller partition schemes and focusing on learning from the entire global latent embeddings for 3D face and full head shapes. In the following, we discuss two types of disentanglement, *i.e.*, identity and expression, and parts-based identity, which are beneficial for improving the modelling of 3D face images.

2.4.1 Facial Identity and Expression Disentanglement

Many of the recent 3D face generative models leverage either VAEs, GANs or a combination of both to deliver the disentanglement of identity and expressions.

Jiang *et al.* developed a non-linear framework to separate 3D face meshes into identity and expression attributes by setting neutral expressions, *i.e.*, identity attributes, as the origin points [46]. They discovered that different individuals sharing the same expressions tend to lie in a similar high-dimensional manifold. Consequently, an expression on the mean face implies the same corresponding expression across different faces. Their expressive latent representations could be learnt from the mean face and then applied to various identities to replicate the expressions on different identities, however, this method does not take into account the uniqueness of expressions as exhibited by different individuals.

Both Sun *et al.* [94] and Taherkhani *et al.* [97] designed conditional pipeline using expressive class labels, employing two decoders to separately learn identity and expression representations. These two models differ in the way to achieve the disentanglement. Sun *et al.* implemented the information bottleneck on identity reconstruction by introducing a mutual information regulariser that eliminates expressive information contained in the identity latent codes [94]. In the meantime, Taherkhani *et al.* employed a combination of supervised VAE and conditional GAN, which not only decouples identity and expressions but also provides subtle control over expressions [97]. Conditional GAN distinguishes input data from real or fake classes and also outputs corresponding identity and expression classes, enabling disentanglement of both 3D geometry shape and appearance.

Olivier *et al.* introduced FaceTuneGAN [77], an innovative adversarial AE architecture equipped with two encoders: one for identity, learning the identity representation for a specific face, and another for expressions, learning the expression representation for a potentially different face. Additionally, there is a decoder that generates faces based on the learnt identity and expression latent vectors. Thus, the ideal outcome for the AE is to reconstruct a face that maintains the identity from the input face to the identity encoder and the expression from the input to the expression encoder. A discriminator is employed to enforce that the generated shapes are realistic and belong to the correct expression class. Abrevaya *et al.* explored the use of an auxiliary classifier GAN (AC-GAN) to factorise the representations, aiming to separate variations within shapes, such as identity and expressions of

faces [1]. Meanwhile, Zhang *et al.* implemented a VAE conbined with graph convolutional networks to model the distribution between identity and expression for 3D face variations, and used adversarial learning to eliminate correlations between these two representations and to ensure their independence [116]. Specifically, the learnt identity and expression latent variables from the same facial mesh were concatenated, representing a coupled distribution, while the independent distribution was represented by the learnt latent variables from different facial meshes. The discriminator was trained on pairs of coupled and independent latent variables to enforce sufficient independence of learnt distributions, effectively disentangling identity and expression in 3D facial models. Kacem *et al.* integrated a graph convolutional autoencoder with a GAN to extract identity representations from expressive 3D faces and reconstructed solely the identity shapes, achieving 3D expressive face neutralisation through joint training of an AE for both expressive and non-expressive shapes [48]. Zhang *et al.* modelled expressions as deviations from identity by a subtraction operation, and extracted an identity-invariant expression latent vector via a deviation learning network with a pseudo-siamese structure [115]. By pretraining an identity model and fixing it during the training of the face model with expressive faces, they diverged from common approaches that typically involve summing the embeddings from identity and expressions, opting instead to use subtraction to obtain the expression representations and finally classify the expression.

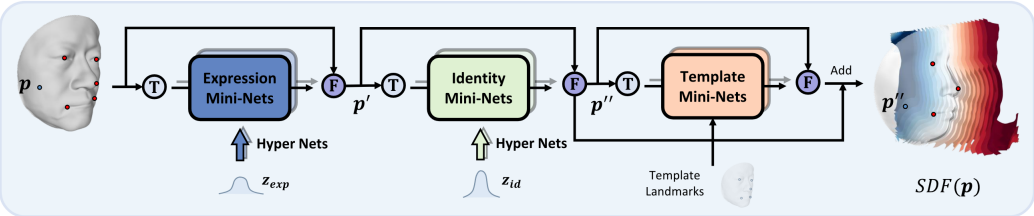

Figure 2.10: The overall pipeline of ImFace [117].

Recent work by Zheng *et al.*, which greatly inspired our architecture design, employs SDF as representations [117]. It is closely related to our research questions RQ1 and RQ2, as outlined in Section 1.1 and Section 1.2 since it utilises implicit representations with the goal of disentangling face identity and expression. This approach builds separate deformation fields, as illustrated in Figure 2.10, enabling the disentanglement of face identities and expressions. They also introduced a data

preprocessing technique for creating pseudo-watertight shapes, effectively addressing the requirement for watertight meshes in preparing SDFs for 3D face shapes. Since achieving watertightness directly from raw data can be trivial, constructing pseudo-watertight shapes offers a practical solution to data processing.

It is noteworthy that most existing methods for 3D face identity and expression disentanglement employ an AE framework in a supervised manner, relying on identity ground truths. However, this may pose a practical limitation in real-world scenarios, as access to identity ground truths may not always be available.

2.4.2 Parts-based Facial Identity Disentanglement

In Section 2.4.1, we introduced multiple methods for 3D facial identity and expression disentanglement. Furthermore, many models classify the corresponding labels for expression embeddings, *e.g.* smiling, crying, shouting, etc., making the generated expressive faces more specific and enabling deformation between different expressions. Therefore, an increasing number of studies have begun to shift focus towards a finer representation of identity, such as learning detailed local representations for facial parts instead of for the entire face.

Recent studies, as highlighted by [34, 33], defined a mesh-convolutional VAE incorporating dense-corresponded points by leveraging known differences and similarities in the latent space to encourage a disentangled representation of identity features. It is primarily related to our research question, RQ2, as discussed in Section 1.2, which focusses on achieving disentanglement of parts-based latent representation, employing explicit representations similar to RQ1 (Section 1.1). Foti *et al.* introduced a novel idea crucial in inspiring our work: mini-batch feature swapping [34]. This strategy swaps features from one mesh to another by replacing the vertices in the selected feature regions, as depicted in Figure 2.11. They defined a mini-batch as size B as a matrix of size $\sqrt{B} \times \sqrt{B}$, wherein each row contains the same mesh with different features, and each column contains different meshes with the same feature. This arrangement allows for the enforcement of similarities row-wise and differences column-wise through specific loss functions.

In the following year, Foti *et al.* continued to improve their work by leveraging spectral geometry, without the need for the curated mini-batch procedure and thereby reducing training time [33]. They designed a novel loss function that encourages latent embeddings to follow the local eigenprojections of identity attributes and

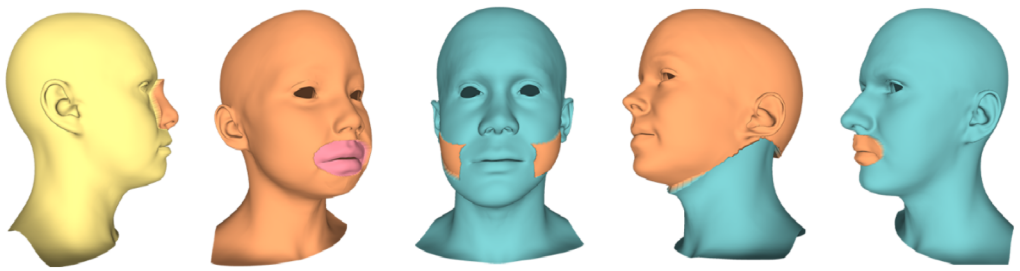

Figure 2.11: Examples of feature swapping in the work of Foti *et al.* [34].

enforces orthogonality between latent embeddings, facilitating disentanglement.

Ghafourzadeh *et al.* presented a face editing system using a part-based 3DMM that segments the face shape into five manually selected non-overlapping parts for localised control [37, 36]. The segmentation is achieved through PCA on 135 3D face meshes. Furthermore, a linear mapping between the anthropometric measurement spaces and the PCA coefficient spaces was established to facilitate intuitive editing. Aliari *et al.* further advanced this topic by proposing an interactive 3D face editing model employing multiple graph-based variational encoders to learn representations of different facial parts [2]. That is, there will be one encoder for each facial part, with a singer decoder to reconstruct whole faces from the concatenated latent representations of each part. This model intrinsically blends all parts without requiring an additional merging process and facilitates predefined vertex-based editing by allowing users to modify predefined vertices, optimising only the corresponding latent variables for those edits. Yan *et al.* also employed sub-models, consisting of six VAEs, to manage different semantic segments of faces based on the Basel Face Model [110], introducing an As-Rigid-As-Possible (ARAP) method [92] to naturally blend facial segments. Their approach maps learnt latent representations to the predefined facial feature measurement spaces, such as nose height, enabling edits based on directional deviations and standard deviations from the mean face shape, rather than directly changing the exact measurement values.

Although the studies by [34, 33, 2, 36, 37, 110, 13] achieved disentanglement of facial parts, their approach to representing 3D shapes in an explicit manner constrained the resolution of the generated faces and required a shared topology across them.

2.5 CRITICAL ANALYSIS

In this chapter, we explored the cornerstone of our research: 3DMMs, which use either explicit or implicit representations to model 3D face shapes. In recent years, numerous studies have shown keen interest in developing deep generative networks based on models such as VAE, GAN or diffusion models to learn latent representations of 3D shapes, enabling the modelling of 3D face or head shapes and generating new unseen shapes. These models show promising performance in both 3D shape reconstruction and generalisation. However, there are still research gaps that inspired our work.

For 3D face modelling, learning either a global latent representation for each face shape or several decoupled latent variables for different facial components, is both effective and popular. For instance, our work, among others, aimed to learn disentangled latent representations for 3D face identity and expressions. Although many focussed on this topic and did achieve the delivery of desirable performance, to the best of our knowledge, none has investigated scenarios in which the ground truth data for face identities (*i.e.*, neutrals) are unavailable.

A commonly used approach involves employing a single encoder to map observed data into two different latent spaces, with two separate decoders to reconstruct the identity and expression from these latent representations. Adversarial learning techniques and specific loss functions are designed to ensure effective disentanglement. However, this approach would rely on the availability of ground truth data for both the observed facial input (to supervise the reconstruction process) and the facial identity (to ensure the effectiveness of the facial identity branch). To bridge this gap, we set out to design a novel network architecture to achieve face identity and expression disentanglement using explicit representations. This architecture will be introduced in detail in Chapter 3.

After successfully disentangling 3D face identity and expressions, we shift our focus towards a division of the global head structure into local semantic parts. To the best of our knowledge, existing research on facial parts-based disentanglement primarily utilises explicit representations. However, these explicit representations typically require a shared mesh topology and restrict the resolution flexibility of the final reconstructed 3D shapes. Given the growing interest in deep networks for implicit representations, which is able to overcome these constraints and offer adaptability in 3D shape modelling, the exploration of models for parts-based disentanglement using

implicit representations remains an open research topic. We, therefore, anticipate developing approaches that leverage these implicit representations, a promising alternative to the explicit representations used in such tasks. We will examine this issue in Chapter 4.

— 3 —

3D Face Disentanglement of Identity and Expression

This chapter is dedicated to the 3D face disentanglement of identity and expressions. We employ adversarial learning and VAE [54] algorithms, as introduced by Kingma *et al.*, to design an effective pipeline for tackling this problem. This approach plays a vital role in understanding a 3D facial image from the shape channel only (*i.e.*, without color-texture), in order to obviate any ambient lighting requirements. Unlike existing 3D face disentanglement that assumes the presence of a corresponding neutral (*i.e.*, identity) face for each subject, our method introduces an identity discriminator to preclude such requirements.

The most immediate problem we face is how to disentangle the 3D shape attributions that derive from a subject's identity from those that result from facial expressions. Such a decomposition has many applications; for example, facial identity and expression interpolations (as illustrated in Figure 3.17 and Figure 3.18), facial expression transfer ([11, 105, 104]), as shown in Figure 3.19, face recognition ([28, 56, 58, 65, 68, 73]), and facial animation [9, 12].

In this chapter, we aim to learn how to disentangle identity and expression features in order to reconstruct 3D human faces, regardless of the availability of neutral faces, corresponding to the identity of the expressive faces. To this end, we propose an adversarial approach that integrates a VAE with an identity discriminator. The VAE incorporates a PointNet-based [86] encoder and two distinct decoders: one for identity and another for expression. These decoders share the same MLP architecture. The same network architecture is used as the base for the identity discriminator, a classifier designed to determine whether a pair of 3D faces shares the same identity.

Our proposed approach is intended to address the limitations of existing 3D face disentanglement methods that feature the presence of a corresponding neutral face for each subject.

We capitalise on the findings of Grasshof *et al.* [40, 41], which indicate that the centre of the expression space is the point of apathy (a state in which all face muscles are relaxed). Our identity discriminator is designed to capture inherent features, *i.e.*, identity features, from various expression faces. The extracted identity parts from the same individual are assumed to be the apathy expression (*i.e.*, emotionless with relaxed facial muscles). In contrast, the identity discriminator is equipped with a task to ensure that different identity representations remain distant from each other. This adversarial process encourages our network to synthesise invariant identity faces for the same subject.

Compared with other methods that require a corresponding neutral face for each subject, we regard the invariant, apathetic identity representations learnt by the discriminator as the 'neutral' face in scenarios where obtaining ground truth neutrals is not feasible. Extensive qualitative and quantitative evaluations validate that our adversarial approach is able to successfully disentangle identity and expression features, and synthesise high-quality 3D face shapes (see Section 3.2.5).

The structure of this chapter is organised as follows. We detail our end-to-end method for 3D facial identity and expression disentanglement in Section 3.1. Section 3.2 evaluates the reconstructed face shapes generated by our method, using both qualitative and quantitative metrics and comparing them to other generative models. Finally, in Section 3.3, we summarise our main contributions in 3D face disentanglement of identity and expression.

3.1 METHODOLOGY

To begin with, we provide a comprehensive explanation of our proposed method for 3D facial identity and expression disentanglement. Figure 3.1 illustrates the overall joint learning pipeline of our method. We present the overall architecture in Section 3.1.1, followed by detailed explanations of the VAE model in Section 3.1.2, the identity discriminator in Section 3.1.3, and the loss functions utilised in our end-to-end training process, which are elaborated on in Section 3.1.4.

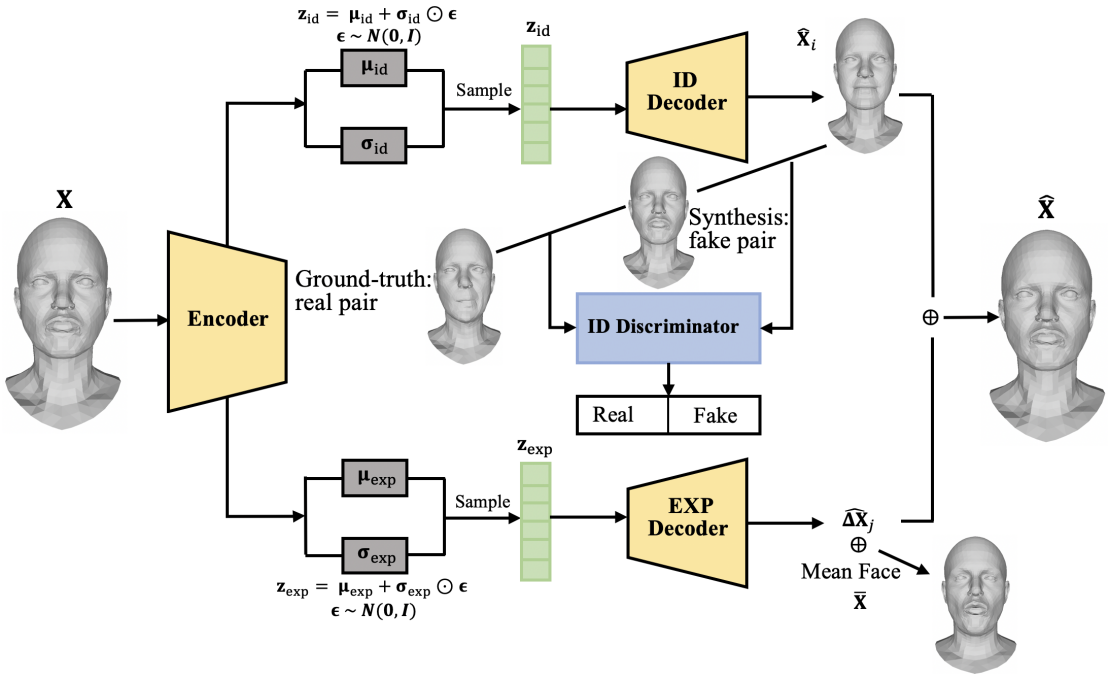

Figure 3.1: A framework for 3D face identity and expression disentanglement. This joint learning network consists of a VAE module for 3D face reconstruction with a discriminator that enforces identity consistency in an adversarial manner to ensure accurate disentanglement of facial identity and expressions [43].

3.1.1 Overall Architecture

We view each aligned and densely corresponded 3D face scan $\mathbf{X}^k_{(i,j)} \in \mathbb{R}^{n\times 3}$ ($k \in [1, \cdots, m]$) as point clouds, where n is the number of vertices, m is the number of input 3D face scans, i denotes the ith identity and j denotes the jth expression. (Note that these scans share the same mesh topology, which is used for visualisation purposes only, *e.g.* in the framework shown in Figure 3.1.)

We simplify $\mathbf{X}^k_{(i,j)}$ to $\mathbf{X}$ in the following. Each instance of $\mathbf{X}$ is divided into two components: the identity part $\mathbf{X}_{(i,\mathrm{a})} \in \mathbb{R}^{n\times 3}$ (where a represents the apathy expression) and the expression deformation part $\Delta\mathbf{X}_j \in \mathbb{R}^{n\times 3}$. We assume that identity and expressions are independent, so that the full face is the sum of the identity shape and the expression deformation, formulated as:

$$\mathbf{X} = \mathbf{X}_{(i,\mathrm{a})} + \Delta\mathbf{X}_j. \tag{3.1}$$

In our architecture, depicted in Figure 3.1, the entire network is designed as GAN in which the autoencoder-decoder network functions as a 'generator' of the

network. We employ a VAE, based on PointNet, to learn the distributions of identity and expression, and sample their latent representations $\mathbf{z}_{\text{id}}$ and $\mathbf{z}_{\text{exp}}$ respectively. Subsequently, two decoders are used to reconstruct the identity shape $\hat{\mathbf{X}}_{(i,\text{a})}$ and expression deformation $\hat{\Delta\mathbf{X}}_j$ from their respective latent vectors $\mathbf{z}_{\text{id}}$ and $\mathbf{z}_{\text{exp}}$. Using Equation (3.1), full faces are synthesised. Detailed descriptive architecture is illustrated in Figure B.1.

Another essential component of the GAN framework is the discriminator and we propose an identity discriminator in our pipeline. The input to this identity discriminator is a face shape *pair* containing a 3D face $\mathbf{X}_{(i_1,j_1)}$ and another 3D face shape with the same identity $\mathbf{X}_{(i_1,j_2)}$, or another 3D face shape with different identities $\mathbf{X}_{(i_2,j)}$. (Note that the j_1 may be equal to the j_2 or j, but the i_1 is not equal to the i_2.) For example, the smiling face $\mathbf{X}_{(i_1,j_1)}$ and crying face $\mathbf{X}_{(i_1,j_2)}$ form a pair, given that they share the same identity $\mathbf{X}_{(i_1,\text{a})}$. Similarly, the smiling/crying face $\mathbf{X}_{(i_1,j)}$ and the angry or smiling face $\mathbf{X}_{(i_2,j)}$ also constitute a pair. This discriminator is pretrained to distinguish a 'real' face shape pair (*i.e.*, same identity) from a 'fake' pair (*i.e.*, different identity). When jointly training the end-to-end GAN model, the original face shape pairs with the same identities ($\mathbf{X}_{(i,j_1)}$, $\mathbf{X}_{(i,j_2)}$) are considered real samples, while those pairs that include predicted identity shapes from the identity decoder ($\mathbf{X}_{(i,j_1)}$, $\hat{\mathbf{X}}_{(i,\text{a})}$) are considered fake. Thus, the generator is encouraged to learn the intrinsic feature of these two pairs, *i.e.*, a common identity, during the process of adversarial learning, enabling the generated identity shape to be least expressive.

The identity discriminator, implemented as a constraint in the entire pipeline, ensures that predicted identity shapes remain neutral. By combining the independent learning of latent representations in separate branches and imposing constraints specifically on the identity part, our method is able to disentangle identity and expression.

3.1.2 Variational Encoder-Decoder Network

Although we aim to disentangle 3D face identity shapes and expressions, 3D face reconstruction is also taken into account. We employ a VAE network in which the encoder is used to predict distributions of latent representations from input point clouds and the decoders are used to synthesise these 3D face shapes. To enable decoupling of identity and expression, the encoder outputs separate distributions

for identities and expressions, and two decoder branches, *i.e.*, the identity decoder (marked as ID Decoder in Figure 3.1) and the expression decoder (marked as EXP Decoder in Figure 3.1), receive their corresponding sampled representations and individually reconstruct 3D identity face shapes and expression deformations.

The VAE models the probability $P(\mathbf{X})$ of the input 3D face shapes and we assume that 3D face shapes are determined by latent features $\mathbf{z}_{\text{id}}$ and $\mathbf{z}_{\text{exp}}$ representing identity and expression, respectively. This generative model estimates parameters that maximise the likelihood of 3D face identities $p_\theta(\mathbf{X}_{(i,\text{a})})$ and expressions $p_{\theta'}(\Delta\mathbf{X}_j)$, as follows:

$$p_\theta(\mathbf{X}_{(i,\text{a})}) = \int p_\theta(\mathbf{z}_{\text{id}})\, p_\theta\left(\mathbf{X}_{(i,\text{a})}|\mathbf{z}_{\text{id}}\right) d\mathbf{z}, \tag{3.2}$$

$$p_{\theta'}(\Delta\mathbf{X}_j) = \int p_{\theta'}(\mathbf{z}_{\text{exp}})\, p_{\theta'}(\Delta\mathbf{X}_j|\mathbf{z}_{\text{exp}})\, d\mathbf{z}, \tag{3.3}$$

where $p_\theta\left(\mathbf{X}_{(i,\text{a})}|\mathbf{z}_{\text{id}}\right)$ and $p_{\theta'}(\Delta\mathbf{X}_j|\mathbf{z}_{\text{exp}})$ are defined for the identity decoder and expression decoder, respectively. We assume a unit Gaussian distribution for the prior distributions $p_\theta(\mathbf{z}_{\text{id}})$ and $p_{\theta'}(\mathbf{z}_{\text{exp}})$.

Due to the intractable posterior $p\left(\mathbf{z}_{id}|\mathbf{X}_{(i,\text{a})}\right)$, the distribution $q_\phi\left(\mathbf{z}_{id}|\mathbf{X}_{(i,\text{a})}\right)$ is defined in the identity encoder to approximate $p\left(\mathbf{z}_{id}|\mathbf{X}_{(i,\text{a})}\right)$. We use the Kullback-Leibler (KL) divergence term $D_{KL}\left(q_\phi\left(\mathbf{z}_{id}|\mathbf{X}_{(i,\text{a})}\right)||p\left(\mathbf{z}_{id}|\mathbf{X}_{(i,\text{a})}\right)\right)$ to minimize their difference. Similar decision is also made for $p(\mathbf{z}_{exp}|\Delta\mathbf{X}_j)$.

In our model, the VAE is assumed to estimate parameters θ and ϕ in order to maximise the log-likelihood of 3D facial identities and expressions. This is intended to maximise the Evidence Lower Bound (ELBO).

3.1.3 Adversarial Training

Face identity shapes and expression deformations are sampled from distributions $p\left(\mathbf{X}_{(i,\text{a})}|\mathbf{z}_{\text{id}}\right)$ and $p(\Delta\mathbf{X}_j|\mathbf{z}_{\text{exp}})$, respectively. To better decouple identity shapes from expressions, an adversarial training process is employed.

Our proposed identity discriminator, abbreviated as $\mathbf{D}_{\text{ID}}$ hereafter, is trained to differentiate between real and fake samples. Additionally, we input a pair of 3D face shapes into the identity discriminator that determines whether the input pair shares the same identity shape. For instance, during the pretraining of our identity discriminator, if we input a face scan pair, denoted as $\mathbf{X}_{(i,j_1)}$ and $\mathbf{X}_{(i,j_2)}$, sharing the same identity $\mathbf{X}_{(i,\text{a})}$, the identity discriminator is expected to classify this pair into the real class (note that the subscripts j_1 and j_2 may refer to the same

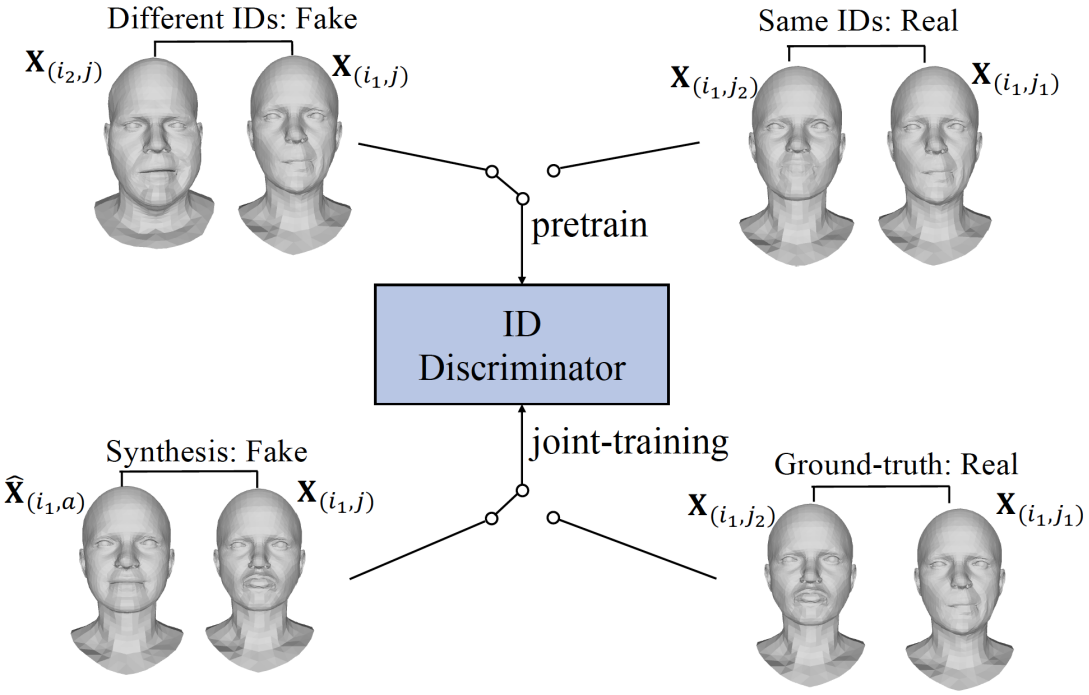

Figure 3.2: The pretrained input pairs and the joint end-to-end training input pairs of the identity (ID) discriminator.

expression). Conversely, a face pair $\mathbf{X}_{(i_1,j)}$ and $\mathbf{X}_{(i_2,j)}$ should be classified into the fake class due to the fact that one comes with the identity $\mathbf{X}_{(i_1,\mathrm{a})}$ and another with the identity $\mathbf{X}_{(i_2,\mathrm{a})}$, as illustrated in Figure 3.2 (note that the subscripts i_1 and i_2 are different identities). During the end-to-end network training process, the identity discriminator tries to enhance the correct separation of identity and expression. To achieve this, we construct a fake pair by combining the predicted identity shape with the original expressive face shape.

In facial expression analysis [15, 16, 90], the latent variables that represent identity and facial expression, lie on a manifold in high dimensional space, as illustrated in Figure 3.3. Stella *et al.* proposed the point of apathy to be the centre of expressions and the expressions trajectories obtained by varying the strength of human emotion originate from this point [40, 41], as shown in Figure 3.4. Based on such observations, we found that it is the apathetic expression that provides implicit connections between various expressions of faces with the same identity. Our adversarial process encourages common information from a face shape pair with the same identity to be retained. If we only compare the smile expression with

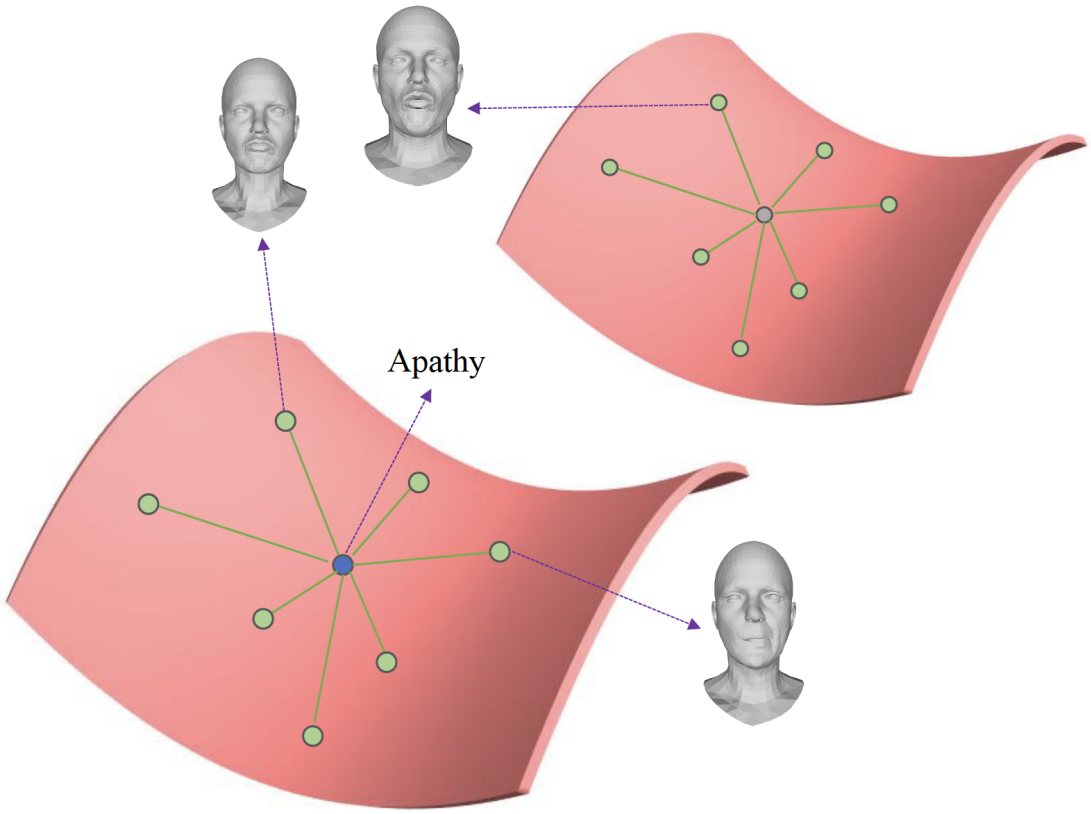

Figure 3.3: 3D face manifold space. Two expressive manifold spaces are shown to represent two individual faces. The central point refers to the apathy expression, with various expressions situated around including 'right jaw' expression.

the surprised expression from the same individual, not only their same identity but their similar expression deformations, *e.g.* mouth opening, will be recorded by the discriminator. However, there are several shape pairs from the same identity and their expressions are distributed in divergent directions that intersect at the point of apathy - so the discriminator will ultimately retain all pairs' common information - *apathy*. Thus, the identity discriminator has the ability to capture similar latent features, *i.e.*, identity features, between pairs belonging to the same subject, and to enforce these features to be close to the *apathy* faces.

In a GAN framework, a generator and discriminator are trained in an adversarial manner. For our network, the desired outcome is that the synthesised face identity shapes from the identity decoder can 'fool' the identity discriminator. This implies that the predicted neutral face shapes should closely resemble the corresponding facial identity ground truths, making the discriminator believe that they belong to

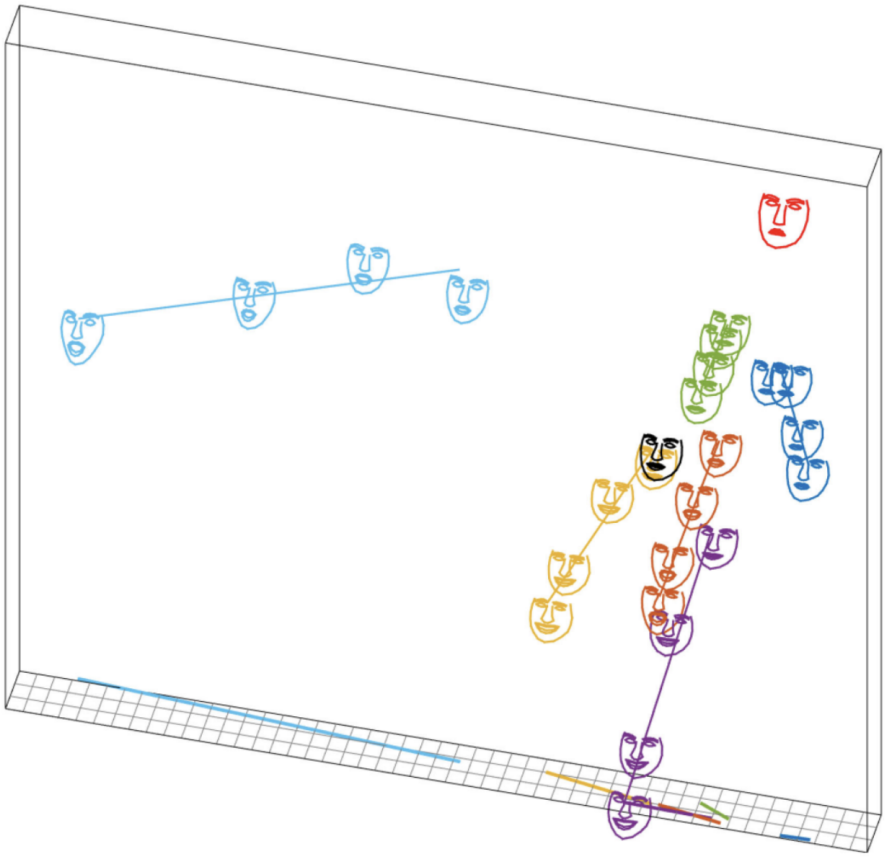

Figure 3.4: A figure from [41] - the expression affine subspace for the BU3DFE dataset. A special point, represented as the top-right red face and defined as the 'point of apathy', is different from the 'neutral' face, represented in black. Within this subspace, each expression is linearly distributed based on its intensity, and different expressions disperse from the origin point, *i.e.*, the 'point of apathy'.

the 'real' class.

During the adversarial process, the discriminator leverages a loss function that enables the identity decoder distribution $p\left(\mathbf{X}_{(i,\mathrm{a})}|\mathbf{z}_{\mathrm{id}}\right)$ to approximate the face identity distribution $p\left(\mathbf{X}_{(i,\mathrm{a})}\right)$. The loss function used to jointly train the generator and discriminator is defined as follows:

$$J_{\mathrm{adv}} = \max_{\theta_{\mathrm{d}}} \Bigg[\mathbb{E}_{\left(\mathbf{X}_{(i,j_1)},\mathbf{X}_{(i,j_2)}\right)\sim p_{\mathrm{same}}} \left[\log\left(\mathbf{D}_{\mathrm{ID}}\left(\mathbf{X}_{(i,j_1)}, \mathbf{X}_{(i,j_2)}; \theta_{\mathrm{d}}\right)\right)\right] + \mathbb{E}_{\mathbf{X}\sim p_{\mathrm{data}}}\left[\log\left(1 - \mathbf{D}_{\mathrm{ID}}\left(\mathbf{G}\left(\mathbf{X}\right), \mathbf{X}; \theta_{\mathrm{d}}\right)\right)\right] \Bigg], \tag{3.4}$$

where p_{same} is the distribution of all same-identity 3D face pairs, and p_{data} is the distribution of all input expressive faces. The pair sampled from p_{same}, *i.e.*, $\left(\mathbf{X}_{(i,j_1)}, \mathbf{X}_{(i,j_2)}\right)$

in Equation (3.4), can be considered two 3D faces sampled from the original dataset. This implies that the two sampled faces share the same identity but may have different expressions. However, having different expressions is not mandatory for these two face shapes; the only feature obtained through multiple samplings that needs to be consistent, is the identity.

In Equation (3.4), $\mathbf{G}(\mathbf{X})$ represents the VAE, *i.e.*, the generator, and its output from the identity decoder is the synthesised identity shape denoted as $\hat{\mathbf{X}}_{(i,\mathrm{a})}$. The terms $\mathbf{D}_{\mathrm{ID}}\left(\mathbf{X}_{(i,j_1)}, \mathbf{X}_{(i,j_2)}; \theta_{\mathrm{d}}\right)$ and $\mathbf{D}_{\mathrm{ID}}\left(\mathbf{G}(\mathbf{X}), \mathbf{X}; \theta_{\mathrm{d}}\right)$ are used to refer to the real pairs and fake pairs, respectively, as depicted in the joint-training branch in Figure 3.2. Thus, the first term of Equation (3.4) is the expectation on real expressive faces and aims to maximise the probability that $\mathbf{D}_{\mathrm{ID}}$ correctly classifies these pairs as real. Conversely, the second term represents the generated pairs and aims to maximise the probability that $\mathbf{D}_{\mathrm{ID}}$ correctly classifies the pairs generated by $\mathbf{G}$ as fake.

During joint training, both $\mathbf{D}_{\mathrm{ID}}$ and $\mathbf{G}$ are updated iteratively. The generator tries to improve its ability to create realistic fake pairs, which in our context, facilitates the neutralisation for the identity generator. Meanwhile, the discriminator works hard to enhance its capacity to distinguish the real pairs from the fake ones. This adversarial process finally results in high-quality and neutralised synthetic data.

3.1.4 End-to-End Loss Function Terms

We define five components in our loss function required to train our end-to-end network that focusses on 3D face reconstruction and identity-expression disentanglement. The overall loss function is represented as:

$$L_{\mathrm{total}} = \lambda_1 L_{\mathrm{recon}} + \lambda_2 L_{\mathrm{KL}} + \lambda_3 L_{\mathbf{D}_{\mathrm{ID}}} + \lambda_4 L_{\mathrm{neu}} + \lambda_5 L_{\mathrm{lap}}, \tag{3.5}$$

where λ_{1-5} are hyperparameters chosen to balance the five losses. To tune these parameters, we employ a small subset of the training data to empirically determine validated values, which are then applied to train the whole training set. Specifically, L_{recon} is the Mean Squared Error (MSE) for 3D face reconstruction. The L_{KL} loss represents the negative ELBO. This loss consists of two KL terms, one for the identity and the other for the expression, constraining the posterior distribution to remain close to the unit Gaussian distribution $\mathcal{N}(0, I)$. The KL loss is defined as follows:

$$L_{\mathrm{KL}} = L_{\mathrm{KL\_ID}} + L_{\mathrm{KL\_EXP}}. \tag{3.6}$$

Furthermore, $L_{\mathbf{D}_{\mathrm{ID}}}$ is simplified from Equation (3.4) by using cross-entropy:

$$L_{\mathbf{D}_{\mathrm{ID}}} = -\left[y \log\left(\mathbf{D}_{\mathrm{ID}}\left(\mathbf{X}_{(i,j_1)}, \mathbf{X}_{(i,j_2)}\right)\right) + (1-y)\log\left(\mathbf{D}_{\mathrm{ID}}\left(\hat{\mathbf{X}}_{(i,\mathrm{a})}, \mathbf{X}_{(i,j)}\right)\right)\right], \quad (3.7)$$

where y is the label (1 for the 3D face pair $\left(\mathbf{X}_{(i,j_1)}, \mathbf{X}_{(i,j_2)}\right)$ sampled from ground truth data, and 0 for the 3D face pair including the predicted identity shape $\hat{\mathbf{X}}_{(i,\mathrm{a})}$). L_{neu} is the L1 loss between $\mathbf{z}_{\mathrm{id}}$ and $\hat{\mathbf{z}}_{\mathrm{id}}$. The identity latent representation, $\mathbf{z}_{\mathrm{id}}$, is obtained from the encoder, which encodes an expressive face shape, as illustrated in Figure 3.1. $\hat{\mathbf{z}}_{\mathrm{id}}$ is also the output of the encoder, but with a different input; in this case, the VAE processes the predicted facial identity shape, denoted as $\hat{\mathbf{X}}_i$ in Figure 3.1. Specifically, after generating the predicted facial identity shapes $\hat{\mathbf{X}}_i$, the VAE uses these shapes as input to generate the corresponding identity latent representations $\hat{\mathbf{z}}_{\mathrm{id}}$. To minimise the mean curvature and ensure smoothness of generated 3D faces, we employ the Laplacian regularisation loss $L_{\mathrm{lap}} = \|L\mathbf{X}\|_2$, where L is the discrete Laplace-Beltrami operator [49].

To further disentangle identity shapes from expression components in our network, we employ an identity discriminator in conjunction with an additional MSE loss for expressions. The reconstruction loss L_{recon} in Equation (3.5) is composed of the MSE for full faces L_{full}, and the MSE for expressions L_{exp}, as detailed in Equation (3.8). When computing L_{exp}, our approach can be applied in two scenarios:

- **With Corresponding Neutral Faces**: When corresponding neutral faces (the facial identity without expressions) are available for the full faces, we can obtain specific expression ground truth for each subject by subtracting the corresponding neutral ground truth from the full face. This ground truth is then used to compute the expression MSE against the predicted expression deformation, ensuring that the generated expressions closely match the ground truths.
- **Without Neutral Faces**: In cases where neutral faces are not available, based on the assumption that expressions of different individuals are similar in the expressive manifold, an expression from a specific individual is considered analogous to the same expression on a mean face. Therefore, we use the expressions from the global mean face as ground truth data to compute the expression MSE.

The approach is formulated as follows:

$$L_{\mathrm{recon}} = \alpha_1 L_{\mathrm{full}} + \alpha_2 L_{\mathrm{exp}}, \quad (3.8)$$

where α_1 and α_2 hyperparameters that balance the aforementioned two types of MSE loss functions. L_{full} represents the loss between the expressive input full faces and the corresponding full faces predicted by the VAE pipeline.

3.2 EVALUATION

In this section, we conduct a comprehensive experimental evaluation of our proposed method for 3D face reconstruction and identity-expression disentanglement. To start with, we provide an overview of the datasets, detail the implementations, and introduce the evaluation metrics. We then compare our methods, in scenarios either with or without neutral ground truths, against baselines across three public datasets. Additionally, we conduct ablation studies to analyse the contributions of each component in our architecture design. These studies help to understand how individual component contributes to the overall performance of our method in a collective manner. To further demonstrate the utility and effectiveness of our approach, we showcase its various applications, including expression transfer, expression interpolation and face recognition. This comprehensive evaluation not only validates our method's performance but also illustrates its versatility in practical applications.

3.2.1 DATASETS

We use three public datasets in our experiments: BU3DFE [114], CoMA [88] and FaceScape [112]. For the BU3DFE and CoMA datasets, we adopt a training-to-test ratio of 9:1. For the FaceScape dataset, we choose the ratio of 7:3. These division ratios are intended to align with those used by baseline methods. Each face scan from these three datasets is normalised to fit within a unit sphere with a diameter of 1cm.

CoMA dataset [88] contains motion sequences of 20,466 meshes, captured from 12 different individuals. Each subject in the dataset performs 12 extreme, asymmetric facial expressions, with significant deformations of facial tissue. The specific expressions include bareteeth, cheeks in, eyebrow, high smile, lips back, lips up, mouth down, mouth extreme, mouth middle, mouth side and mouth up. Following the same partitioning scheme in [88], we divide these meshes into a training and test set. These sequences are organised in alphabetical order, and we select 10

frames from every 100 frames as test samples to guarantee that the test set is typical and unbiased. The training data consists of 18,422 meshes, and the test set includes 2,040 meshes. This division helps the training of our network on a range of diverse expressions and facilitates a robust evaluation on the test set.

BU3DFE dataset [114] consists of 3D facial scans from 100 subjects, with a gender distribution of 56% female and 44% male, ranging in age from 18 to 70 years and representing a variety of ethnic backgrounds. This dataset includes 2,500 facial meshes in total. Each subject is asked to perform seven different expressions: happiness, disgust, fear, angry, surprise and sadness. With the exception of the neutral expression, each of the other six expressions is captured at four levels of intensity, resulting in 25 expressive meshes for each individual. Following one of the baselines proposed in [116], we select the first 10 subjects for our test set while keeping the rest for training purposes. There are 2,247 meshes in the training set and 250 meshes in the test set. It is important to note that the identities in the test set are unseen in the training set.

FaceScape dataset [112] includes 3D face scans from 847 subjects, aged between 16 and 70 years old. Each subject is requested to perform 20 expressions, including mouth-opening, smiling, eyes-closing, kissing, and others. For our study, we randomly select 30% of these subjects for the test set, with the remaining subjects used for training. Thus, there are 11,812 and 5,055 meshes in the training and test set, respectively.

3.2.2 Implementation Details

In the FaceScape dataset, the face mesh of each subject comprises 26,317 vertices and 52,261 faces, which gives rise to significant challenges in terms of GPU memory usage and computational time during training. To mitigate it, we simplify these meshes using a quadric-based edge collapse strategy [35], reducing the target number of faces to 9,000. After this simplification, each face scan contains 4,547 vertices and 8,999 faces. This reduction greatly enhances the training efficiency, though it may slightly compromise the overall smoothness of the meshes. The differences between the original and simplified face scans are illustrated in Figure 3.5. A reduction in surface smoothness is mainly observable in the nose and jaw regions. However, the savings in time and memory consumption are considerable and this does not have a major adverse impact on identity-expression disentanglement.

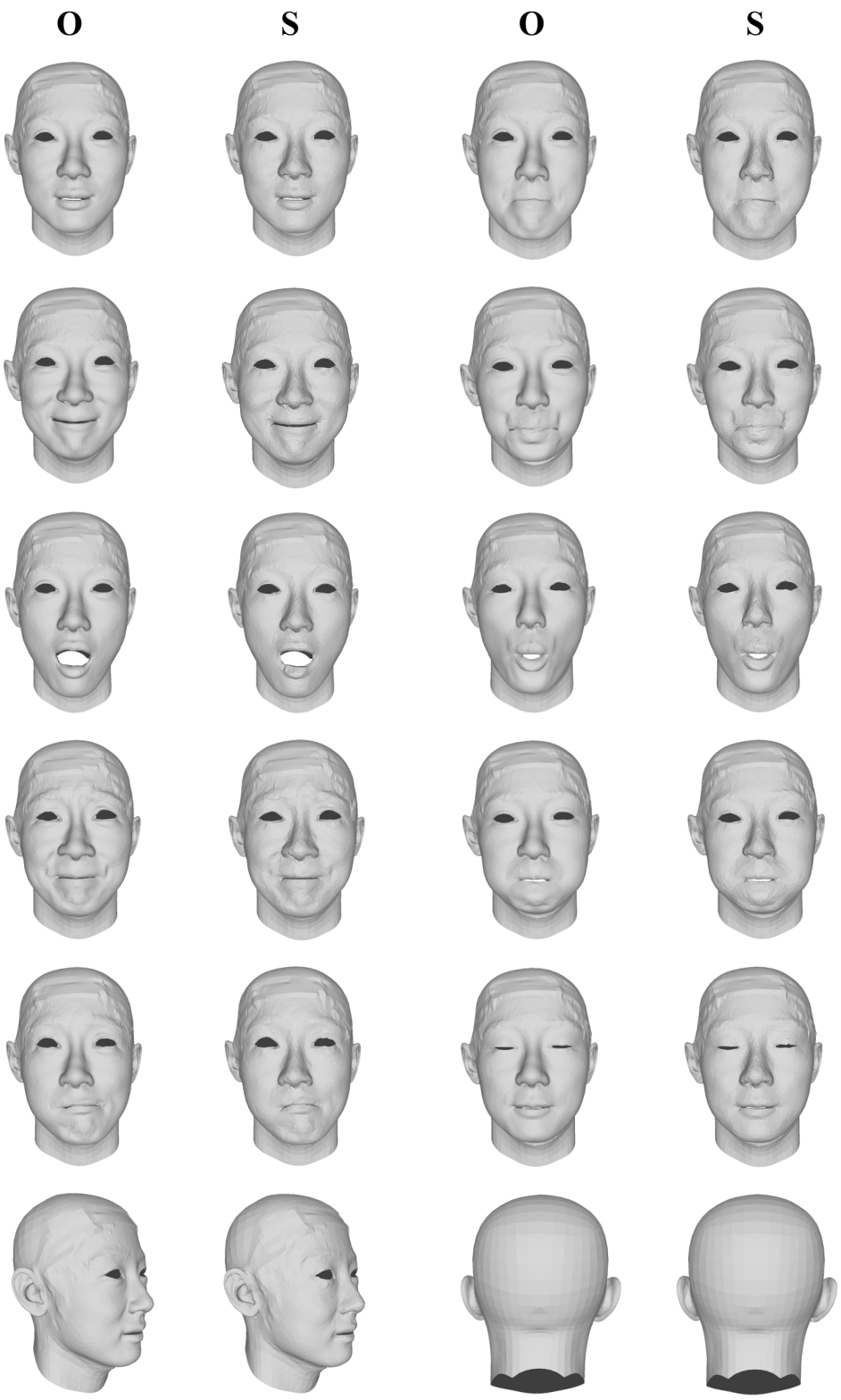

Figure 3.5: A comparison between the original face scans and the preprocessed expressive face scans from FaceScape. Frontal views of randomly selected expressive faces are displayed, as well as side and back views. 'O' denotes the original face scans, and 'S' represents the simplified versions.

We pretrain a PointNet-based network as the identity discriminator. As shown in Figure 3.2, during each training iteration, we sample a 3D face scan from the training dataset to construct a pair with a specific input face shape. If the subject identities of the face shapes in a pair are the same, the pair is labelled as 'True', *i.e.*, a real pair in Figure 3.2. Otherwise, 'False', *i.e.*, a fake pair in Figure 3.2. To ensure the existence of a 'True' labelled pair, we first sample from the subset faces with the same identity as the input, and then sample from the remaining dataset, excluding those with the same identity. Therefore, we can manage to maintain a ratio of 1:1 between the real and the fake pairs during the pretraining process of the identity discriminator.

After the pretraining, our identity discriminator is able to identify the same subject with different expressions. We then employ this pretrained identity discriminator for the initialisation of our sequentially joint end-to-end training. To achieve the facial expression neutralisation, we further explore alternative pairing strategies to simultaneously train the encoder-decoder and the identity discriminator, aiming to improve their performance in identity-expression disentanglement.

In the end-to-end training process, the 'True' pair, similar to the one used in pretraining, consists of the input ground truth 3D face shape and another face shape sampled from the training data, both with the same identity. The key change in the end-to-end training is the construction of the 'False' pair. Unlike that in the pretraining step, a 'False' pair is designed by combining the predicted identity shape, which is the output from the identity decoder, with the original face shape.

As discussed in the Section 3.1.3, adversarial learning is beneficial for the generator to reconstruct more realistic shapes. In our specific context, this, however, implies capturing the inherent nature of two different expressive shapes of the same subject. The identity branch aims to reconstruct a face without any expressions, which can also be regarded as a face with a 'special' expression. Combining with separate decoder components, adversarial learning helps to make the identities and expressions independent.

For a fair comparison with other methods, we set the PointNet-based encoder with four identity latent dimensions and four expression latent dimensions for the CoMA dataset. For the BU3DFE dataset we use 40 dimensions for each latent vector, whereas for the FaceScape dataset, 64 dimensions.

Different hyperparameters are explored for each dataset to optimally balance each loss component. Specifically, λ_1 is set to 250 for BU3DFE and 5000 for both

CoMA and FaceScape. For λ_3, we use 5×10^{-4} on CoMA and FaceScape, and 1×10^{-3} on BU3DFE. For the FaceScape and CoMA dataset, the Laplacian loss is exclusively used to enforce the smoothness of predicted identity faces, when neutral ground truths are unavailable.

We implement our network using PyTorch [80] and run it on an NVIDIA A40 system. The identity discriminator is pretrained with a batch size of 32, and 50, 100, 100 epochs for the CoMA, BU3DFE, and FaceScape datasets, respectively. For the training of the joint end-to-end network, we adopt different number of epochs and batch sizes for each dataset. Specifically, we train for 280 epochs with a batch size of 8 on BU3DFE, 280 epochs with a batch size of 32 on FaceScape, and 300 epochs with a batch size of 32 on CoMA. The Adam optimiser [52] is used, with an initial learning rate of 1×10^{-4} and a learning rate decay factor of 0.7 every 50 epochs. Additionally, we conduct each leave-one-out experiment three times and report the average results to ensure reliability and robustness for our architecture.

3.2.3 Evaluation Metrics

For a fair comparison, we adopt the evaluation metrics used in [46, 48, 116], which include both reconstruction and disentanglement metrics. Given that our method is based on point clouds and all face shapes are densely corresponded, we measure the reconstruction error, E_{rec}, as the average vertex distance between the synthesised 3D face shapes $\hat{\mathbf{X}}$ ($\hat{\mathbf{X}}_{(i,j)}$, the sum of $\hat{\mathbf{X}}_{(i,\text{a})}$ and $\Delta\hat{\mathbf{X}}_j$) and the original 3D face shapes $\mathbf{X}$ ($\mathbf{X}_{(i,j)}$). The reconstruction error is calculated using the following equation:

$$E_{\text{rec}}(\mathbf{X}, \hat{\mathbf{X}}) = \frac{1}{n} \sum_{1}^{n} \left\| \mathbf{X} - \hat{\mathbf{X}} \right\|_2, \tag{3.9}$$

where n represents the number of vertices in $\mathbf{X}$. We report both the mean and the median of this average vertex distance.

The standard deviation of reconstructed identity shapes from 3D faces with the same identity is defined as the disentanglement error E_{dis}. This metric is designed to evaluate the stability of the disentanglement process. In a given test set, raw 3D faces with various identity shapes are represented as $\mathbf{X}_{(i_1,j)}$, $\mathbf{X}_{(i_2,j)}$, $\cdots$, $\mathbf{X}_{(i_N,j)}$, where N denotes the number of different identities in the dataset. For example, the raw 3D faces that share the same identity $\mathbf{X}_{(i_1,\text{a})}$ but own different expressions $\Delta\mathbf{X}_j$. In this context, we assume that $\mathbf{X}_{(i_1,j_1)}$ represents a '*happy*' $\mathbf{X}_{(i_1,\text{a})}$, and $\mathbf{X}_{(i_1,j_2)}$ represents a '*sad*' $\mathbf{X}_{(i_1,\text{a})}$. The E_{dis} metric is applied to measure whether the predicted

identity shapes $\hat{\mathbf{X}}_{(i_1,\mathrm{a})}$ from these two different expressive shapes $\mathbf{X}_{(i_1,j_1)}$ and $\mathbf{X}_{(i_1,j_2)}$ are consistent. The disentanglement error E_{dis} is computed as follows:

$$E_{\mathrm{dis}} = \sigma\left(\left\|\hat{\mathbf{X}}_{(i,\mathrm{a})} - \bar{\mathbf{X}}_{(i,\mathrm{a})}\right\|_2\right), \tag{3.10}$$

where $\bar{\mathbf{X}}_{(i,\mathrm{a})}$ represents the mean of predicted identity shapes $\hat{\mathbf{X}}_{(i,\mathrm{a})}$ from $\mathbf{X}_{(i,j)}$ ($j \in [1, \cdots, K]$), and K denotes the number of expression types. We opt not to evaluate the predicted expression shapes quantitatively, as the same expressions vary among subjects both due to anatomical differences and varying levels of expression intensity.

The average vertex distance, denoted as $\mathrm{AVD}_{\mathrm{neu}}$, is employed to evaluate both the identity reconstruction and disentanglement processes. This metric, defined in Equation (3.11), is used to evaluate how closely the predicted shapes approximate the identity ground truths.

$$\mathrm{AVD}_{\mathrm{neu}} = \frac{1}{n}\sum_{1}^{n}\left\|\mathbf{X}_{(i,\mathrm{a})} - \hat{\mathbf{X}}_{(i,\mathrm{a})}\right\|_2, \tag{3.11}$$

where n represents the number of vertices in $\mathbf{X}_{(i,\mathrm{a})}$.

3.2.4 Comparative Studies

We conduct a comparative experiment of our method with five state-of-the-art 3D face disentanglement methods. Specifically, we compare against FLAME [61], the method proposed by Jiang *et al.* [46] (referred to as Jiang's work hereafter), and DI-MeshEncoder [116] on the CoMA and BU3DFE datasets. We also compare our method with the method proposed by Kacem *et al.* [48] (referred to as Kacem's work hereafter) and Convolutional Mesh Autoencoder (Conv-MeshAE) [88] on the FaceScape dataset. We carefully report results published in [116] as we use the same training and test sets.

The FLAME model represents 3D faces in identity, pose, and facial expressions separately, using a learned shape space for identity variations and expression deformations to capture the non-rigid deformations of faces. Jiang's work and DI-MeshEncoder adopt a Graph Convolutional Network (GCN) based auto-encoder to reconstruct 3D face shapes and to decouple identity from expression attributes. Kacem *et al.* deploy a GCN network and a discriminator for expression neutralisation and face recognition. Conv-MeshAE, mainly a 3D reconstruction method rather

than a disentanglement approach for identity and expression shapes, employs a GCN architecture to map 3D face shapes into a non-linear latent space. Following [48], we consider pairs of expressive and neutral faces as input and ground truths for Conv-MeshAE.

Our method uses a widely-adopted auto-encoder architecture based on PointNet. Moreover, unlike the methods we compare against, our approach does not have a strict requirement on the availability of neutral ground truths. Our discriminator is designed to work directly on raw 3D face data, which differs from [116] that is inspired by Kim and Mnih's work [50] and uses a discriminator to enforce independence between two distributions. Furthermore, our work is different from [48] in which a discriminator in the latent space is proposed to enable a valid translation from expressive to neutral representations.

3.2.5 Results and Discussions

3.2.5.1 Quantitative Results

Table 3.1: 3D face shape reconstruction results (E_{rec}) and disentanglement results (E_{dis}) on the CoMA dataset, compared with FLAME [61], Jiang's work [46], DI-MeshEnc [116] and Conv-MeshAE [88]. All errors are measured in millimeters.

Method	E_{rec} (mm)		E_{dis} (mm)	
	mean ± std	median	mean	median
FLAME [61]	1.451 ± 1.649	0.871	0.599	0.591
Jiang's work [46]	1.413 ± 1.639	1.017	0.064	0.062
DI-MeshEncoder [116]	0.665 ± 0.748	**0.434**	0.019	0.020
Conv-MeshAE [88]	—	—	0.313	0.317
Ours	0.783 ± 0.225	0.772	0.176	0.180
Ours+ne-gt	**0.651 ± 0.208**	0.625	**0.014**	**0.013**

Reconstruction Analysis The quantitative results E_{rec} of 3D full face reconstruction on the CoMA and BU3DFE dataset, in comparison with FLAME, Jiang's work and DI-MeshEncoder, are presented in Table 3.1 and Table 3.2, respectively. The E_{rec} results for Conv-MeshAE are not reported because this method generates identity shapes, rather than reconstructing the original input faces.

Table 3.2: 3D face reconstruction results (E_{rec}) and disentanglement results (E_{dis}) on the BU3DFE dataset, compared with FLAME [61], Jiang's work [46], DI-MeshEnc [116] and Conv-MeshAE [88]. All errors are measured in millimeters.

Method	E_{rec} (mm)		E_{dis} (mm)	
	mean ± std	median	mean	median
FLAME [61]	2.596 ± 2.055	2.055	0.600	0.632
Jiang's work [46]	2.054 ± 1.199	1.814	0.611	0.590
DI-MeshEnc [116]	1.551 ± 0.924	1.375	0.361	**0.327**
Conv-MeshAE [88]	—	—	0.361	0.377
Ours	**1.421 ± 0.412**	**1.306**	0.443	0.439
Ours+ne-gt	1.500 ± 0.423	1.467	**0.348**	0.339

Table 3.3: 3D face reconstruction results (E_{rec}) and disentanglement results (E_{dis}) on the FaceScape dataset, compared with Conv-MeshAE [88]. All errors are measured in millimeters.

Method	E_{rec} (mm)		E_{dis} (mm)	
	mean ± std	median	mean	median
Conv-MeshAE [88]	—	—	0.64	0.62
Ours	**1.157 ± 0.286**	**1.109**	0.77	0.76
Ours+ne-gt	1.370 ± 0.369	1.307	**0.57**	**0.55**

Table 3.4: Average vertex distance of identity shapes (AVD_{neu}) on the BU3DFE and FaceScape datasets, compared with Kacem's work [48] and Conv-MeshAE [88]. All errors are measured in millimeters.

Method	BU3DFE		FaceScape	
	mean ± std	median	mean ± std	median
Kacem's work [48]	—	—	2.02 ± —	—
Conv-MeshAE [88]	1.939 ± 0.318	1.924	2.00 ± 0.52	1.90
Ours	2.429 ± 0.667	2.283	3.11 ± 0.92	2.96
Ours+ne-gt	**1.894 ± 0.430**	**1.764**	**1.93** ± 0.61	**1.82**

In these tables, the label "Ours" refers to our method that *does not access the neutral ground truths in end-to-end training.* Such a feature is particularly relevant for real-world scenarios in which corresponding identity shapes might not be available.

On the other hand, the "Ours+ne-gt" label corresponds to our method that uses neutral faces as ground truths for training, aligning with all other methods that we compare with.

Based on the results presented in Table 3.1 and Table 3.2, we observe the robustness of our method in 3D face reconstruction on both CoMA and BU3DFE datasets. Notably, our method achieves better reconstruction results on the CoMA dataset when neutral ground truths are used ("Ours+ne-gt"), showing the effectiveness of our method in scenarios with available ground truths. Furthermore, the strong performance when neutral ground truths are unavailable ("Ours") on the BU3DFE dataset verifies the flexibility of our method. It proves that our pipeline can effectively predict 3D full face reconstruction in various contexts.

Due to the recent release of FaceScape and the relatively limited number method proposed for both disentanglement and reconstruction on it, our comparative experiment is confined to Kacem's work and Conv-MeshAE. These two approaches focus on predicting the corresponding neutral faces rather than the original expressive faces. Thus, we report only the 3D expressive face reconstruction results of our method in Table 3.3.

Disentanglement Analysis The quantitative results E_{dis} for 3D face identity expression disentanglement on the CoMA and BU3DFE datasets, in comparison with FLAME, Jiang's work and DI-MeshEncoder, are also presented in Table 3.1 and Table 3.2, respectively. The E_{dis} results on the FaceScape datset, compared with Conv-MeshAE, are reported in Table 3.3 as well.

Results in Tables 3.1, 3.2, 3.3 and 3.4 demonstrate that our method is able to deliver strong performance in 3D face identity and expression disentanglement, especially when neutral ground truths are used ("Ours+ne-gt"). Even in the scenarios lacking neutral ground truths, our method still shows competitive results.

As we observed, the results for E_{dis} are constantly better when neutral ground truths are used than when they are not, across all three datasets. This improvement can be ascribed to the fact that the process with neutral ground truth acts as the strong supervision. Combined with L2 constraints, they ensure that the neutralised expressive face shapes closely align with the ground truths, resulting in more stable and accurate predicted identity shapes.

We observe a notably lower performance in E_{dis} on the CoMA dataset. This is mainly due to the specific split scheme employed in the CoMA dataset. In this scheme, given that the training set consists of 18,422 meshes with merely 12 different

expressions, pairs for the identity discriminator with the same identities may have similar expression deformations. This overlap causes the inherent nature of these pairs to include not only identity but also slight expressions, leading to higher level of disentanglement error E_{dis} and instability. Meanwhile, the reconstructed identity loss enforces that the predicted identity shapes closely align to ground truth neutral in each iteration. These reasons explain the significant difference in our results between using and without using neutral ground truths in the CoMA dataset.

From Table 3.2 and Table 3.3, we discover improvements in reconstruction error E_{rec} in scenarios where neutral ground truths are unavailable. Conversely, we note improvements in disentanglement error E_{dis} when neutral ground truths exist in the dataset. The reason is that our use of a GAN network introduces a trade-off between reconstruction and disentanglement performance. Thus, it is unsurprising that disentanglement performance is compromised when neutral ground truths are not available in our method.

We compare with Conv-MeshAE on the BU3DFE dataset and with Kacem's work and Conv-MeshAE on the FaceScape dataset, employing another evaluation metric, AVD_{neu}, as listed in Table 3.4.

3.2.5.2 Qualitative Results

In Figure 3.6, we present some representative unseen identity results for 3D face reconstructions and disentanglement, using neutral ground truths on FaceScape. These results are divided into three groups to show the best, average and worst performance of our work, as evaluated by the quantitative metrics. In each group, the first row displays the ground truth of the full face, neutral, and expression. The second row presents corresponding prediction results. Error heat maps are shown in the third row.

Figure 3.7 depicts the results of 3D face disentanglement and reconstruction on the FaceScape dataset without the use of ground truth neutrals. A comparison between Figure 3.6 and Figure 3.7 shows that, when neutral ground truths can be obtained, even in the worst case, the identity and expression on the mean face are effectively decoupled. The reconstruction error mainly occurs in the neck region. In scenarios where neutral ground truths do not exist, the performance of the overall full face reconstruction is desirable due to the L2 constraint. In the best and average case, the identity and expression are successfully disentangled, but with minor unsmoothness observed on the face and particularly around the mouth region. In

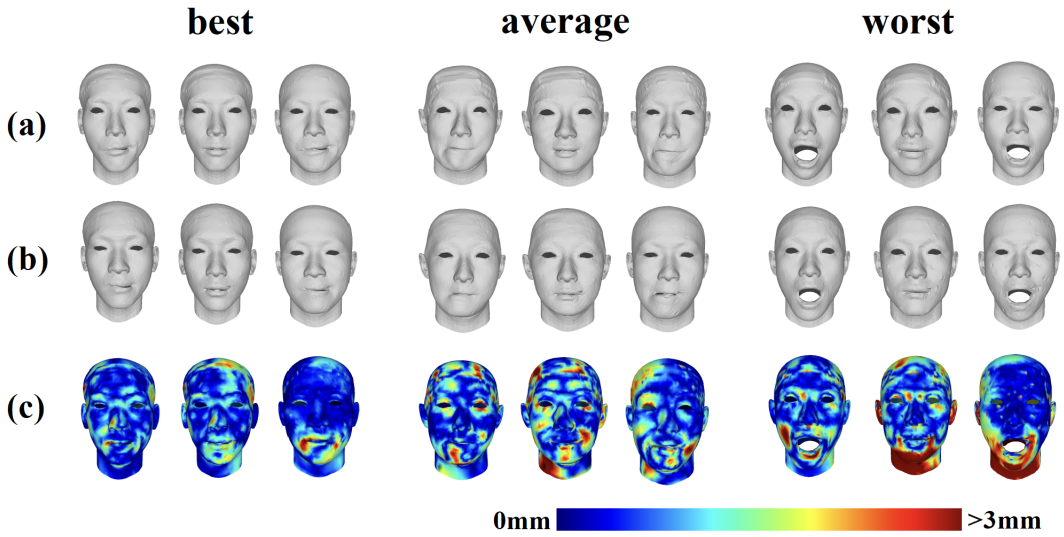

Figure 3.6: Results of unseen 3D face identity-expression disentanglement on the FaceScape dataset when neutral ground truths are available - groups from left to right: the best, the average, and the worst. (a) illustrates the face ground truths (input); (b) represents the predicted face shapes; and (c) depicts the error map. For each group, the first column displays full faces, the second column denotes identity faces, and the third column shows expressions on the mean face.

the worst case, slight expressions can be observed in the mouth region (left-shift mouth), with a comparatively higher reconstruction error.

Similarly, as shown in Figure 3.8 and Figure 3.9, the predicted neutral faces on the CoMA dataset present extremely low error when neutral ground truths are used, although the expression predictions perform slightly worse than identity parts. This is mainly due to the fact that there are a large number of face scans sharing the same identity in this dataset. L2 constrains occur numerous times during the training process. The constraints imposed on the expression branch are relatively fewer than those on the identity, which leads to a slightly higher reconstruction error for expressions.

The unseen identity results of the BU3DFE dataset are illustrated in Figure 3.10 and Figure 3.11. In Figure 3.10, our method demonstrates a strong performance, particularly in both the best and average groups. The process of identity expression disentanglement works effectively, and the primary reason for the higher reconstruction error is the predicted full face with a more exaggerated expression that the

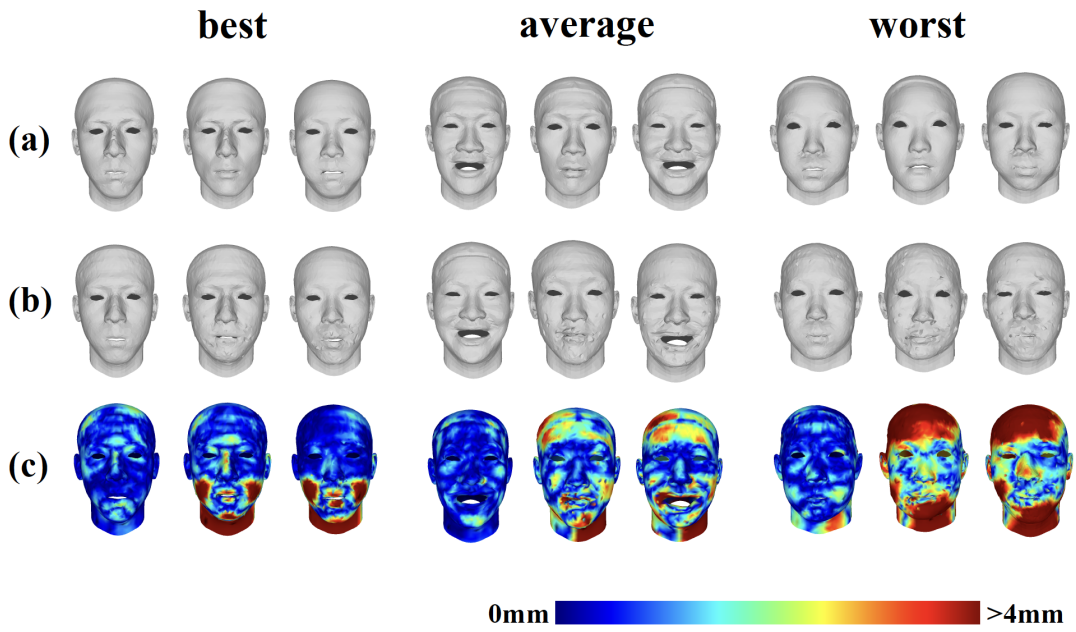

Figure 3.7: Results of unseen 3D face identity-expression disentanglement on the FaceScape dataset when neutral ground truths are unavailable - groups from left to right: the best, the average, and the worst. (a) illustrates the face ground truths (input); (b) represents the predicted face shapes; and (c) depicts the error map. For each group, the first column displays full faces, the second column denotes identity faces, and the third column shows expressions on the mean face.

ground truth, which also results in a similar condition for the predicted expression. From the analysis of the best and average cases depicted in Figure 3.11, it is evident that our method is able to decompose facial identity and expression, even in the unavailability of neutral ground truths. While the predicted identity occasionally shows a slight open mouth, this may be attributed to the center point of all expressions in the space converging at this 'neutral' expression. It is especially notable in the worst-case scenario without neutral ground truths that the predicted identity shapes show slight expressions, an open mouth, especially when expression shapes are exaggerated.

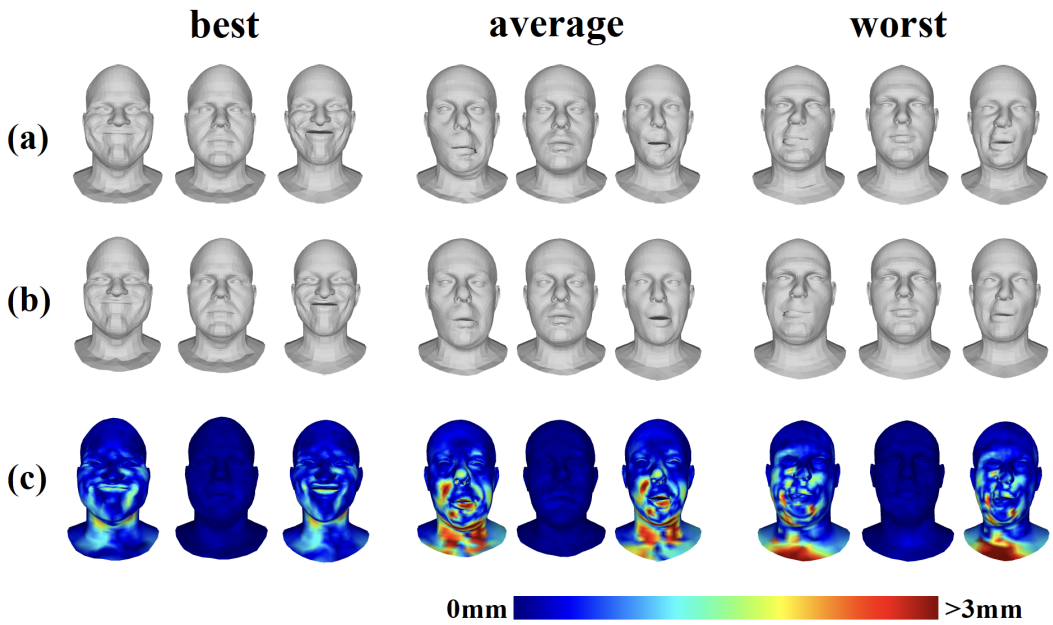

Figure 3.8: Results of unseen 3D face identity-expression disentanglement on the CoMA dataset when neutral ground truths are available - groups from left to right: the best, the average, and the worst. (a) illustrates the face ground truths (input); (b) represents the predicted face shapes; and (c) depicts the error map. For each group, the first column displays full faces, the second column denotes identity faces, and the third column shows expressions on the mean face.

3.2.6 Ablation Studies

3.2.6.1 The Identity Discriminator

In our architecture, we introduce an identity discriminator designed to separate identity and expression, even if corresponding neutral ground truths are not available. To evaluate the effectiveness of our discriminator, we present disentanglement results E_{dis}, average vertex distance results (AVD_{neu}) and reconstruction results E_{rec} on the CoMA, BU3DFE and FaceScape datasets in Table 3.5, Table 3.6 and Table 3.7, respectively. Qualitative results illustrating how our identity discriminator performs in scenarios without neutral ground truths, are shown in Figure 3.12, Figure 3.13 and Figure 3.14.

From these tables, we can observe that our discriminator greatly improves disentanglement performance, particularly in scenarios where neutral ground truths are not available. For example, as shown in Table 3.5 and Table 3.6, our results

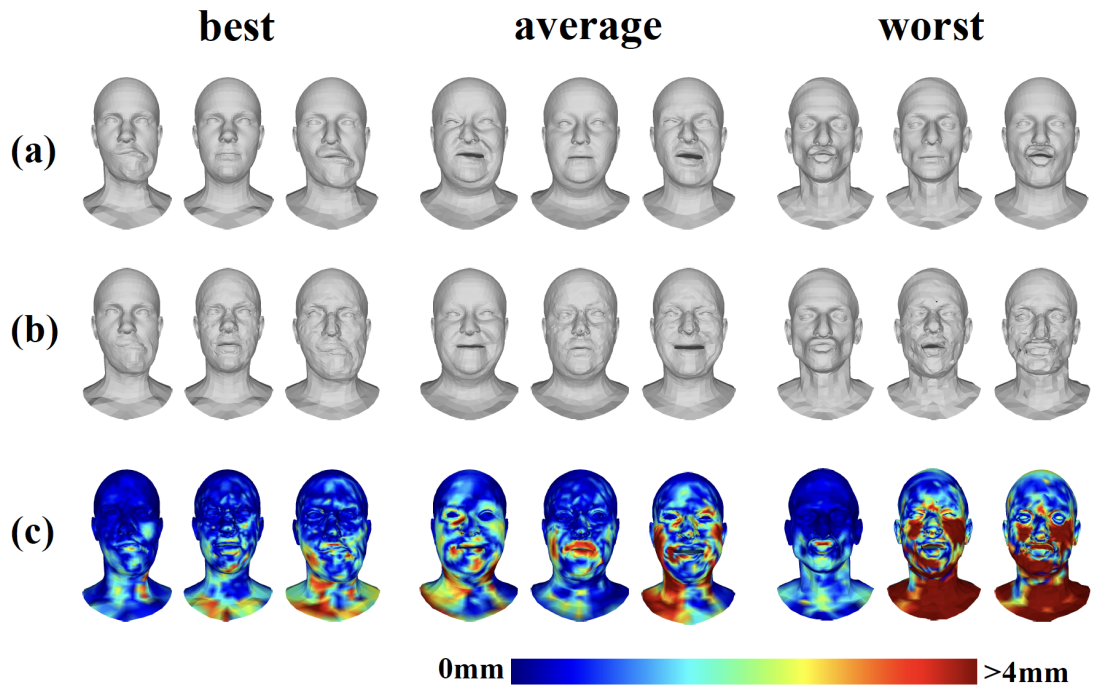

Figure 3.9: Results of unseen 3D face identity-expression disentanglement on the CoMA dataset when neutral ground truths are unavailable - groups from left to right: the best, the average, and the worst. (a) illustrates the face ground truths (input); (b) represents the predicted face shapes; and (c) depicts the error map. For each group, the first column displays full faces, the second column denotes identity faces, and the third column shows expressions on the mean face.

with '+id-dis' significantly outperform those with '−id-dis' on the FaceScape dataset, especially when ground truth neutrals are not available, with around a reduction of 75% in AVD_{neu} (decreasing from 12.020 to 3.112) and a reduction of 57% reduction of E_{dis} (decreasing from 1.791 to 0.765). Similar improvements are evident on the CoMA dataset, where AVD_{neu} decreases from 2.775 to 1.528, and E_{dis} reduces from 1.439 to 0.176.

The improvements of disentanglement with neutral ground truths are not as significant as those of the case with unavailable ground truths. Note that using ground truth neutrals is a strong supervised training process, whereas, VAE and discriminator learn identity representations adversarially in an weakly-supervised process. In few cases presented in the tables above, when identity ground truths are available, we observe, from a few cases in the tables above in which identity ground

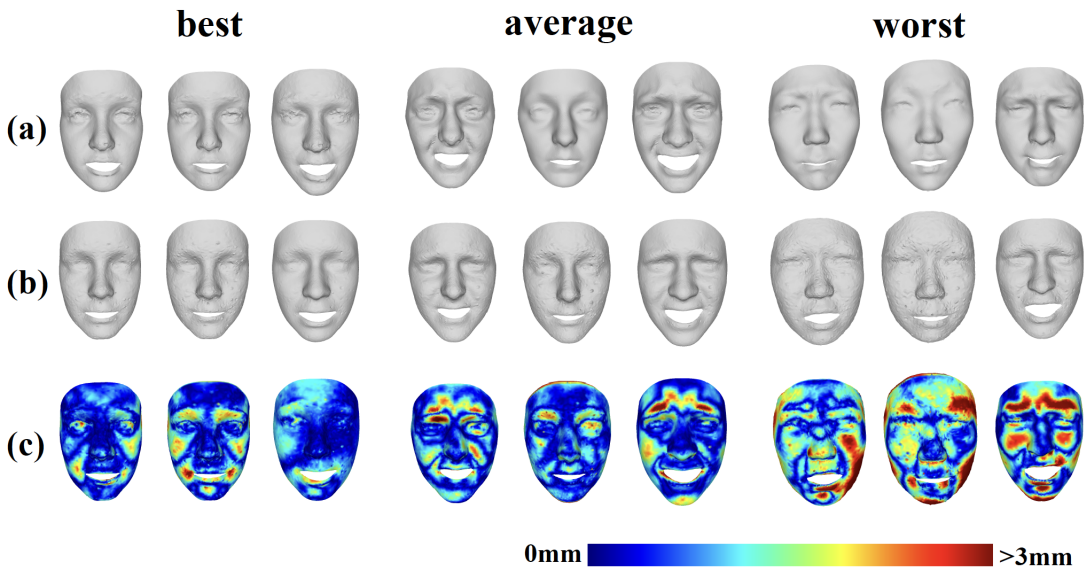

Figure 3.10: Results of unseen 3D face identity-expression disentanglement on the BU3DFE dataset when neutral ground truths are available - groups from left to right: the best, the average, and the worst. (a) illustrates the face ground truths (input); (b) represents the predicted face shapes; and (c) depicts the error map. For each group, the first column displays full faces, the second column denotes identity faces, and the third column shows expressions on the mean face.

Table 3.5: Comparison results of E_{dis} on the CoMA, BU3DFE and FaceScape datasets. The term '−ne-gt' represents our methods without neutral ground truths, and '−id-dis' indicates our method without the identity discriminator. Conversely, the '+ne-gt' and '+id-dis' represent our method with neutral ground truths and the identity discriminator, respectively. Results are highlighted for comparisons between using (+id-dis) or not using (−id-dis) the identity discriminator. All errors are measured in millimeters.

Method	Dataset	−id-dis		+id-dis	
		mean	median	mean	median
−ne-gt	CoMA	1.439	1.443	**0.176**	**0.180**
	BU3DFE	1.211	1.144	**0.443**	**0.439**
	FaceScape	1.791	1.795	**0.765**	**0.758**
+ne-gt	CoMA	0.016	0.015	**0.014**	**0.013**
	BU3DFE	**0.345**	**0.337**	0.348	0.339
	FaceScape	0.582	0.563	**0.569**	**0.551**

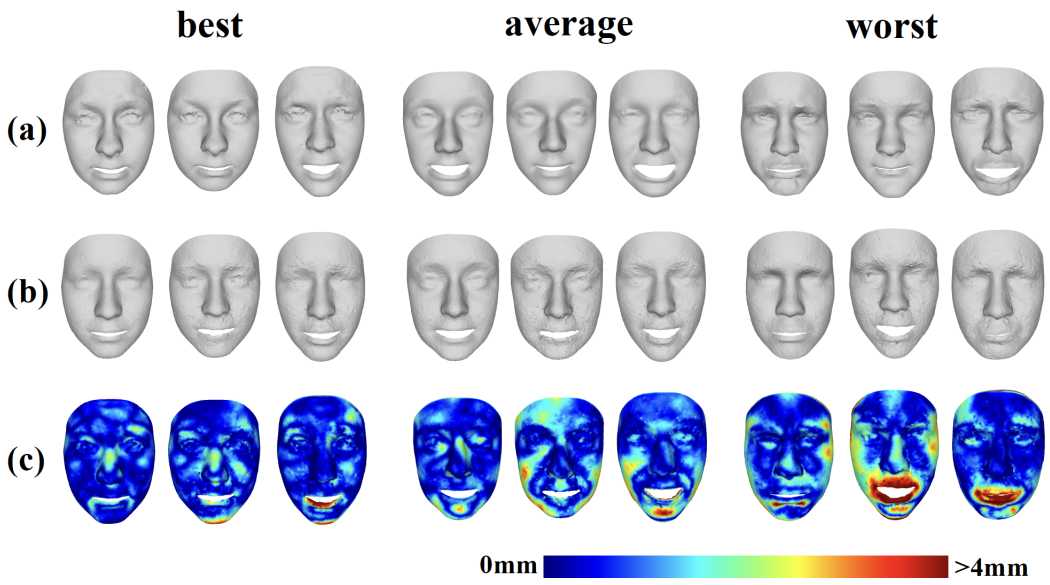

Figure 3.11: Results of unseen 3D face identity-expression disentanglement on the BU3DFE dataset when neutral ground truths are unavailable - groups from left to right: the best, the average, and the worst. (a) illustrates the face ground truths (input); (b) represents the predicted face shapes; and (c) depicts the error map. For each group, the first column displays full faces, the second column denotes identity faces, and the third column shows expressions on the mean face.

Table 3.6: Comparison results of AVD_{neu} on the CoMA, BU3DFE and FaceScape datasets. The term '−ne-gt' represents our methods without neutral ground truths, and '−id-dis' indicates our method without the identity discriminator. Conversely, the '+ne-gt' and '+id-dis' represent our method with neutral ground truths and the identity discriminator, respectively. Results are highlighted for comparisons between using (+id-dis) or not using (−id-dis) the identity discriminator. All errors are measured in millimeters.

Method	Dataset	−id-dis		+id-dis	
		mean ± std	median	mean ± std	median
−ne-gt	CoMA	2.775 ± 0.948	2.509	**1.528 ± 0.675**	**1.268**
	BU3DFE	4.108 ± 1.246	3.958	**2.429 ± 0.667**	**2.283**
	FaceScape	12.020 ± 0.514	11.876	**3.112 ± 0.916**	**2.957**
+ne-gt	CoMA	0.071 ± 0.012	0.070	**0.065 ± 0.012**	**0.063**
	BU3DFE	**1.885 ± 0.459**	**1.733**	1.894 ± 0.430	1.764
	FaceScape	1.927 ± 0.617	1.821	**1.927 ± 0.610**	**1.815**

Table 3.7: Comparison results of E_{rec} on the CoMA, BU3DFE and FaceScape datasets. The term '−ne-gt' represents our methods without neutral ground truths, and '−id-dis' indicates our method without the identity discriminator. Conversely, the '+ne-gt' and '+id-dis' represent our method with neutral ground truths and the identity discriminator, respectively. Results are highlighted for comparisons between using (+id-dis) or not using (−id-dis) the identity discriminator. All errors are measured in millimeters.

Method	Dataset	−id-dis		+id-dis	
		mean ± std	median	mean ± std	median
−ne-gt	CoMA	**0.686 ± 0.190**	0.674	0.783 ± 0.225	**0.651**
	BU3DFE	1.469 ± 0.405	1.359	**1.421 ± 0.412**	**1.306**
	FaceScape	1.187 ± 0.300	1.138	**1.157 ± 0.286**	**1.109**
+ne-gt	CoMA	0.669 ± 0.213	0.647	**0.651 ± 0.208**	**0.625**
	BU3DFE	1.509 ± 0.427	**1.382**	**1.500 ± 0.423**	1.404
	FaceScape	1.393 ± 0.379	1.330	**1.370 ± 0.369**	**1.307**

truth are available, a slight decrease in reconstruction error when using '+id-dis'. Thus, when ground truth neutrals (strong supervised process) works, the effectiveness of weakly-supervised process is not obvious. In addition, some reconstruction results are compromised to a small extent because of the use of adversarial learning.

The same effectiveness is qualitatively depicted in Figures 3.12, 3.13 and 3.14. A great performance of not only in face identity-expression disentanglement but also face reconstruction, has been achieved if the identity discriminator, as in (c) group of each figure being compared, is used.

3.2.6.2 The Decimation Algorithm Applied to the FaceScape dataset

As outlined in Section 3.2.2, to optimise model training efficiency on the FaceScape dataset, we apply a decimation algorithm to reduce the number of vertices in the input meshes from 26,317 to 4,547 and the number of faces from 52,261 to 8,999. The Python library [76] is used for mesh simplification, leveraging a quadric-based edge-collapse strategy [35]. Different weighting schemes are applied to address issues related to aspect ratios and degenerate quadrics. To evaluate the impact of this decimation process, we also conduct experiments on the original FaceScape dataset, where each face mesh contains 26,317 vertices and 52,261 faces. Figure 3.15 presents a comparison of the reconstructed face shapes derived from both the original and simplified face scans, without the corresponding identity ground truths and without

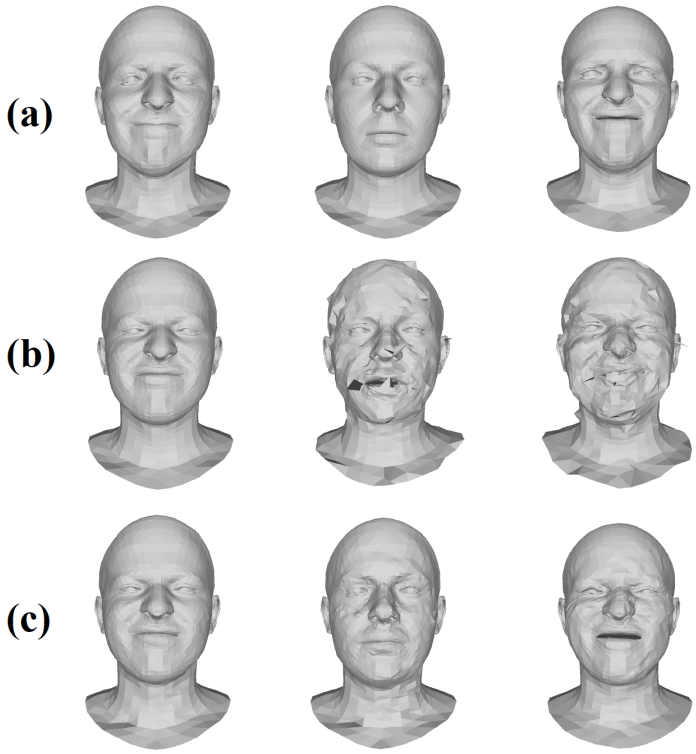

Figure 3.12: Comparisons between using and not-using the identity discriminator when neutral ground truths are unknown on the CoMA dataset. (a) illustrates the full face ground truths (input); (b) represents the predicted face shapes without using our identity discriminator; and (c) depicts the predicted face shapes using the identity discriminator. In each row, the first face displays the full face, the second one denotes the identity face, and the third one shows the expression on the mean face.

using the identity discriminator. In contrast, Figure 3.16 shows a comparison of the reconstructed face shapes derived from both the original and simplified face scans, with the corresponding identity ground truths and using the identity discriminator.

As illustrated in Figure 3.15, the predicted identity and expression exhibit significant noise when simplified face meshes are used as input, as also observed in Figure 3.14. However, this noise is less evident when the original face meshes are used. This is attributed to the subsampling process, where the decimated face meshes lack dense correspondence between points, thereby reducing the fidelity of the reconstruction. Moreover, in the absence of identity ground truths, *i.e.*, when the L2 constraint between the identity ground truths and predicted identities is not enforced, the predicted shapes are insufficiently supervised, leading to topological

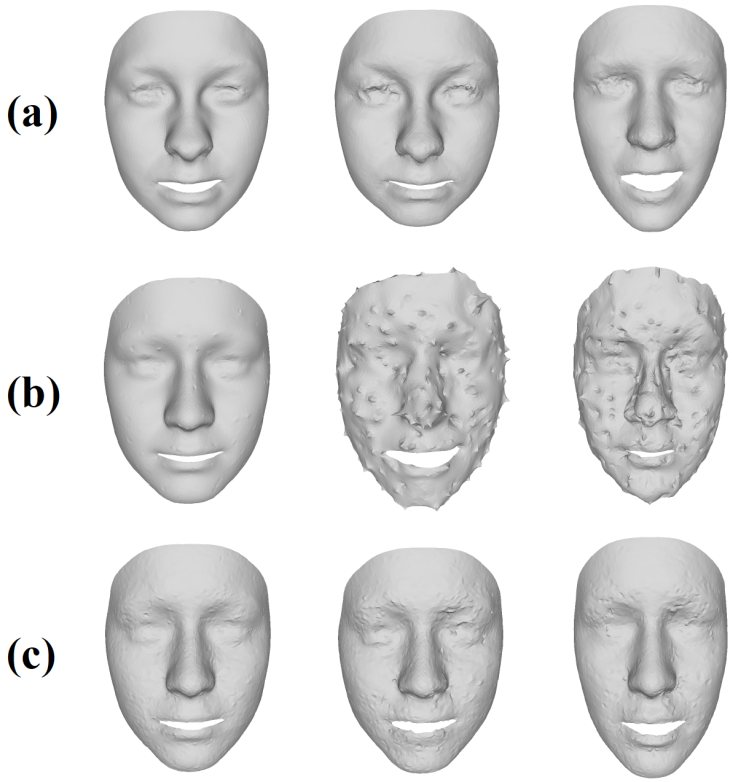

Figure 3.13: Comparisons between using and not-using the identity discriminator when neutral ground truths are unknown on the BU3DFE dataset. (a) illustrates the full face ground truths (input); (b) represents the predicted face shapes without using our identity discriminator; and (c) depicts the predicted face shapes using the identity discriminator. In each row, the first face displays the full face, the second one denotes the identity face, and the third one shows the expression on the mean face.

inconsistencies.

While the predicted faces in Figure 3.16 perform well with both simplified and original face shapes as the input, the original predicted face meshes exhibit more noise. This increased noise can be explained by our decoder networks, which employs an MLP to predict the final vertices from a low-dimensional latent space. The simplified face meshes contain only 4,547 vertices, while the original faces have more than five times as many (26,317 vertices). This significant increase in the number of vertices presents a challenge for the decoder, which, with its comparatively limited

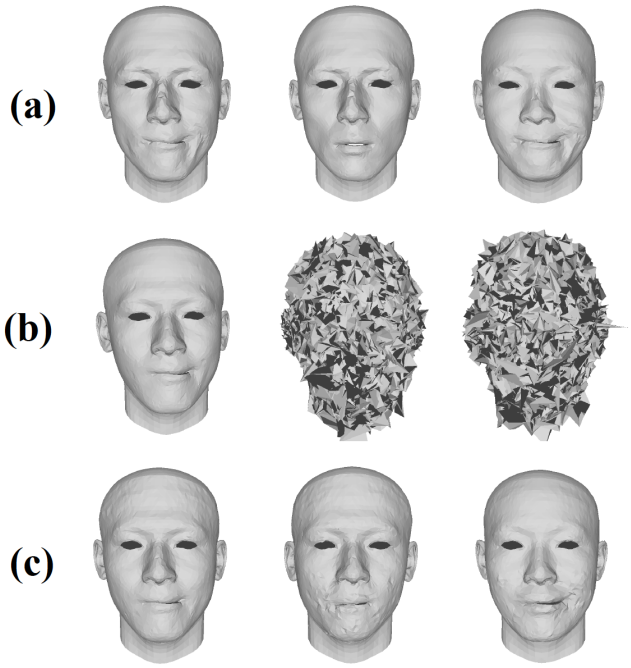

Figure 3.14: Comparisons between using and not-using the identity discriminator when neutral ground truths are unknown on the FaceScape dataset. (a) illustrates the full face ground truths (input); (b) represents the predicted face shapes without using our identity discriminator; and (c) depicts the predicted face shapes using the identity discriminator. In each row, the first face displays the full face, the second one denotes the identity face, and the third one shows the expression on the mean face.

number of hidden neurons, struggles to predict each vertex with accuracy.

3.2.7 APPLICATIONS

Table 3.8: The rank-1 accuracy results on the FaceScape and BU3DFE datasets

Method	Dataset	
	FaceScape (%)	BU3DFE (%)
Kacem's work [48]	99.88	—
Conv-MeshAE [88]	98.81	100.00
Ours	98.66	100.00
Ours+ne-gt	99.39	100.00

We apply our network for several applications, including expression transfer, identity and expression interpolation, and face recognition. Taking the CoMA and

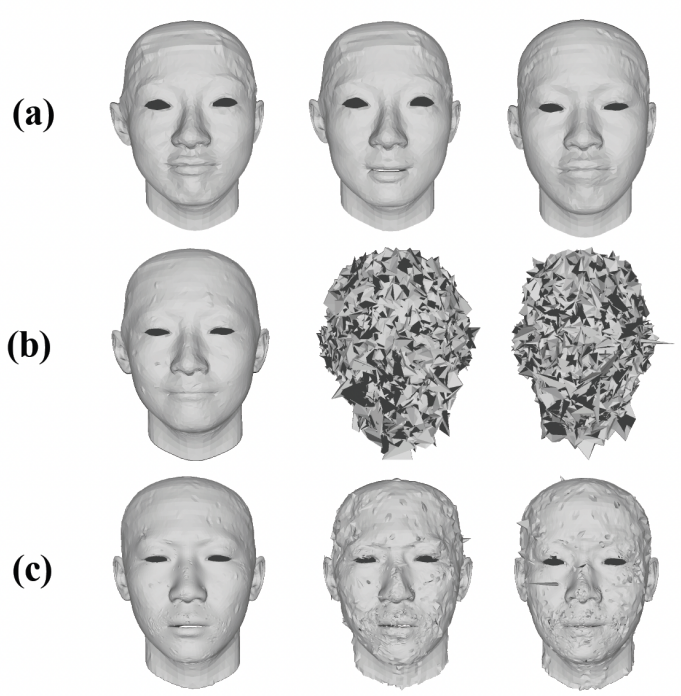

Figure 3.15: Comparisons between simplified and original input faces on the FaceScape dataset, where identity ground truths are unknown and the identity discriminator is not used. (a) illustrates the full face ground truths (input); (b) represents the predicted face shapes using the simplified faces as the input; and (c) depicts the predicted face shapes using the original faces as the input. In each row, the first face displays the full face, the second one denotes the identity face, and the third one shows the expression on the mean face.

FaceScape datasets as an example, we randomly select two subjects with different expressions from the test set and transfer their expression latent representations, as shown in Figure 3.19.

We also display the disentangled identity and expression interpolations in Figure 3.17 and Figure 3.18. For these interpolations, two sets of latent representations corresponding to different identities and expressions are learnt by our encoder. We then perform interpolation with a step length that increases by 25% of the preset length on each iteration.

For the face recognition application, we implement it on the FaceScape and BU3DFE datasets, as the CoMA dataset contains only 12 individuals, which limits its utility for this application. The face shapes in the test sets of both BU3DFE and FaceScape are unseen in the training data. This application is executed after

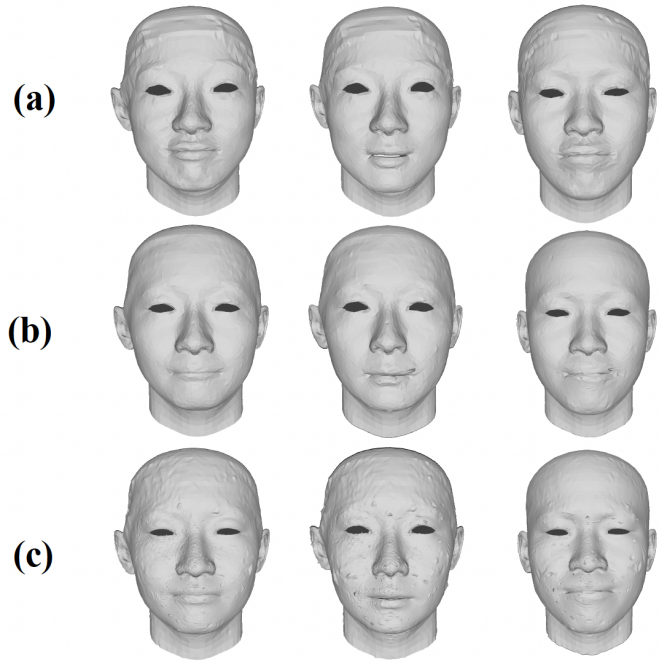

Figure 3.16: Comparisons between simplified and original input faces on the FaceScape dataset, where identity ground truths are known and the identity discriminator is used. (a) illustrates the full face ground truths (input); (b) represents the predicted face shapes using the simplified faces as the input; and (c) depicts the predicted face shapes using the original faces as the input. In each row, the first face displays the full face, the second one denotes the identity face, and the third one shows the expression on the mean face.

disentangling identity and expression from full face shapes, and then the generated identity latent representations are used for face recognition.

We follow the practice of [48] to employ the cosine similarity measure to evaluate face recognition performance. The neutral faces from the test set act as references and expressive face shapes serve as probes. We use rank-1 accuracy for evaluation, with results detailed in Table 3.8.

We report accuracy results from Kacem's work [48]. In our work, we set the dimensions of latent vectors to be those in Conv-MeshAE, specifically 80 (40 for identities and 40 for expressions) for BU3DFE and 128 (again, split equally) for FaceScape. The evaluation in [48] adopts identity features equal in size to the number of training subjects. The identity features are much larger than our configuration, potentially enabling more detail information storage in their latent space and leading to their higher reported accuracy. The high accuracy in BU3DFE is caused by the

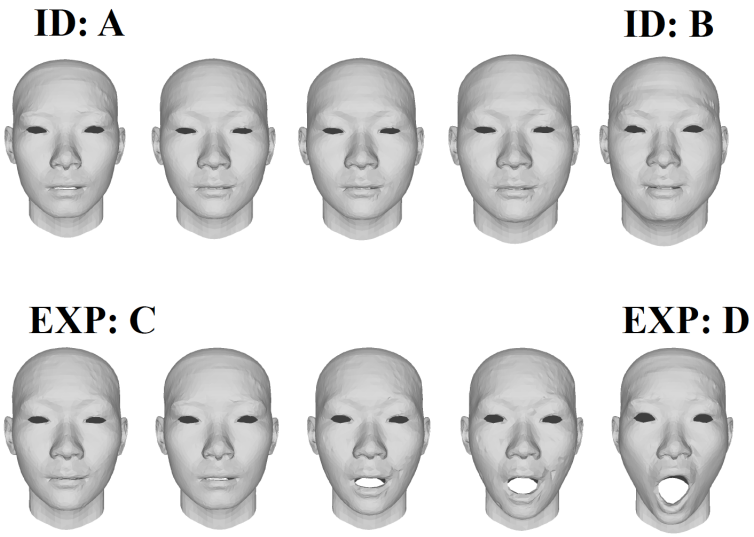

Figure 3.17: Interpolations of identity and expression latent representations on the FaceScape dataset. The interpolations are from the identities of subject A to subject B (with same expressions) and from expression C to expression D (with same identities) individually.

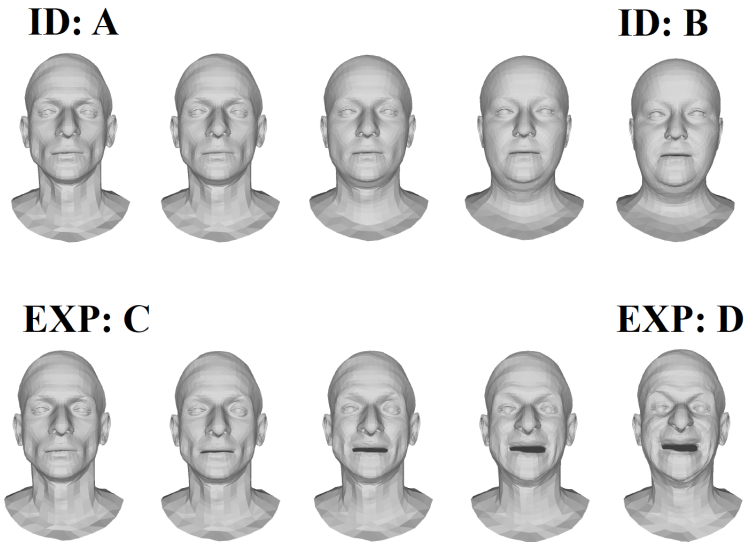

Figure 3.18: Interpolations of identity and expression latent representations on the CoMA dataset. The interpolations are from the identities of subject A to subject B (with same expressions) and from expression C to expression D (with same identities) individually.

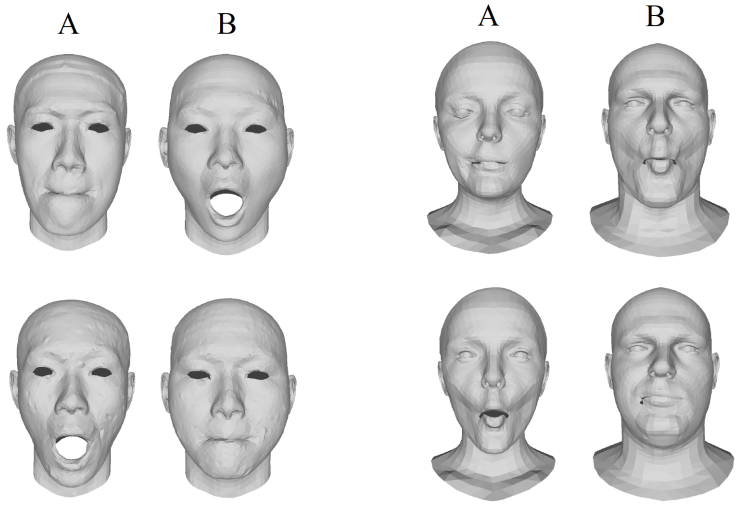

Figure 3.19: Expression transfer using our disentanglement network on the FaceScape (a) and CoMA (b) datasets. There are two subjects, A and B, for each dataset. The first row displays the original input faces of A and B, and the second row shows the faces with tranferred expressions, *i.e.*, A's subject with B's expression and B's subject with A's expression.

small number of identities (10) in its test set. Even if corresponding neutrals are not given, our method still achieves desired face recognition results.

3.3 SUMMARY

We proposed a method employing a VAE and a discriminator for disentangling 3D face identities and expressions. To learn identity representations, we use pairs of 3D faces to train an identity discriminator, which is forced to capture identity features of the same subjects only. This particularly improves the performance in the situations where neutral expressions are not available. Additionally, the joint end-to-end learning of the encoder-decoder network and the identity discriminator helps reconstruct 3D faces. We performed evaluations on CoMA, FaceScape and BU3DFE, showing the high effectiveness of our network for 3D face reconstruction and identity/expression disentanglement.

In summary, contributions are:

- An adversarial approach to facial identity and expression disentanglement that exploits a PointNet-based VAE and discriminator.
- To the best of our knowledge, we are the first to address the scenario of unknown ground truth neutrals, leveraging the invariance of identities from same individuals and employing the apathy 'expression' as the centre of expression space in order to train an end-to-end model, *i.e.*, the identity discriminator and VAE are trained simultaneously in an unsupervised manner.
- We compare the results of using and without using neutral ground truths, and observe the desirable performance of disentanglement on applications including face recognition, expression transfer and expression interpolation.
- Evaluation on publicly-available datasets demonstrates state-of-the-art results with the option of operating in a more versatile application setting of no known neutral ground truths.

— 4 —

Parts-Based Implicit 3D Face Modelling

This chapter extends identity/expression disentanglement by additionally disentangling facial parts in a new parts-based 3D face modelling approach, within which we employ implicit shape representations. Thus we address the open challenges of independently controlling different facial parts and providing the ability to learn explainable parts-based latent shape embeddings for implicit surfaces. Additionally, we present a new scheme for facial feature swapping that has allowed significant data augmentation for network training. We are able to demonstrate state-of-the-art reconstruction results on the FaceScape dataset, with particularly good performance on the facial parts. Finally, we extend evaluations by utilising the Headspace dataset of full head shapes.

Existing 3D facial generative models that employ a VAE are able to learn latent embeddings for each face shape. Some recent works have aimed to disentangle the latent embeddings on expressive facial datasets, which makes the latent representations more explainable. Among others, learning that decouples identity and expression latent representations has achieved remarkable results [43, 46, 94]. However, the learning of both controllable and disentangled latent embeddings for distinct facial parts is still a challenging task and remains crucial for many applications where local controllability is important. Examples include 3D photofit, craniofacial surgery (*e.g.* minor adjustments of the nose) or, in gaming, where small, localised facial adjustments of game characters is required.

Historically, most 3D face models were based on explicit representations such as point clouds, voxel grids and meshes. However, more recently, implicit representations that use signed distance functions (SDFs), unsigned distance fields or occupancy functions have become the preferred approach [79, 72, 18, 66, 20, 22, 118, 23]. The

benefit is that such representations are compact and have the flexibility to represent complex shapes that are rich in detail, without being tied to a particular mesh resolution and topology. Here, we focus on implicit 3D face modelling, where a SDF and shape deformation fields are employed to represent face shapes, with the goal of disentangling the encoding of specific and distinct facial parts.

We propose a method for 3D face modelling that learns a continuous parts-based deformation field. This field maps the various semantic parts of a subject's face to a template. By swapping affine-mapped facial features among different individuals from predefined regions we achieve significant parts-based training data augmentation. Moreover, by sequentially morphing the surface points of these parts, we learn corresponding latent representations, shape deformation fields, and the SDF of a template shape. This yields improved shape controllability and better interpretability of the face latent space, while retaining all of the known advantages of implicit surface modelling.

Evaluations verify the effectiveness of both facial expression and parts disentanglement, independent control of those facial parts, as well as state-of-the art facial parts reconstruction, when evaluated on FaceScape and Headspace datasets.

The structure of this chapter is as follows: in Section 4.1, we discuss the problem that we tackle and detail the mechanism of the parts-based implicit network. Section 4.2 evaluates the network performance in terms of expressions, parts and whole shape, using both qualitative and quantitative metrics on our generated 3D face shapes and compares them with other implicit representation methods. Finally, we conclude and highlight our contributions in Section 4.3.

4.1 METHODOLOGY

In this section, we describe the problem and provide a comprehensive introduction to our method for learning parts-based facial representations in an implicit manner. A key component of our method involves swapping facial features across subject pairs, which enables significant data augmentation via affine transform of facial parts in conjunction with Laplacian-regulated mesh morphing.

Our architecture, shown in Figure 4.1, is designed as a generative model for 3D face expression and part deformations. Within this framework, we adopt the 'mini-nets' structure, proposed by Zheng *et al.* [117], for cascaded 3D shape deformations.

In Section 4.1.1, we introduce our problem setting, followed by preliminary knowledge of a base network used in our architecture in Section 4.1.2. We introduce a detailed expression and parts-based deformation network that we propose in Section 4.1.3. Section 4.1.4 details the data augmentation process, which includes swapping facial parts among face shapes. Finally, the loss function components applied in our network are introduced in Section 4.1.5.

4.1.1 Problem Setting

We utilise an implicit function, specifically a SDF, as a template shape representation, due to its compactness and resolution-free expressivity, for modelling the fine details of human faces. Given a 3D query point, $\mathbf{p} \in \mathbb{R}^3$, and a set of latent variables that represent (global) facial expression, along with (neutral) facial part shapes, we aim to learn a conditional SDF:

$$s = \Phi(\mathbf{p}|\mathbf{z}_{exp}, \mathbf{z}_{nose}, \mathbf{z}_{eyes}, \mathbf{z}_{mouth}, \mathbf{z}_{rem}), \tag{4.1}$$

where $s \in \mathbb{R}$ is the signed distance. Facial features, *i.e.*, exp, nose, eyes, mouth and the remaining face/head part (denoted by *'rem'*), are represented by corresponding latent vectors denoted as $\mathbf{z}_{exp} \in \mathbb{R}^e$, $\mathbf{z}_{nose} \in \mathbb{R}^d$, $\mathbf{z}_{eyes} \in \mathbb{R}^d$, $\mathbf{z}_{mouth} \in \mathbb{R}^d$ and $\mathbf{z}_{rem} \in \mathbb{R}^{d'}$, respectively. Then the surface, Ω_0, of a facial shape is represented by the zero-level set of the SDF:

$$\Omega_0\left(\Phi\right) = \{\mathbf{p} \in \mathbb{R}^3 \mid \Phi\left(\mathbf{p}|\mathbf{z}_{exp}, \ldots, \mathbf{z}_{rem}\right) = 0\}. \tag{4.2}$$

To learn independent latent vectors for expressions and facial parts, as well as a conditional SDF, we propose a sequential deformation neural network that leverages augmented face shape data for training, by using affine transforms and Laplace-regulated mesh deformations to swap facial parts between different subjects.

4.1.2 SIREN-based Architecture

The Sinusoidal Representation (SIREN) approach [91], introduced by Sitzmann *et al.*, employs MLPs with the sine as a periodic activation function for implicit neural representations and their derivatives. It is able to fit highly-detailed shapes based on SDFs by enforcing the Eikonal constraints for points and supervising the gradients of sampled oriented points to remain consistent with surface normals.

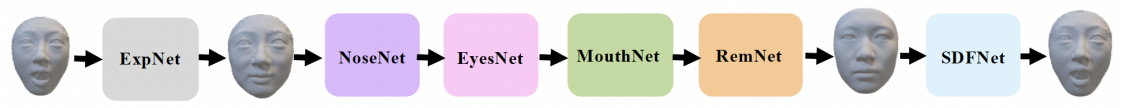

Figure 4.1: The overall architecture of our model. This end-to-end deformation network is composed of six modules, namely ExpNet, NoseNet, EyesNet, MouthNet, RemNet and SDFNet. The input to the overall network is an expressive face. After ExpNet, a neutral face is obtained, and the parts-based deformations are processed sequentially on the neutral face.

Inspired by [91], we employ similar loss functions for our SDF network as:

$$\mathcal{L}_{SDF} = \lambda_{Eik} \sum_{\mathbf{p}\in\Omega} |\|\nabla\Phi(\mathbf{p})\| - 1| + \lambda_{normal} \sum_{\mathbf{p}\in\Omega} \left(1 - \langle \nabla\Phi\left(\mathbf{p}\right), \mathbf{n}\left(\mathbf{p}\right)\rangle\right), \tag{4.3}$$

where $\|\cdot\|$ ($\|\cdot\|_2$) represents the L2 (Euclidean) norm, and λ_{Eik} and λ_{normal} are weights for these two terms. $\nabla\Phi(\mathbf{p})$ denotes the gradients at points, and $\mathbf{n}\left(\mathbf{p}\right)$ represents the surface normal. Points $\mathbf{p}$ are sampled from the entire domain Ω, and the first term in the loss function is designed to find a surface where the gradients of $\mathbf{p}$ are constrained to have a unit Euclidean norm. The second term is used to align the gradients $\nabla\Phi(\mathbf{p})$ with the surface normal $\mathbf{n}\left(\mathbf{p}\right)$ at all points $\mathbf{p}$. A hyper-network is also proposed to predict the parameters of SIREN, which can be modelled in a latent space. We adopt this design in our model to effectively map the parts-based latent representations of each facial region onto the weights of our deformation network.

4.1.3 PARTS-BASED DEFORMATION NETWORKS

We propose to learn separate representations for different regions of 3D faces. In other words, the 3D face surface that we aim to reconstruct is described by Equation (4.2). Our objective is to learn the latent representations of both the (global) facial expression and of pre-defined identity-based (neutral expression) facial parts, *i.e.*, the nose, eyes, mouth and the remaining region, in order to generate 3D faces with independent control over each part.

Considering the advantages such as resolution independence, the ability to handle complex topologies, and the superior capacity to capture highly detailed shapes, we choose to employ implicit representations, *i.e.*, SDFs, in our method for 3D modelling. Our network generally consists of two main functional parts: one for deformations (*i.e.*, expression to neutral and between various predefined facial regions up to the

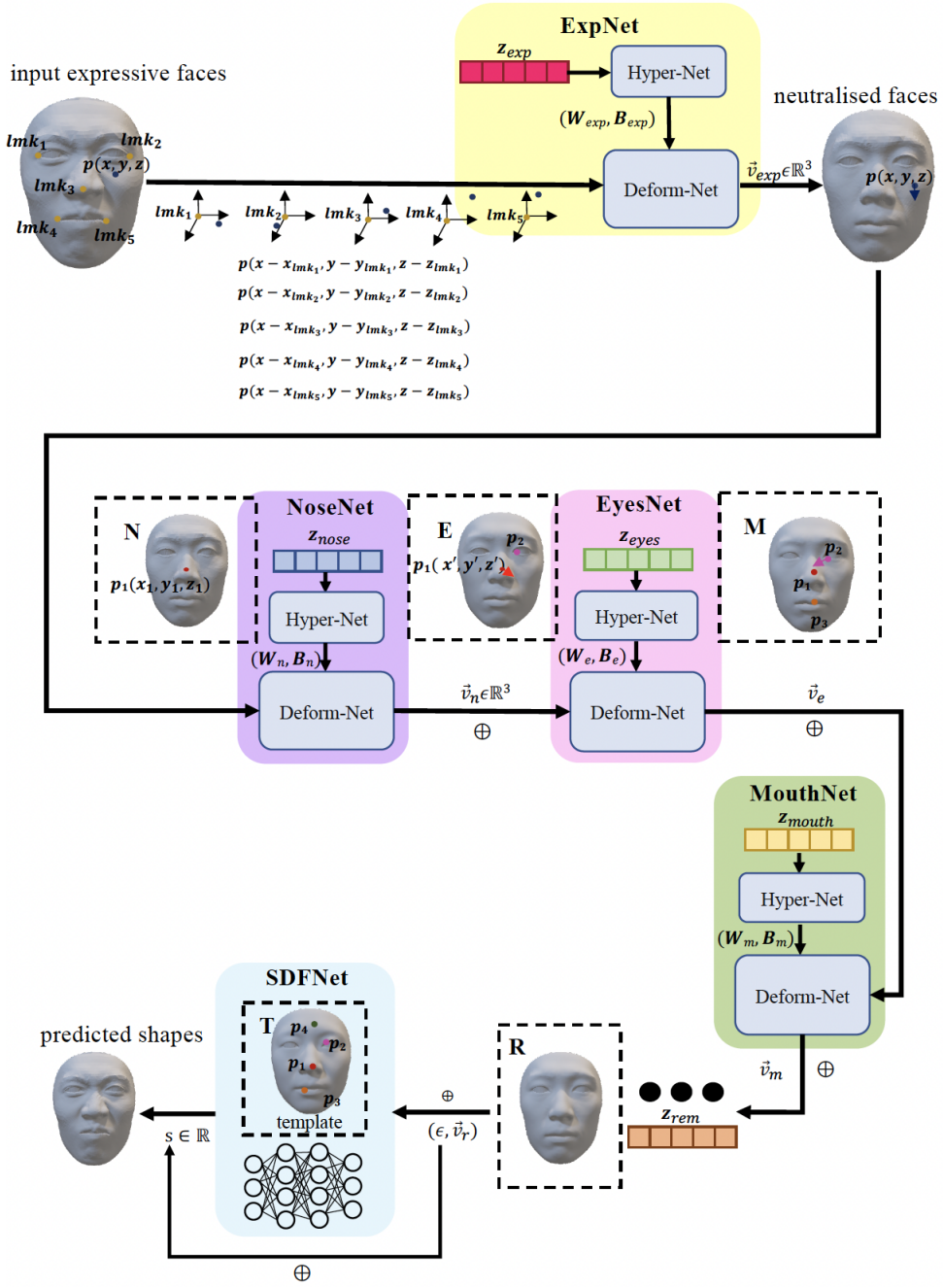

Figure 4.2: The detailed architecture of our model [42]. The end-to-end deformation network is composed of six modules, *i.e.*, ExpNet, NoseNet, EyesNet, MouthNet, RemNet - indicated by ellipsis (...) for compactness - and SDFNet. The five deformation modules share the same base network and deform the expressive/swapped neutral shape components back to their corresponding shape components on the template shape. The SDFNet employs a similar network and initialisations to SIREN [91] to learn the SDF of the template.

template shape), and the other dedicated to computing the SDF for the template face shape.

Deformation Networks. To achieve separate representations of facial expression and neutralised face regions, *i.e.*, the nose, eyes, mouth and the remainder, we propose a system of five deformation networks, arranged in a cascading manner. As illustrated in Figure 4.1, our method processes an input expressive face through the 'ExpNet', *i.e.*, the Expression Network, that is designed to remove expressions from the expressive facial shape, thereby learning a global expressive representation for the entire face. The neutralised face, obtained after the expression removal, is then fed into a sequential network consisting of four components, *i.e.*, 'NoseNet', 'EyesNet', 'MouthNet' and 'RemNet', each corresponding to a specific facial region. Within each of these part networks, we extract the unique features of the respective facial parts from different subjects, as well as their swapped parts, and ensure that the outputs are consistent with the corresponding parts on the template face, facilitating the learning of latent variables for each specific facial region. Overall, each component of our model is tailored to learn the latent representations and deformations for either global expression or the shape of a specific local face region relative to the corresponding local shape of the learnt template.

The input to the entire sequential network consists of expressive facial shapes, which are initially processed by the 'ExpNet' to learn global facial expression representations. As depicted in Figure 4.2, five predefined landmarks, *e.g.* the outer corners of the eyes, outer corners of the mouth, and the nose tip, are used to divide the global face/head into smaller, local area. This segmentation helps better represent facial details. Rather than using fixed global point coordinates, the input is represented by point positions relative to these five landmarks. Consider the point $\mathbf{p}(\mathbf{x}, \mathbf{y}, \mathbf{z})$ on the expressive input face as shown in Figure 4.2. Its position relative to these key landmarks is described, for example, by its location with respect to the right eye corner, denoted as lmk_1 in Figure 4.2. This relative position is expressed as $\mathbf{p}(\mathbf{x} - \mathbf{x}_{lmk_1}, \mathbf{y} - \mathbf{y}_{lmk_1}, \mathbf{z} - \mathbf{z}_{lmk_1})$.

Upon passing through the 'ExpNet', expressive components are removed from the input face shape, the network predicts point position translations to achieve neutralised deformation. Ideally, in the parts-based Deform-Net (the deformation network), the on-surface points corresponding to each predefined facial part should further morph within their respective regional scopes, relative to their corresponding landmarks, as they are processed by each parts-based deformation module. This

enables the learning of corresponding swapped features and the alignment with the template shape. Taking the nose part as an example (the eyes and mouth regions follow in a similar process), in box 'N' of Figure 4.2, the point $\mathbf{p}_1(\mathbf{x}_1, \mathbf{y}_1, \mathbf{z}_1)$ from subject A is initially defined within the nose region. After passing through 'NoseNet', all points on the subject A's nose move to align with the template. This indicates that in box 'E', subject A retains the template nose but all other regions remain unchanged. Therefore, the displacement in the position of $\mathbf{p}_1$ can be observed in box 'E' as it moves to $\mathbf{p}_1(\mathbf{x}', \mathbf{y}', \mathbf{z}')$. Simultaneously, in box 'E', the vertices in the eye regions are defined and undergo displacements as they pass through the subsequent network.

For each block network shown in Figure 4.2, for example 'ExpNet', consists of two primary components: a hyperparameters network, referred to as 'Hyper-Net', and a deformation network, denoted as 'Deform-Net'. Following the idea from the SIREN approach [91], we do not learn the latent representations directly from the deformation network. Instead, the initialisation for training and validating the distributions of these latent representations is achieved through a hyperparameters network that maps the latent codes to the specific weights and biases of our deformation network. More specifically, the expressive and parts-based latent codes $\mathbf{z}_{exp} \in \mathbb{R}^e$ and $\mathbf{z}_{part} \in \{\mathbb{R}^d, \mathbb{R}^{d'}\}$, following zero-mean multivariate Gaussian distributions, are input into a auto-decoder network to be mapped to weights (*e.g.* $\mathbb{R}^d \rightarrow \mathbb{R}^k$). The stacked hyperparameter network and the deformation network for one facial region are defined as follows:

$$\hat{\mathbf{p}} = \mathcal{D}_{\mathbf{W},\mathbf{B}}(\mathbf{p}) + \mathbf{p} = \mathcal{D}(\mathcal{H}_{\mathbf{z}}, \mathbf{p}) + \mathbf{p}, \tag{4.4}$$

where $\mathcal{D}$ represents the Deform-Net and $\mathcal{H}$ represents the Hyper-Net. $\mathcal{D}(\mathcal{H}_{\mathbf{z}}, \mathbf{p}) = \vec{v} \in \mathbb{R}^3$ is used for position translation based on the given on-surface point $\mathbf{p}$. The predicted translated point, denoted by $\hat{\mathbf{p}}$, is expected to be located in a position according to its corresponding point on the template face.

In the final deformation module of our model, denoted as 'RemNet' that transforms vertices of the remainder (excluding the nose, eyes and mouth) from specific individuals to the template mean face, except position translations on vertices, a displacement $\epsilon \in \mathbb{R}$ is used to control facial shape variation of faces and improve the facial shape reconstruction. Due to the variety in detail among human faces, point positional transformations are not sufficient to fit complex deformations. Therefore, displacements applied on signed distance fields are essential and the form of the final

Deform-Net, abbreviated as 'rem', is as follows,

$$\mathcal{D}_{rem} : \mathbf{p} \in \mathbb{R}^3 \rightarrow \left(\epsilon \in \mathbb{R}, \vec{v}_r \in \mathbb{R}^3\right) \tag{4.5}$$

SDF Network. In previous discussion, we explored one functional component, *i.e.*, sequential deformation networks. These networks yield latent representations for expressions and each predefined facial region. To reconstruct 3D shapes using implicit representations, a network for computing the SDF of sampled points on the template face should be introduced, as shown in Figure 4.1 as 'SDFNet'.

As illustrated in Figure 4.2, a fully-connected network SDFNet is employed as the final net in the end of our architecture to compute a signed distance for the template face. The final signed distance for the input face is represented as follows:

$$\Phi\left(\mathbf{p}\right) = \mathcal{S}\left(\mathbf{p} + \sum_{j}^{r(j)} \vec{v}_j\right) + \epsilon, \tag{4.6}$$

where $\mathcal{S}$ represents SDFNet and $r(j)$ ($r(j) \in \{\text{exp}, \text{n}, \text{e}, \text{m}, \text{r}\}$) corresponds to the expressive full face and four predefined facial regions, *i.e.*, nose, eyes, mouth and the remaining part, respectively.

Landmarks Networks Inspired by [83] and [117], a landmarks generative model $\mathcal{G}_{\mathbf{z}}$ and a neural blend skinning algorithm [57] are incorporated into our network to improve facial detail reconstruction, as shown in Figure 4.3.

A supervised MLP network is tailored to predict landmarks for each facial region, which helps bolster the effectiveness of the learnt parts-based latent representations. As demonstrated in Figure 4.3 (with the mouth region as an example), the latent representations are used to not only predict parameters for our deformation nets but also generate five landmarks per region. (We show the predefined semantic parts-based landmarks marked by different colours in Figure 4.4 and Figure 4.5 for different datasets.) The predicted landmarks further subdivide each predefined region into finer details, as deformations for input points are computed relative to these landmarks in a local semantic field, following a similar scheme to that employed in 'ExpNet', as shown in Figure 4.2. Following [117], we use a lightweight module to blend local fields into a global field, as depicted as 'Mini Weights-Net' in Figure 4.3, which is applied to generate the blend coefficients of each subdivided field based on the five landmarks for each region. Thus, our final SDF $\Phi\left(\mathbf{p}\right)$ is an extension of

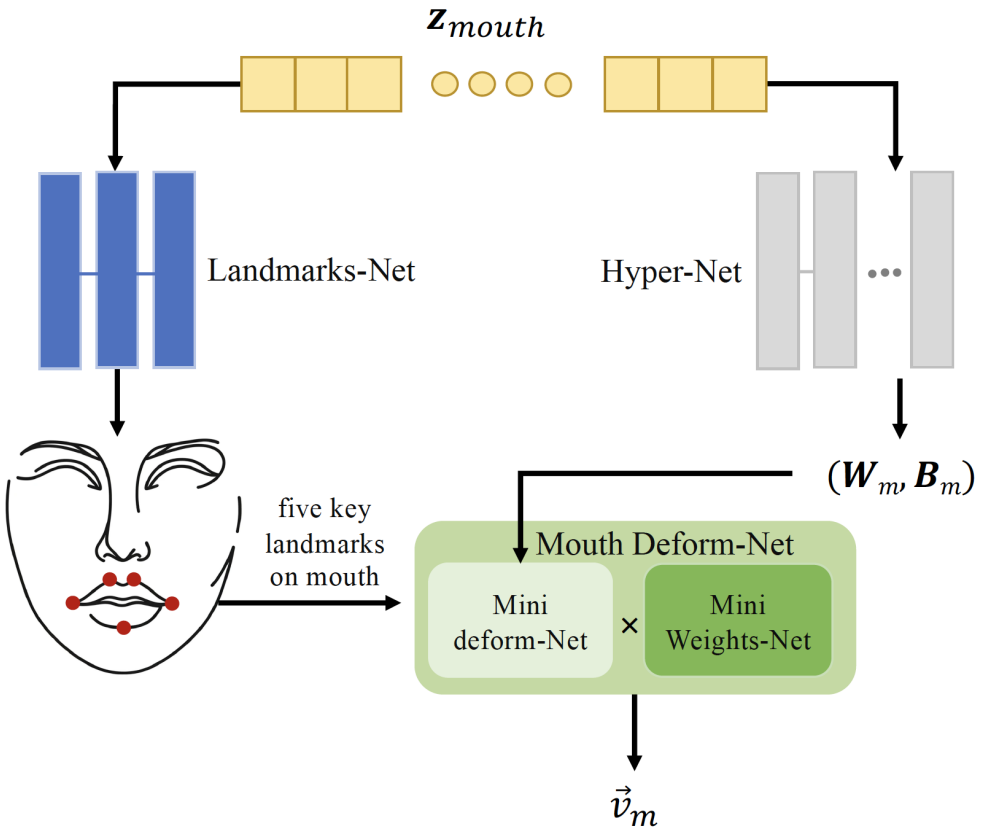

Figure 4.3: The landmarks generative model and detailed deformation network for the mouth region. The landmarks generative network, marked in blue and denoted as 'Landmarks-Net', outputs the positional coordinates of five key landmarks within the mouth region. This facilitates the subdivision of the mouth into smaller fields, which are then blended together using the weights generated by the 'Mini Weights Net'.

Equation (4.6), as follows:

$$\Phi(\mathbf{p}) = \mathcal{S}\left(\mathbf{p} + \sum_{j}^{r(j)} \sum_{l=1}^{L} \omega_l \left(\mathbf{p}, \mathbf{p}_l^j\right) \left(\vec{v}_j, \mathbf{p}_l^j\right)\right) + \sum_{l=1}^{L} \omega_l \epsilon_l, \tag{4.7}$$

where L is the number of landmarks for each facial region, ω represents the blend coefficients generated from the 'Mini Weights-Net' to weigh local facial regions based on predefined facial landmarks as shown in Figure 4.3.

Summary To summarise and clarify the design of the sequential deformation networks, we present separate pseudo-code algorithms in Algorithms 1, 2 and 3 to describe the workings of our three networks: 'ExpNet' (similar to 'NoseNet', 'EyesNet', and'MouthNet'), 'RemNet' and 'SDFNet'.

Algorithm 1 ExpNet Deformation Process

Input: Expression Representations $\mathbf{z}_{exp} \in \mathbb{R}^e$, Input Points $\mathbf{P} \in \mathbb{R}^{N\times 3}$ ($\mathbf{p} \in \mathbb{R}^3$),
Output: Deformed Points $\mathbf{P}' \in \mathbb{R}^{N\times 3}$ ($\mathbf{p}' \in \mathbb{R}^3$)

Initialise $\mathbf{z}_{exp}$
Hyper-Net $\mathcal{H}(\mathbf{z}_{exp}) \rightarrow$ Mini deform-net weights $(\mathbf{W}_{exp}, \mathbf{B}_{exp})$
Landmarks-Net$(\mathbf{z}_{exp}) \rightarrow$ face landmarks $\mathbf{p}_{lmk} \in \mathbb{R}^{l\times 3}$ where $l = 5$
for $l = 1$ to 5 **do**
    Compute point relative coordinates $\mathbf{p}^l_{\text{relative}} = \mathbf{p} - \mathbf{p}^l_{lmk}$
    Predict local deformations $\vec{v}_l$ for $\mathbf{p}^l_{\text{relative}}$ using Mini deform-Net
    Compute blend weights ω_l for part l using Mini Weights-Net
end for
Compute final deformations $\vec{v} = \sum_{l=1}^{5} \omega_l \times \vec{v}_l$
Apply transformation $\mathbf{p}' = \mathbf{p} + \vec{v}$
return $\mathbf{p}'$

Algorithm 2 RemNet Deformation Process

Input: Remainder Part Representations $\mathbf{z}_{rem} \in \mathbb{R}^{d'}$, Input Points $\mathbf{P} \in \mathbb{R}^{N\times 3}$ ($\mathbf{p} \in \mathbb{R}^3$),
Output: Deformed Points $\mathbf{P}' \in \mathbb{R}^{N\times 3}$ ($\mathbf{p}' \in \mathbb{R}^3$) and displacement ϵ

Initialise $\mathbf{z}_{rem}$
Hyper-Net $\mathcal{H}(\mathbf{z}_{rem}) \rightarrow$ Mini deform-net weights $(\mathbf{W}_{rem}, \mathbf{B}_{rem})$
Landmarks-Net$(\mathbf{z}_{rem}) \rightarrow$ face landmarks $\mathbf{p}_{lmk} \in \mathbb{R}^{l\times 3}$ where $l = 5$
for $l = 1$ to 5 **do**
    Compute point relative coordinates $\mathbf{p}^l_{\text{relative}} = \mathbf{p} - \mathbf{p}^l_{lmk}$
    Use the Mini deform-Net to predict local deformations $\vec{v}_l$ for $\mathbf{p}^l_{\text{relative}}$ and to compute the displacement ϵ to correct the final SDFs
    Compute blend weights ω_l for part l using Mini Weights-Net
end for
Compute final deformations $\vec{v} = \sum_{l=1}^{5} \omega_l \times \vec{v}_l$
Apply transformation $\mathbf{p}' = \mathbf{p} + \vec{v}$
return $\mathbf{p}', \epsilon$

Algorithm 3 SDFNet Computation Process

Input: Template Face Landmarks $\mathbf{p}_{lmk} \in \mathbb{R}^{l\times 3}$ where $l = 5$, Input Points $\mathbf{P} \in \mathbb{R}^{N\times 3}$ ($\mathbf{p} \in \mathbb{R}^3$),

Output: Template Face SDFs s

1: **for** $l = 1$ to 5 **do**
2: Compute point relative coordinates $\mathbf{p}^l_{\text{relative}} = \mathbf{p} - \mathbf{p}^l_{lmk}$
3: Use the Mini deform-Net to compute local SDFs s_l
4: Compute blend weights ω_l for part l using Mini Weights-Net
5: **end for**
6: Compute final template face SDFs $s = \sum_{l=1}^{5} \omega_l \times s_l$
7: **return** s

4.1.4 Dataset Augmentation by Facial Part Swapping

Affine Transformation for Swapping In order to augment our training datasets, we swap facial features (nose, eyes, and mouth) across pairs of subjects, using an affine transformation that optimally (least squares) matches the facial feature peripheral vertices into the graft site vertices of the face/head. We predefined surface regions for the nose, eyes and mouth on the FaceScape dataset [112, 120], and used the parts division scheme provided by the FLAME fitting of the Headspace dataset [24, 122]. Figure 4.4 and Figure 4.5 show the region definitions for the FaceScape and Headspace datasets in a colour coding, respectively.

To train our network, we create composite faces from a pairs of subjects (a, b) in the training dataset partition, where a composite face is composed from the surface parts set as: $\mathbb{P} = \{\text{nose}_a, \text{eyes}_a, \text{mouth}_a, \text{rem}_b\}$. Figure 4.4 and Figure 4.5 show a 3×3 array of face shapes, where each column represents a different subject ($a_{1\ldots 3}$), while subject b, which supplies the remainder part, is kept constant. Then, as we progress through the rows - the nose, then the eyes are deformed towards the learnt template shape. The shape shown under the Figure 4.4/4.5 colour coding additionally has the mouth deformed and so has the nose, eyes and mouth of the template and the remainder part is that of subject b. This final surface part is deformed by RemNet to generate the full template shape.

To independently learn latent representations for various facial regions and to augment the facial regions dataset, we sequentially swap facial features with those from several random subjects, where, from the augmentation perspective, different affine transforms are required for different subject pairings. The order of facial part deformation is arbitrary but must be consistent with the sequence adopted

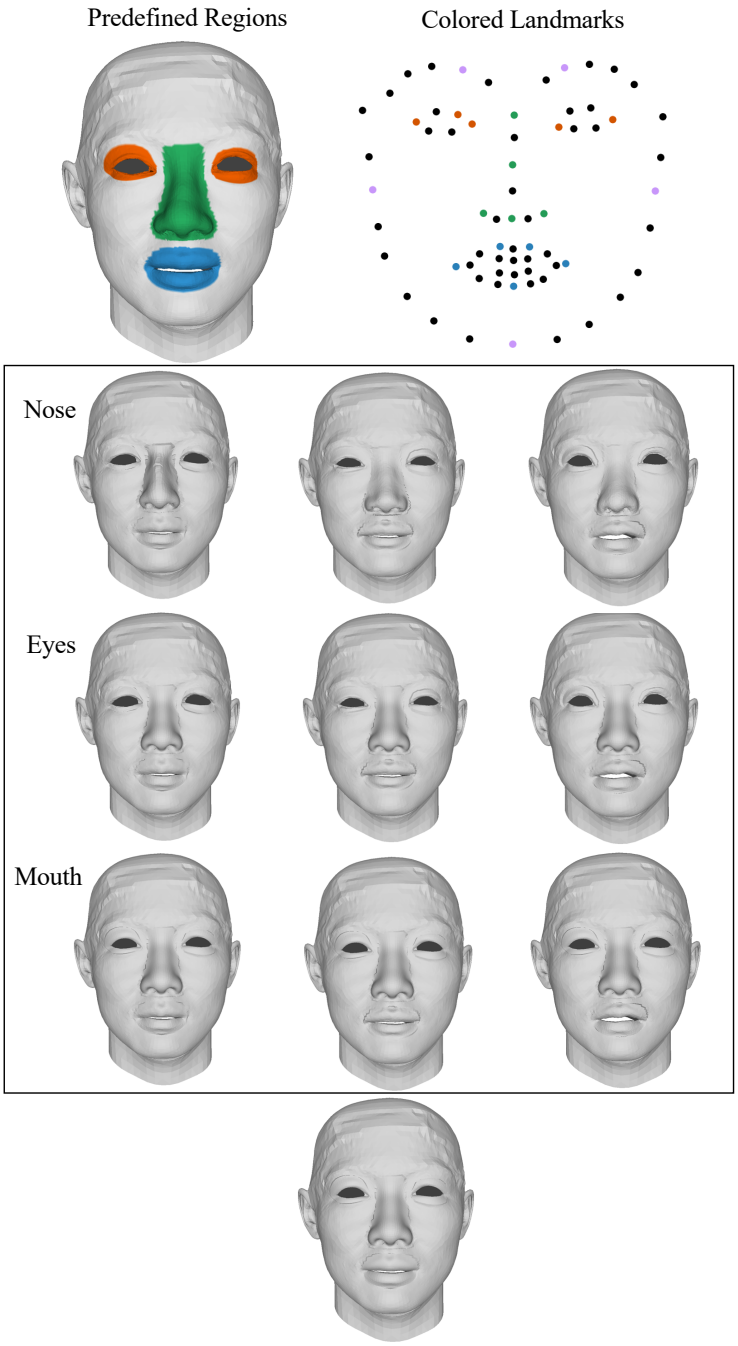

Figure 4.4: Predefined facial regions and semantic part-based landmarks on the FaceScape dataset. On the top left, the nose, eyes, and mouth parts are marked in green, orange, and blue, respectively. On the top right, five feature-salient landmarks are selected for each region, i.e., nose, eyes, mouth, and remainder, and are marked in green, orange, blue, and purple colors. In the 3×3 block, the first row shows composite faces with subject pairings: (a_1, b), (a_2, b), (a_3, b). The second row shows the nose feature being replaced by that of the template, and the third row additionally shows the eyes being replaced by that of the template. The bottom shape has all template features except the remainder part, which is that of subject b.

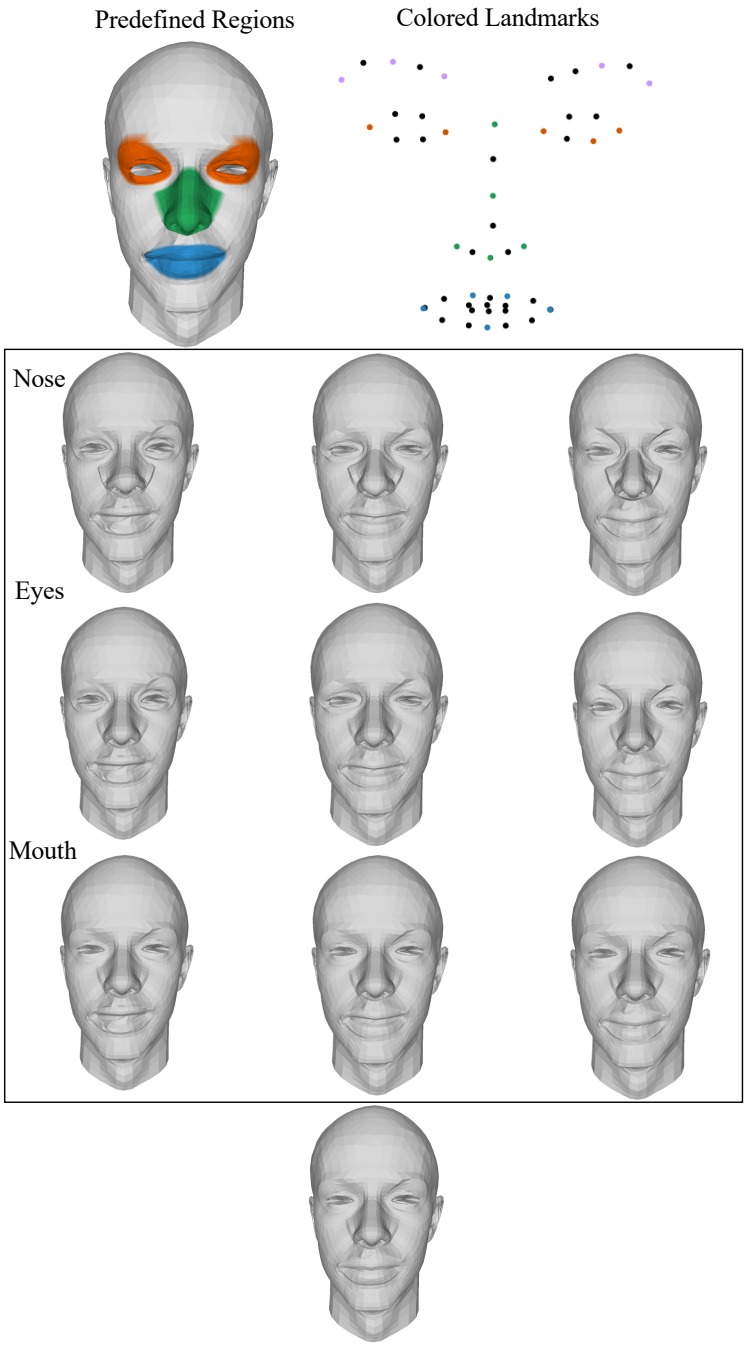

Figure 4.5: Predefined facial regions and semantic part-based landmarks on the Headspace dataset. On the top left, the nose, eyes, and mouth parts are marked in green, orange, and blue, respectively. On the top right, five feature-salient landmarks are selected for each region, i.e., nose, eyes, mouth, and remainder, and are marked in green, orange, blue, and purple colors. In the 3×3 block, the first row shows composite faces with subject pairings: (a_1, b), (a_2, b), (a_3, b). The second row shows the nose feature being replaced by that of the template, and the third row additionally shows the eyes being replaced by that of the template. The bottom shape has all template features except the remainder part, which is that of subject b.

by the latent codes learning networks for each facial region. In the design of our network, the sequence of learning nets is arranged as nose, eyes, mouth, and then the remainder.

For example, when processing the original face b, one of the input features-swapped faces retains the nose, eyes and mouth of the swapped subject a, but keeps the remainder part same as b, as depicted in the first row of the 3×3 face array in Figure 4.4 and Figure 4.5. Sequentially, after passing through the 'NoseNet' in our pipeline, the nose feature of a is removed to align with the template nose, while the eyes and mouth remain those of a. This step focusses on learning the nose latent variables, as shown in the second row of the face array in Figure 4.4 and Figure 4.5. The subsequent processes follow a similar pattern: the input goes through the 'EyesNet' to replace the specific eyes feature with those of the template (as illustrated in the third row, which shows the nose and eyes of template, the mouth of a, and the remainder of b). Then, the 'MouthNet' removes the mouth of a so that the input to 'RemNet' consists of a face with the template's nose, eyes, mouth, and the remainder of b. Finally, after processing through 'RemNet', all subjects become identical to the template.

Therefore, our part networks are strategically designed to learn the corresponding latent vectors using this approach. Each parts-based hyperparameter network outputs its corresponding factors based on the parts-based latent embeddings, enabling the model to learn the deformation weights separately as well as in an end-to-end manner.

While it is feasible to further subdivide the *remainder* surface into smaller parts (*e.g.* chin, forehead, cheeks), differences among these parts are harder to observe. Further subdivision increases network training time. Focusing on three key parts, *i.e.*, nose, eyes, mouth, is sufficient for us to demonstrate the efficacy of our model.

Laplacian Deformation In our approach, where an affine transformation is used to swap facial features among various subjects, obvious discontinuities appear between the swapped facial region and the remaining part of face. This is because the least-squares optimal fitting of the facial part to the graft site has some residual error. To address it, we employ Laplacian deformation [93, 82], based on the Laplacian operator, to preserve the geometric details of face meshes, which ensures a smoother transition between the swapped feature and the original face.

After aligning all face shapes, we firstly perform a least square alignment of facial part edge vertices for the swapped face to the graft site. In the following step, we deform the original part to the affine transformed facial feature using Laplacian

regularised deformation.

In detail, we use Laplacian coordinates, denoted as $\Delta = [\delta_1, \ldots, \delta_n]^T$, where each $\delta_i = \mathcal{L}(\mathbf{p}_i) = \mathbf{p}_i - \frac{1}{d_i}\sum_{j \in \mathcal{N}_i} \mathbf{p}_j$ (d_i is the degree of vertex $\mathbf{p}_i$, and $\mathcal{N}_i$ is the set of neighbouring points of $\mathbf{p}_i$). The Laplacian coordinate matrix Δ is computed as $\Delta = (I - D^{-1}A)P$, with P being the matrix of vertex positions of the first affine-transformed face parts. To ensure that the details at the junctions remain unchanged, the vertices obtained from the least squares fitting should be consistent with the vertices of swapped features, which is expressed as $\Delta - \mathcal{L}(P') = 0$, where P' represents the positions of vertices after Laplacian deformation, representing our targeted vertex positions. This ensures that the junctions blend seamlessly with the rest of the mesh.

4.1.5 Loss Functions

To learn signed distance fields, given that the ground truth signed distance values of on-surface and near-surface points can be obtained, we define the loss function $\mathcal{L}_{rec}$ as follows:

$$\mathcal{L}_{rec} = \mathcal{L}_{SDF} + \lambda_{gt} \sum_{\mathbf{p} \in \Omega} \mathcal{L}\left(\Phi\left(\mathbf{p}\right), s\right), \tag{4.8}$$

where $\mathcal{L}_{SDF}$ is the loss function introduced in Equation (4.3), and λ_{gt} is the weight for the second term. We employ the l_1-norm as the loss metric for the predicted signed distance of the sampled points $\mathbf{p}$ (as defined in Equation (4.7)) and their corresponding ground truth signed distance s. This loss function is used to constrain the final SDFs in 3D face reconstruction and to regulate displacements in facial features.

For the learning of parts-based latent representations, we introduce a regularisation loss $\mathcal{L}_{reg}$ for all latent embeddings, defined as follows:

$$\mathcal{L}_{reg} = \sum_{j}^{r(j)} \left\|\mathbf{z}_j\right\|_2, \tag{4.9}$$

where $r(j) \in \{\mathrm{exp}, \mathrm{n}, \mathrm{e}, \mathrm{m}, \mathrm{r}\}$ represent the expressive full face, nose, eyes, mouth and remainder parts, respectively.

The loss for landmarks $\mathcal{L}_{lmk}$ is defined as:

$$\mathcal{L}_{lmk} = \lambda_{dl}\mathcal{L}\left(\mathcal{D}\left(\mathbf{p}_{lmk}\right), \mathbf{p}_{lmk}^T\right) + \lambda_{gl} \sum_{j}^{r(j)} \mathcal{L}\left(\mathcal{G}_{\mathbf{z}_j}, \mathbf{p}_{lmk}^j\right), \tag{4.10}$$

where λ_{dl} and λ_{gl} are weights for the landmarks deformation loss and the landmarks generation loss, respectively. The l_1-norm is employed to enforce the alignment between deformed original facial landmarks $\mathcal{D}(\mathbf{p}_{lmk})$ and the template landmarks $\mathbf{p}_{lmk}^T$. Additionally, the second term in Equation (4.10) is the loss function for the landmarks-generative model $\mathcal{G}_{\mathbf{z}}$, where $\mathbf{p}_{lmk}^j$ represents the corresponding ground truth landmarks of the j-th part.

Therefore, our network is trained in an end-to-end manner by minimising the final loss function, denoted as:

$$\mathcal{L} = \mathcal{L}_{rec} + \mathcal{L}_{lmk} + \lambda_{reg}\mathcal{L}_{reg}, \tag{4.11}$$

where λ_{reg} is the weight for the regularisation loss of all latent embeddings.

During inference, the network's weights are fixed, and the optimal latent representations $\mathbf{z}_j$ are determined as follows:

$$\mathbf{z}_j^* = \arg\min_{\mathbf{z}_j} \left(\sum_{(\mathbf{z}_j, \mathbf{p})} \mathcal{L}_{rec}(\mathbf{z}_j, \mathbf{p}) + \sum_{\mathbf{z}_j} \mathcal{L}_{reg}(\mathbf{z}_j) \right). \tag{4.12}$$

4.2 EVALUATION

In this section, we present a comprehensive evaluation of our method for implicit representation learning of 3D facial regions. We begin by introducing two datasets used for our evaluation: one focuses on 3D faces and the other on 3D heads. We then outline the four methods against which we compare our method. Further, we provide detailed explanations for the implementation settings and the evaluation metrics employed in our comparative analysis. This includes both qualitative and quantitative results to verify the robust performance of our model. Finally, we present our ablation studies, which explore the impact of the landmark selection strategies and Laplacian deformation within our approach. We conclude this section with discussions on both the limitations and contributions of our method, providing potential improvements as a foundation for our subsequent work.

4.2.1 DATASETS

In our experiments, we utilise two public datasets: FaceScape [112, 120] and Headspace [24]. To ensure a fair comparison, we align the training-to-test ra-

tio in our method with those in baseline methods. A detailed introduction to the datasets, along with their respective division ratios, is provided in the following.

FaceScape dataset [112, 120] is a large-scale detailed face dataset consisting of 847 subjects, each performing 20 expressions. For a fair comparison, we follow the split scheme for training and test sets as proposed in [117] Our employed dataset includes 365 publicly available individuals, using face scans of 355 subjects for training and 10 for the test set. For training expression-identity disentanglement, we use 17 expressions from each subject. Additionally, we randomly select 16 different subjects and swap their three features (nose, eyes and mouth) to train the parts-based branch. Consequently, the training set consists of 12,070 scans divided equally between expressive-identity representation learning and facial parts representation learning. The test set includes 170 unseen scans. Given the requirement of watertight shapes for the SDF, we apply the same data preprocessing scheme to crop the defined unit sphere and generate pseudo watertight shapes. In particular, each face scan is normalised to a scale unit of 1cm. The coordinate origin is set at a point 4mm behind the nose tip, as outlined in [117]. Subsequently, we crop any part of the head that extend beyond a sphere with a radius of 1cm. We then apply the Ray-Triangle Intersection Algorithm [74] to remove hidden surfaces, such as oral cavity. Finally, we use Delaunay triangulation [55] to properly orient the scans, as shown in Figure 4.6, and the final pseudo watertight shapes are obtained.

Headspace dataset [24] is a set of 3D images of the human head, consisting of 1,519 subjects. Due to the unregistration from the raw face data, we utilise the FLAME [62] fitting of the Headspace dataset as provided by [122], for the Headspace dataset. This enables the direct generation of watertight shapes from the densely corresponded 3D faces. During data preprocessing, we remove the internal structures. Our first step is to crop the neck region and remove the eyeballs for all subjects. Dealing with the eyeballs that are separate watertight surfaces becomes challenging when distinguishing between the outer and inner surface of the full head. Moreover, considering the complex geometry in the oral cavities, we decide to remove the back structure of the mouths as well. To ensure that these cropped shapes are suitable for learning with implicit representations, we manually close any open areas to guarantee their watertightness, as shown in Figure 4.7. For efficiency in time and memory, we randomly select 300 subjects from the whole dataset. Following a ratio of 9:1, we use 270 subjects for training and allocate the remaining 30 for the test set.

4.2.2 COMPARATIVE STUDIES

We conduct a comprehensive experiment to compare our method with four publicly available state-of-the-art methods, which mainly focus on implicit representations for 3D shape reconstruction. Specifically, we compare our approach with DeepSDF [79], i3DMM [113] and ImFace [117] on the Headspace dataset. These three solutions use implicit techniques for 3D shape reconstruction. We also compare our method with FLAME [62] on the FaceScape dataset. However, we do not include a comparison with FLAME on the Headspace dataset since we use FLAME-fitting data as ground truths for this dataset. All evaluations of these methods are conducted using the same training and test datasets. We report results we achieved on both sets of preprocessed data.

DeepSDF [79], proposed by Park *et al.*, introduces an auto-decoder model that directly maps a latent code to the output, in order to learn continuous SDFs for various classes of shapes, such as planes, cars and lamps. It facilitates numerous tasks, including shape completion, representation of complex topologies and high-quality shape reconstructions.

i3DMM, as presented in [113], proposes an implicit 3DMM for full heads, under the situation where dense correspondences between head scans are not necessary. This

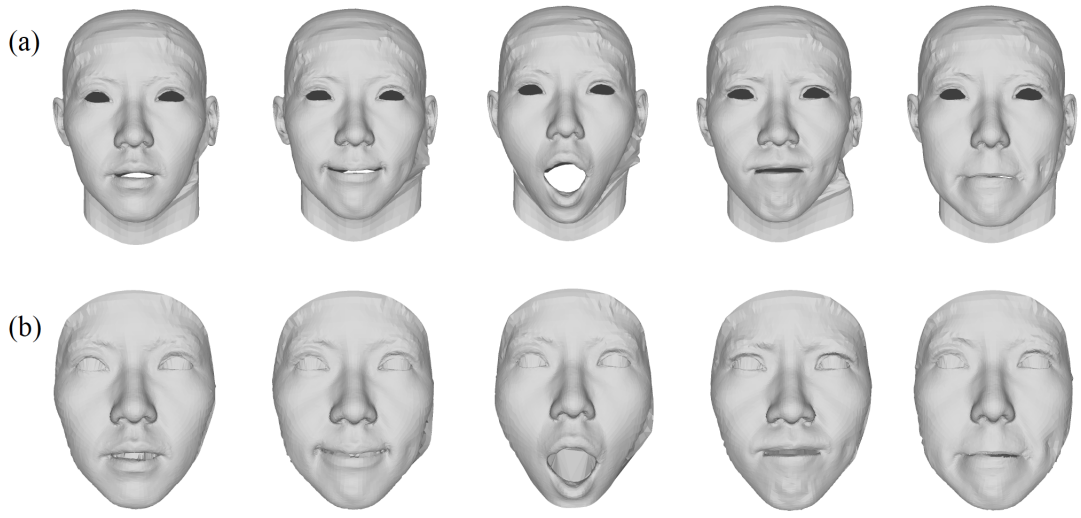

Figure 4.6: 3D face scans in FaceScape dataset [112, 120]. The sub-figure (a): the original faces in the dataset; (b): the corresponding preprocessed faces, using the same preprocessing method as described in [117].

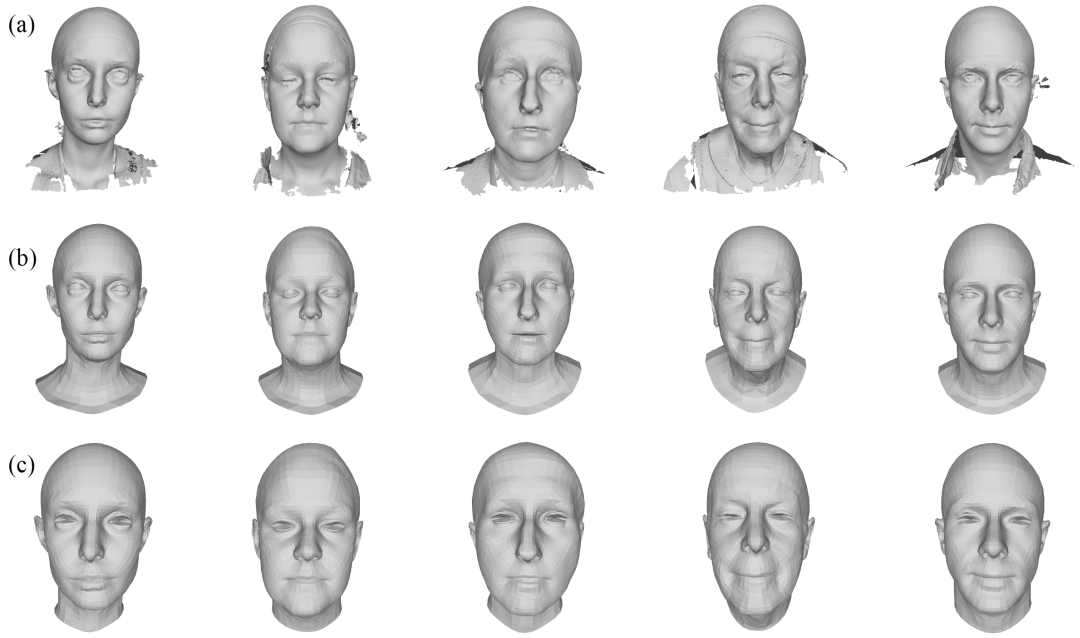

Figure 4.7: 3D face scans in Headspace dataset [24]. The sub-figure (a): the original faces in the dataset; sub-figure (b): the FLAME-fitting faces, as provided by [122]; sub-figure (c): the corresponding preprocessed watertight faces.

model disentangles the identity, expressions and hairstyle for geometry components, and separate the identity and hairstyle for color components.

ImFace [117] employs two explicitly disentangled deformation fields, *i.e.*, identity and expressions, to handle complex face variations. This method proposes a novel nonlinear 3DMM for face scans, using an effective pipeline to learn representations in a fine-grained and semantically meaningful manner.

Our approach was primarily inspired by ImFace that revolves around disentanglement of expressions and identities. Unlike ImFace, our architecture not only separates identities and expressions but also independently represents each facial region on a neutral expression in an implicit way. We adopt an auto-decoder architecture for our hyperparameter networks, with the deformation networks for each facial regions being based on the SIREN framework. We learn representations separately for each regions as well as expressions, which is different from those methods we introduced in Section 2.4.2. Similar to DeepSDF, i3DMM and ImFace, our method employs SDFs for representations, in contrast to the explicit approach, FLAME, which also serves as a baseline in our evaluation. Implicit learning becomes popular in recent

years as it learns a continuous field for the shape surface and imposes no limitations on reconstruction resolutions.

4.2.3 Implementation Details

We explain the architecture of our model using one part-based deformation module as the latter shares the same architecture with all the other modules. The Hyper-Net is a ReLU MLP with one hidden layer. Both the Deform-Nets and SDFNet consist of five fully connected layers, each followed by a sine activation function. The landmarks-generative network is a Leaky ReLU MLP with one hidden layer, similar in design to the 'Weights-Net', which also uses Leaky ReLU MLPs and a last layer with Softmax activations. We set the dimensions of the latent vectors to 48 for the nose, eyes and mouth modules, 112 for the remainder, and 128 for the expression latent codes across both the FaceScape and Headspace datasets. The detailed architectures for implementation are presented in Appendix B. To optimise the balance of each loss, various hyperparameters are explored, including λ_{Eik} being set to 50, λ_{normal} to 100, λ_{gt} to $3e3$, λ_{reg} to $1e6$, λ_{dl} to 100, and λ_{gl} to $1e3$.

The inputs to our network are point positions, normals and SDFs for faces with swapped features, only using affine transform, as shown in Figure 4.8 and Figure 4.9 for the FaceScape and Headspace dataset, respectively. For each mesh in both datasets, we sample 250,000 points, of which 235,000 points are located on the surface and each of these points comes with a corresponding signed distance value of 0. The remaining 15,000 points are uniformly distributed within the unit sphere. Their SDFs are pre-computed using the Python library [71], and we depict the visualised representations in Figure 4.10 for the FaceScape dataset and in Figure 4.11 for the Headspace dataset, respectively.

We implement the network using PyTorch and execute the training on two NVIDIA A40 GPUs. For the Headspace dataset, we train our model with a batch size of 120 over 800 epochs, while for the FaceScape dataset, we use a batch size of 36 and train for 850 epochs. Additionally, we optimise for 1,000 epochs to fit latent representations on both datasets. The Adam Optimiser [52] is employed with an initial learning rate of 1×10^{-4}. A learning rate decay factor of 0.95 is applied every 10 epochs, starting from the 200th epoch. The training process takes approximately 47 hours for the Headspace dataset and 124 hours for the FaceScape dataset.

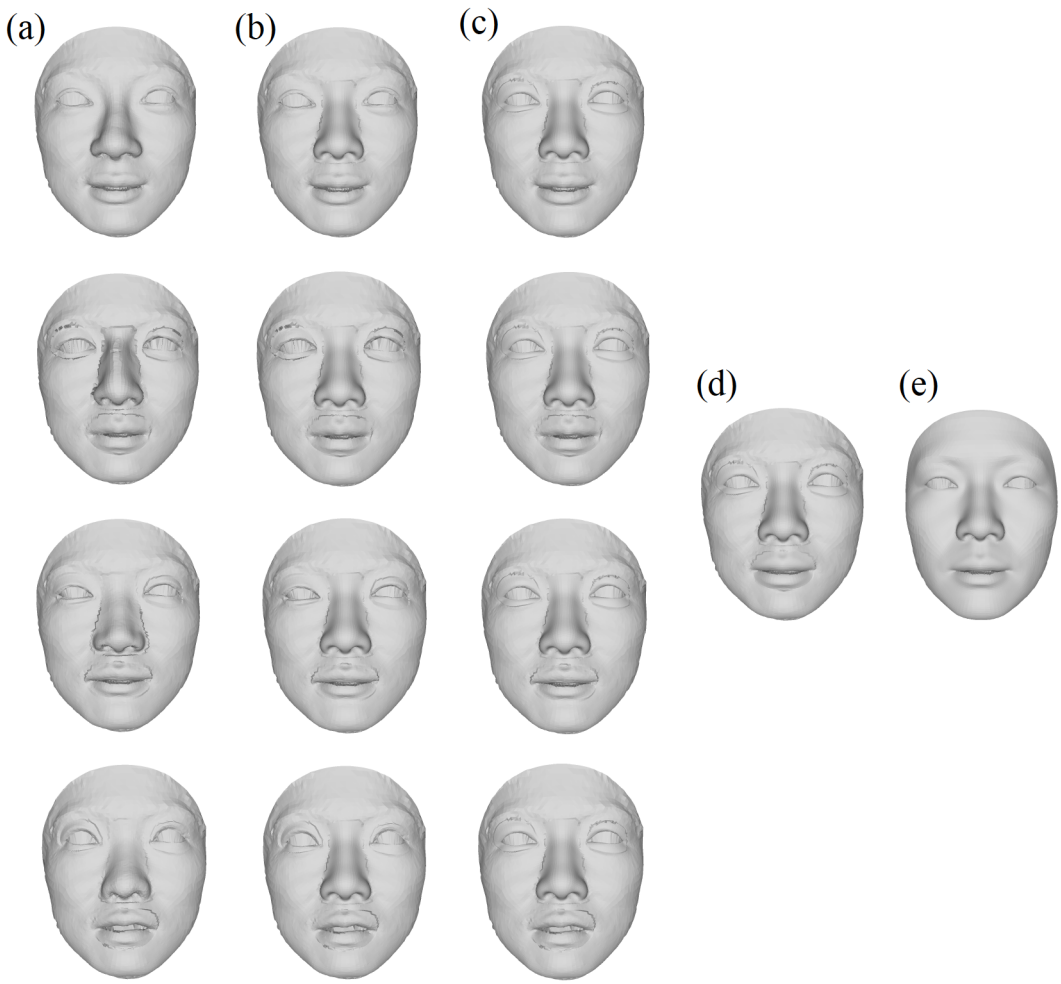

Figure 4.8: Swapped facial parts features in the FaceScape dataset. Each column displays four different swapped faces. In column (a), subject A is shown with four different sets of noses, eyes and mouths, considering as the ground truths for the 'NoseNet' input. Column (b) illustrates the corresponding ground truths for 'EyesNet', featuring subjects with the template nose and four different subjects' eyes and mouths. Column (c) shows the ground truths for the 'MouthNet', where different subjects' mouths are swapped, but the nose and eyes remain consistent with the template. (d) represents subject A's remainder combined with the template's nose, eyes and mouth, as the ground truths for 'RemNet'. (e) shows the template as the ground truths of 'SDFNet'.

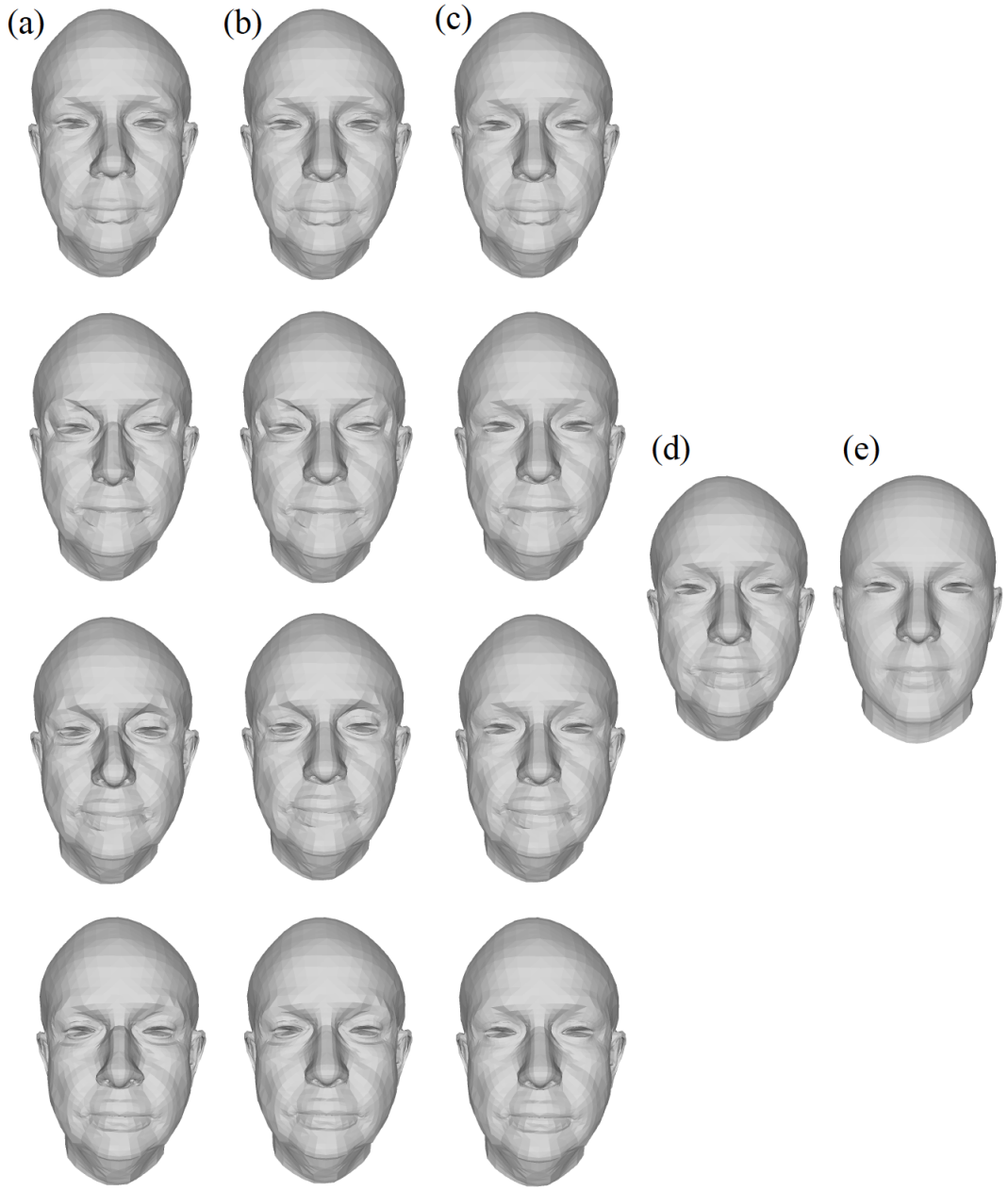

Figure 4.9: Swapped facial parts features in the Headspace dataset. Each column displays four different swapped faces. In column (a), subject A is shown with four different sets of noses, eyes and mouths, considering as the ground truths for the 'NoseNet' input. Column (b) illustrates the corresponding ground truths for 'EyesNet', featuring subjects with the template nose and four different subjects' eyes and mouths. Column (c) shows the ground truths for the 'MouthNet', where different subjects' mouths are swapped, but the nose and eyes remain consistent with the template. (d) represents subject A's remainder combined with the template's nose, eyes and mouth, as the ground truths for 'RemNet'. (e) shows the template as the ground truths of 'SDFNet'.

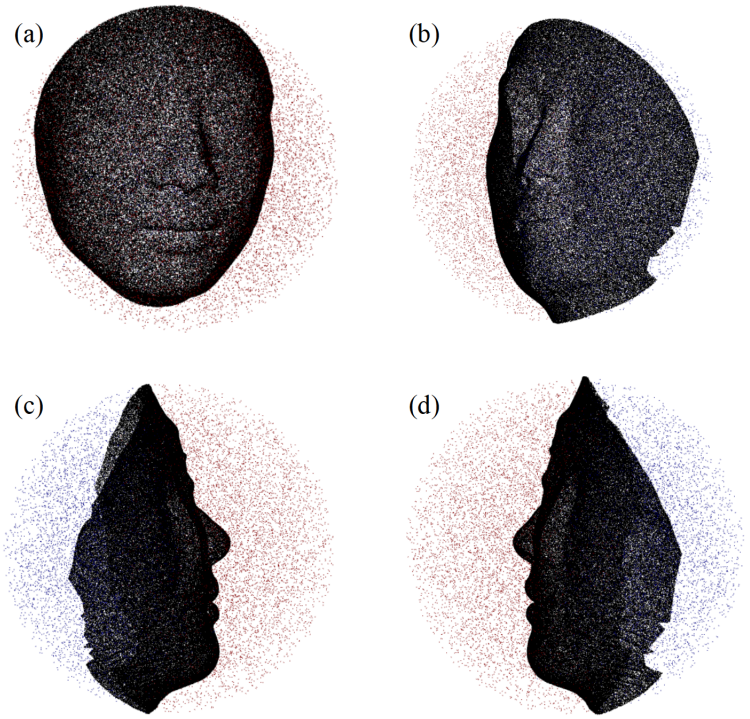

Figure 4.10: The SDF representations of one face in the FaceScape dataset are shown from four different perspectives (a)-(d). The points in black represent the implicit surface (SDF = 0); the points in blue represent the inside part (SDF $<$ 0); and the points in red represent the outside part (SDF $>$ 0).

4.2.4 Evaluation Metrics

For a fair comparison, we adopt the same evaluation metrics as those used in [113] and [117]. We re-run all the methods under comparison on both of our datasets, employing these metrics to compare our performance. The Symmetric Chamfer Distance (SCD) and F-Score are commonly used for evaluating 3D shape reconstructions, particularly in the context of uncorresponded shapes. The Chamfer Distance measures the distance between two sets of points. In our evaluation, we employ the SCD that computes the distance bidirectionally, the equation as follows:

$$SCD(A,B) = \frac{1}{n}\sum_{i=1}^{n}\min_{j}\|A_i - B_j\|_2 + \frac{1}{n}\sum_{i=1}^{n}\min_{j}\|B_i - A_j\|_2, \tag{4.13}$$

where A and B represent our ground truth shape and its corresponding reconstruction shape, respectively. Due to the variability in the number of vertices in the reconstruction shapes – caused by random resolutions and the use of the Marching

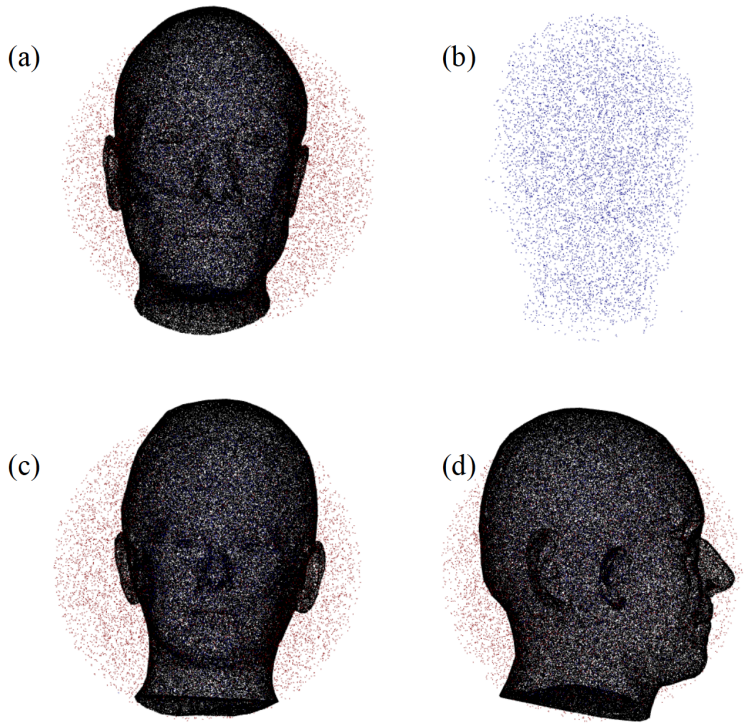

Figure 4.11: The SDF representations of one face in the Headspace dataset are shown from three different perspectives (a) frontal, (c) back, and (d) side. The points in black represent the implicit surface (SDF = 0); the points in blue represent the inside part (SDF < 0); and the points in red represent the outside part (SDF > 0). Given that the full head is a watertight shape, the inside part, marked in blue, is difficult to be observed when the outside part and surface are displayed. Thus, we exclusively show the inside part (SDF < 0) in the sub-figure (b). Sub-figures (a) and (b) share the same perspective, and by combining them, a comprehensive view of the inside, outside and surface areas is provided.

Cubes algorithm in the final generation step – we sample 150,000 vertices from both generated shapes and ground truths. Thus, in Equation (4.13), the variable n denotes this consistent number of vertices. The terms A_i, B_i, A_j and B_j represent the sampled points.

Additionally, we employ F-Score as another metric to measure the completeness and accuracy of 3D reconstructed shapes. The completeness is obtained from computing the distance from each sampled point on the ground truth face shape to the nearest point in the corresponding reconstructed shape. The accuracy, conversely, is measured the distance from each sampled point on the reconstruct face shape to

Table 4.1: SCD results for expressive original 3D face shape reconstruction on the FaceScape dataset [112, 120]. Compared with DeepSDF [79], FLAME [62], i3DMM [113], and ImFace [117].

Method	Part				
	Full Face	Nose	Eyes	Mouth	Rem
DeepSDF [79]	1.9393	2.0287	1.5491	1.462	1.982
FLAME [62]	1.483	0.623	0.803	0.717	0.695
i3DMM [113]	0.875	0.622	**0.564**	0.652	0.693
ImFace [117]	**0.567**	0.578	0.582	0.607	0.570
Ours	0.598	**0.558**	0.579	**0.585**	**0.519**

Table 4.2: F-Score results for expressive original 3D face shape reconstruction on the FaceScape dataset [112, 120]. Compared with DeepSDF [79], FLAME [62], i3DMM [113], and ImFace [117].

Method	Part				
	Full Face	Nose	Eyes	Mouth	Rem
DeepSDF [79]	25.69	27.28	35.21	37.56	27.39
FLAME [62]	75.78	87.23	72.08	76.78	84.00
i3DMM [113]	74.91	86.56	89.40	81.74	84.19
ImFace [117]	**94.81**	90.15	88.75	84.85	96.40
Ours	92.86	**91.41**	**89.40**	**86.67**	**96.52**

Table 4.3: SCD results for original 3D face shape reconstruction on the Headspace dataset [24, 122]. Compared with DeepSDF [79], i3DMM [113] and ImFace [117].

Method	Part				
	Full Face	Nose	Eyes	Mouth	Rem
DeepSDF [79]	0.9809	1.1972	1.0740	0.9027	0.8612
i3DMM [113]	0.9009	0.7126	**0.5623**	0.6710	0.8810
ImFace [117]	**0.6992**	0.7173	0.6966	0.7077	0.7357
Ours	0.7184	**0.7093**	0.6496	**0.5910**	**0.7207**

the ground truth. We apply the specified threshold, 0.01, for all comparative analyses to determine the proportions of distances falling within this threshold. It provides a balanced measure of both completeness and accuracy of the reconstruction.

Table 4.4: F-Score results for original 3D face shape reconstruction on the Headspace dataset [24, 122]. Compared with DeepSDF [79], i3DMM [113] and ImFace [117].

Method	Part				
	Full Face	Nose	Eyes	Mouth	Rem
DeepSDF [79]	70.41	49.23	55.00	63.95	73.47
i3DMM [113]	69.61	79.67	**89.17**	81.73	70.51
ImFace [117]	**84.22**	75.71	79.92	78.07	80.93
Ours	82.03	**81.75**	84.57	**87.26**	**82.13**

4.2.5 RECONSTRUCTION EVALUATION

In our experiments, we evaluate the capability of our model for 3D face reconstruction using SCD and F-Score, as introduced in Section 4.2.4. SCD is estimated by sampling 150,000 surface points from both the generated and ground truth full faces. To further demonstrate the effectiveness of our learnt part latent representations, we present results not only for full face reconstruction but also for the reconstruction of each individual facial part (nose, eyes and mouth). For evaluating these part latent vectors, we sample 6000 points for each part in the FaceScape dataset and 10,000 points in the Headspace dataset.

We show the results for the all expressive original face shapes from the FaceScape dataset in Table 4.1 and Table 4.2, as well as visually in Figure 4.12. For the Headspace dataset, the corresponding results for the original face shapes (exclusive of any faces with swapped features) are detailed in Table 4.3 and Table 4.4, and illustrated in Figure 4.13. It is noteworthy that DeepSDF, which learns a latent code for each face shape and shows weak performance on capturing fine details, was re-trained exclusively on 355 neutral face shapes rather than the full set of expressive face shapes. Thus, the reconstructed expressive shapes by DeepSDF are not included in Figure 4.12.

As observed from Table 4.1 and Table 4.3, lower SCD values indicate better performance, as they represent a smaller distance between the reconstructed shapes and their corresponding ground truths. Our method demonstrates state-of-the-art performance in the reconstruction of local detail parts, particularly in the nose, mouth and remainder region across both datasets. While our results are slightly outperformed by ImFace in full face reconstruction – with ImFace achieving the best results for full face reconstruction, showing its effectiveness in capturing overall facial

structure – the minor gap in our full face reconstruction can be attributed to feature swapping in predefined regions. The swapping process affects the smoothness of the boundary between different facial parts, thereby impacting the overall SCD value for full face reconstruction.

From the results presented in Table 4.2 and Table 4.4, it is evident that the higher F-Score values indicate better performance. Similarly, as observed in Table 4.1 and Table 4.3, our method achieves the best performance in reconstructing individual facial part, while ImFace excels in full face reconstruction. We aim to address this issue and improve our performance in full face reconstruction by employing the Laplacian Blending algorithm. i3DMM demonstrates superior performance in reconstructing one specific part (the eyes) compared to other methods due to its preprocessing computation of ground truth SDFs wherein it samples more points around semantic landmarks.

In Figure 4.12, the first three columns depict face shapes with neutral expressions, and the remaining five columns show faces with different expressions. It can be proven that our method facilitates both neutral and expressive face reconstruction through our ExpNet and parts-based nets (NoseNet, EyesNet, MouthNet and RemNet). We do not train expressive faces with DeepSDF, as it struggles to accurately capture fine details, especially in the mouth region – one of the most expressive parts of the face, saving both time and memory.

From the results displayed in Figure 4.13, we can observe that our method demonstrates strong performance in both full face and specific part reconstructions, particularly in the mouth region. However, i3DMM performs slightly better in capturing certain details, *e.g.* in the eyes region, as it samples larger proportion of vertices around key facial features like the nose, eyes and mouth. The Headspace dataset consists of 3D shapes of the full head, which includes less semantically significant regions like the back of the head. Therefore, the focussed sampling of points in specific regions is advantageous for learning small local features on full heads. This may suggest that improving our method's performance in part reconstruction could be achieved by adopting a similar preprocessing strategy.

4.2.6 Parts-based Disentanglement

Our proposed method aims to disentangle latent embeddings from each predefined facial region. We conduct comprehensive experiments to evaluate the disentanglement

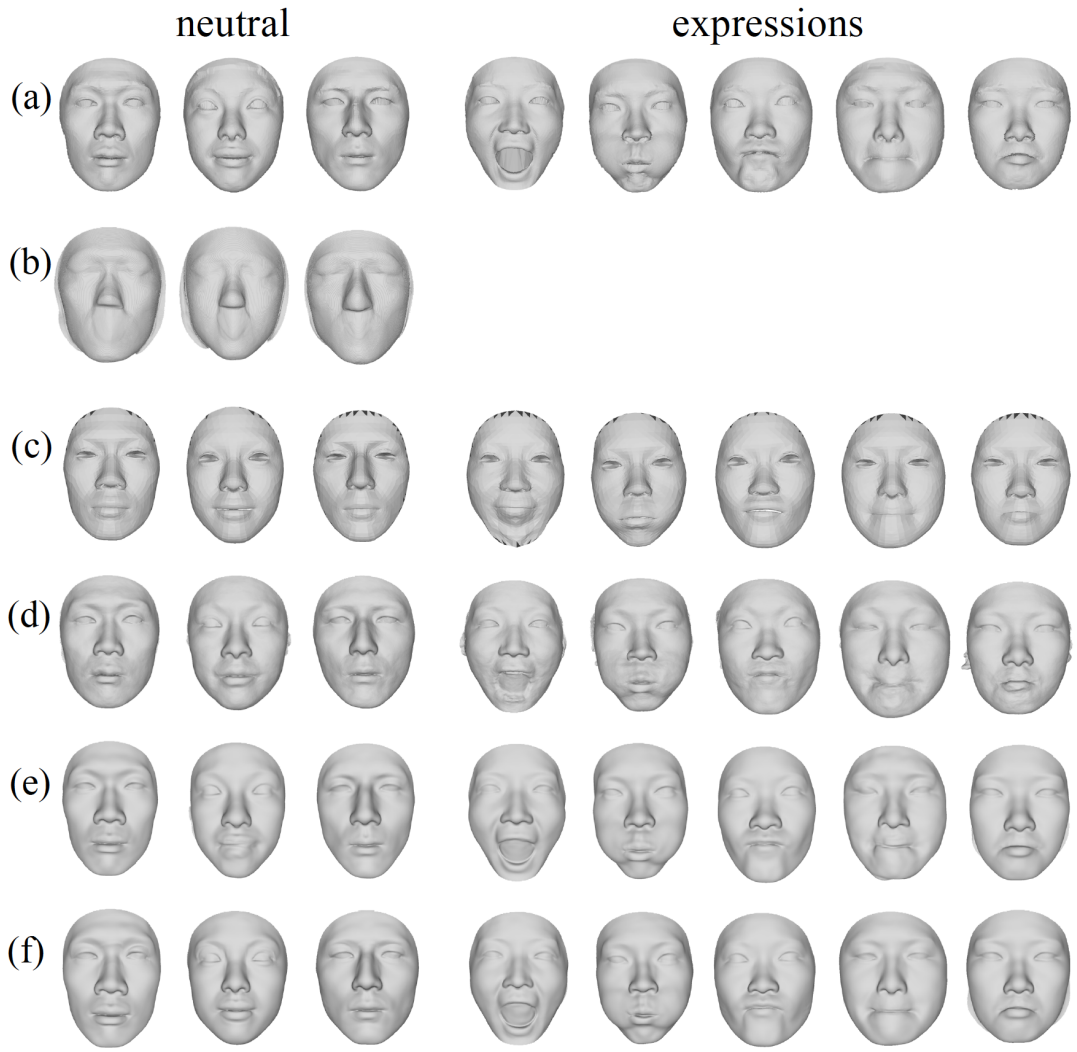

Figure 4.12: Face reconstruction for unseen face shapes on the FaceScape dataset. (a): Ground truths; (b) DeepSDF [79]; (c) FLAME [62]; (d) i3dMM [113]; (e)ImFace [117]; (f) and Our method, respectively. Improved qualitative performance is most evident in the mouth part. No generated expressive face shapes from DeepSDF [79] due to the weak performance on detailed learning, especially the variation on the expressive mouth.

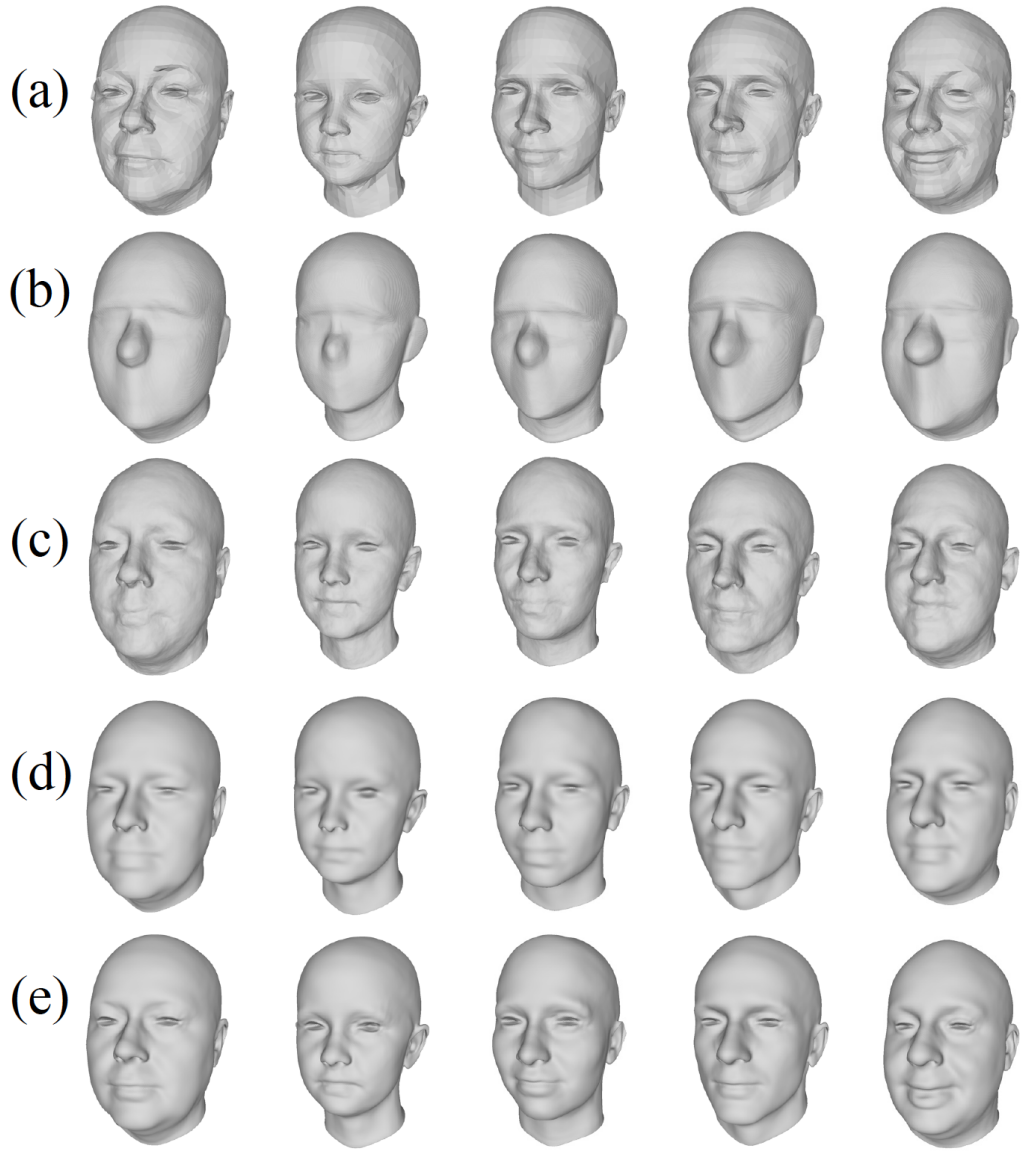

Figure 4.13: Face reconstruction for unseen face shapes on the Headspace dataset. (a): Ground truths; (b) DeepSDF [79]; (c) i3dMM [113]; (d) ImFace [117]; (e) and Our method, respectively. Note our qualitatively superior reconstruction around the semantic facial parts, particularly evident on the mouth region.

ability of our method. As presented in Figure 4.14 for the Headspace dataaset and Figure 4.15 for the FaceScape dataset, we perform parts-based latent code interpolation between two unseen reconstructed shapes from the test set in order to observe the gradual deformation of each individual part.

Specifically, as shown in Figure 4.14, we have two subjects denoted as Subject 'A' and Subject 'B' of the Headspace dataset. The reconstruction shapes of these two subjects are shown in the top box, which are generated from their corresponding latent representations of all facial regions. Using these predicted latent representations, we conduct part interpolations and visually illustrate the deformations of each facial region from Subject 'A' to Subject 'B' in the respective color blocks. Starting from the purple 'Nose' block, we display three noses in a sequence: the original nose of Subject 'A', the interpolated nose, and the original nose of Subject 'B', all presented from the same perspective to facilitate easier observation of the deformation process. Notably, during the deformation from A's nose to B's nose, the position of the nose tip shifts from the blue dot to the green dot, passing through the red dot. When viewed from the side perspective of the whole head, there are differences in the height of the nose tip, indicating that Subject 'A' has a higher nose bridge compared to Subject 'B'. In the second deformation example, as shown in the pink block of Figure 4.14, we observe the transition from Subject A's eyes to Subject B's eyes. During this process, the eyes gradually become smaller. In the green block displaying the deformation of the mouth region, the appearance of beards and a downward shift in the mouth corners can be observed. In the first three deformations, the nose, eyes and mouth, transformations are localised to the corresponding region. However, in the remainder block, the overall shape of Subject 'A' progressively aligns more closely with that of Subject 'B'. This comprehensive interpolations demonstrate how our model not only modifies specific facial features but also captures the differences between fine details among subjects.

In Figure 4.15, we show the interpolations of the learnt parts-based representations from the subject 'A' to subject 'B' for the FaceScape dataset. It is important to note that the deformation sequence is not strictly from nose to the remainder parts for either dataset. Due to the independence of corresponding part latent representations, various sequences can be achieved, *e.g.* interpolating from eyes, remainder, nose to the mouth. The error maps of the per-vertex distance between two shapes, are visualised in Figure 4.15. In the second row, the maps display the distance between the current mesh and the first shape of the corresponding part, while for each initial

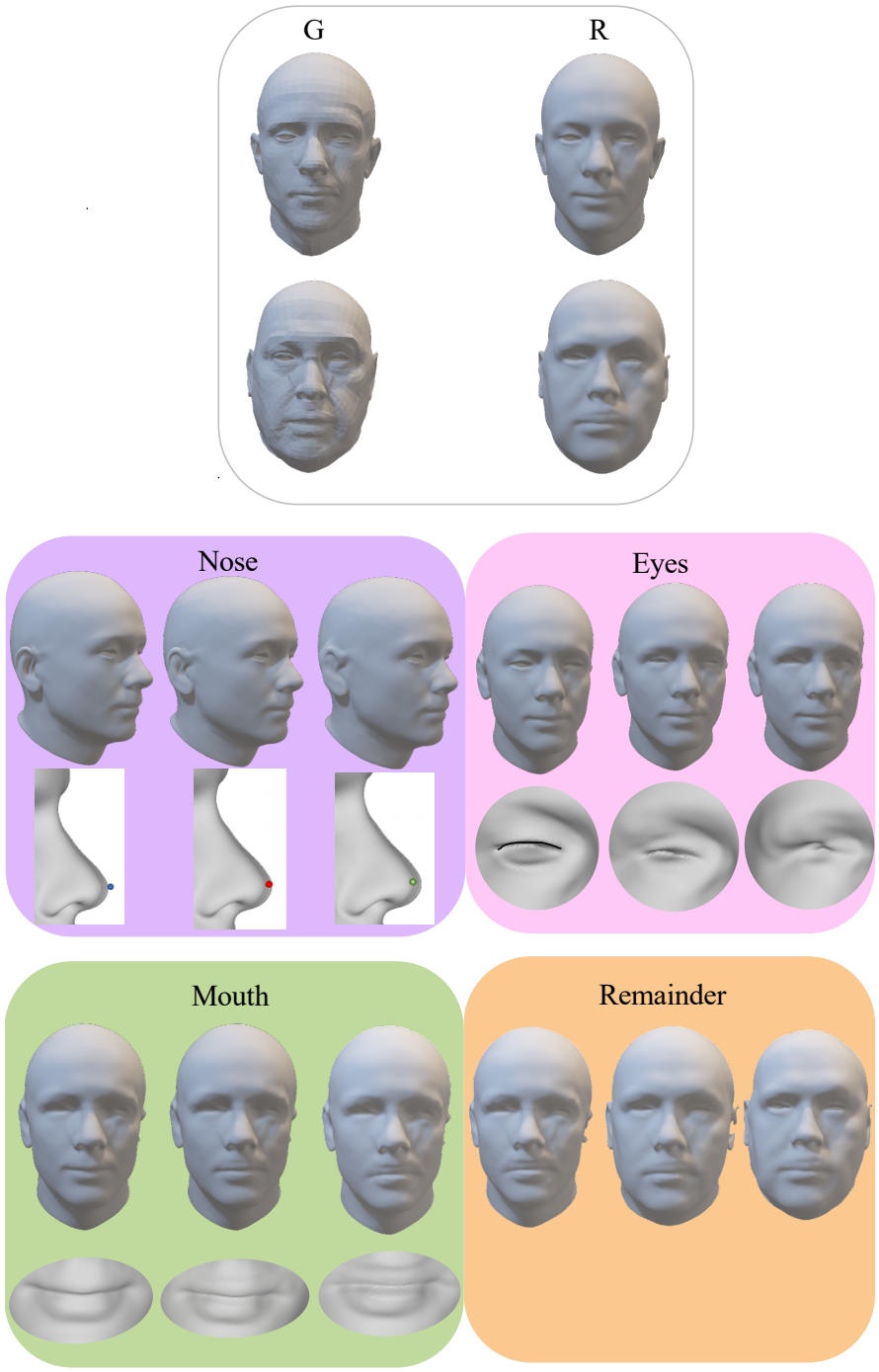

Figure 4.14: Shape reconstruction and parts-based latent representations interpolation for the Headspace dataset. The top box displays four head shapes of two subjects ('A' in the first row, 'B' in the second row), with 'G' for ground truths and 'R' for reconstructions by our network. Coloured blocks show sequential interpolation of facial features (nose, eyes, mouth, remainder) from left (Subject 'A') to right (Subject 'B'). The purple 'Nose' box, for example, illustrates this with a middle column interpolating between 'A' and 'B's nose embeddings. Locally-deformed details are magnified, highlighting nose tip position shifts (blue to green dot via red dot) and differences in nose bridge height in the side view.

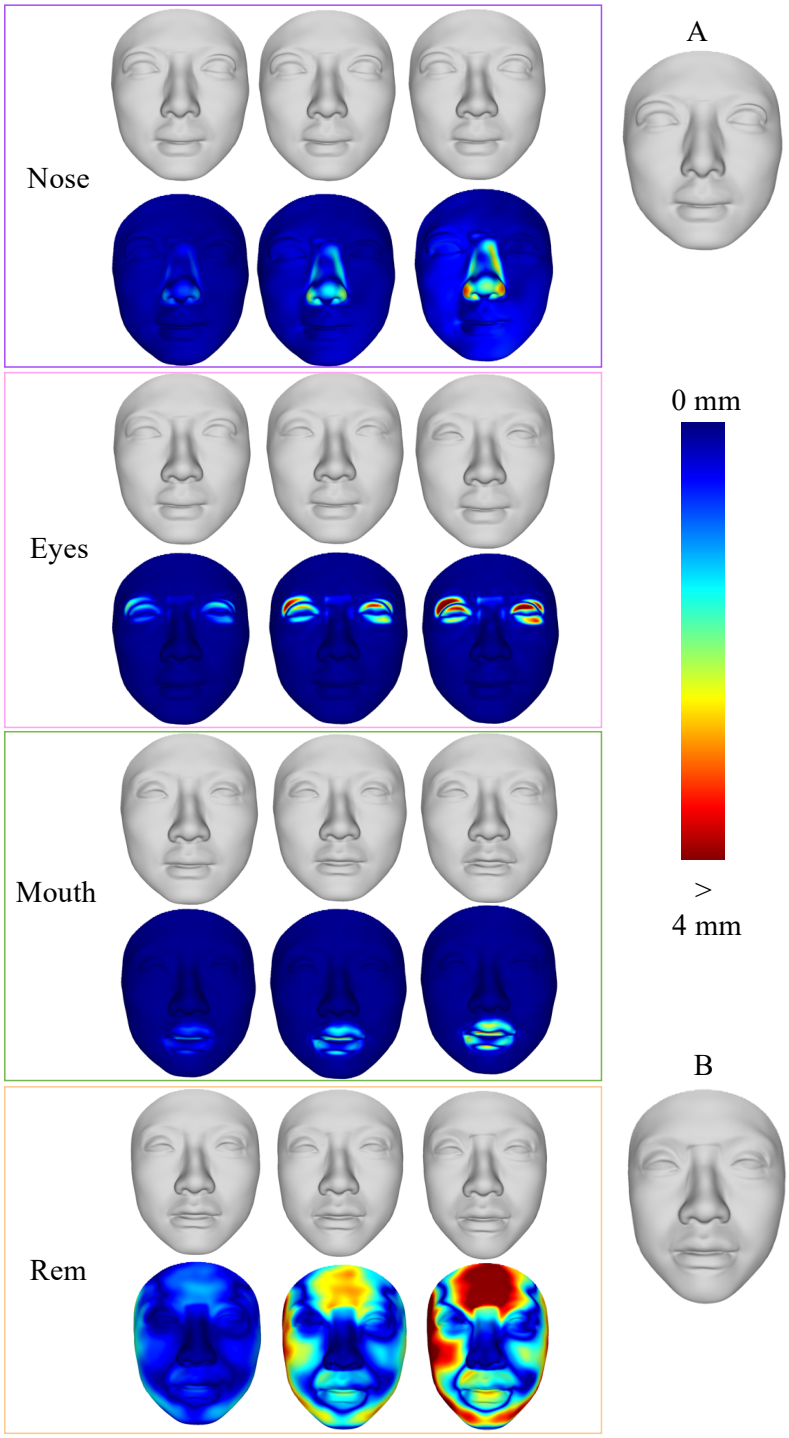

Figure 4.15: Shape reconstruction and interpolation of Parts-based latent representations for two individuals (A and B) in the FaceScape dataset. We perfrom independent interpolations of the latent representations for different facial features, *i.e.*, the nose, eyes, mouth, and remainder ('rem'), deforming from Subject 'A' (top) to Subject 'B' (bottom), which are visually segmented into four groups, each using a corresponding colour box. Within each group, we display the reconstructed and interpolated face shapes in the first row. The second row presents the error maps that represent the per-vertex distance between the current shape and the first shape of the corresponding group. For each initial shape within each group, it is compared with its predecessor.

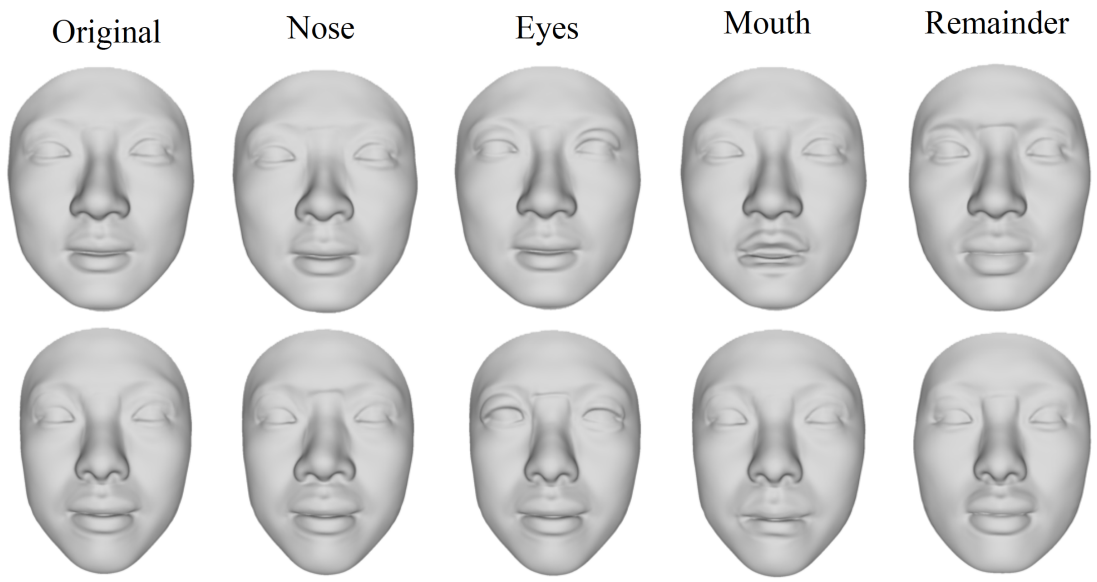

Figure 4.16: Examples of randomly generated faces/parts. The left columns are original, unseen face shapes from the FaceScape dataset. Parts are generated through random Gaussian sampling applied to their corresponding part latent vectors, as illustrated in the 'Nose', 'Eyes', 'Mouth' and 'Rem' columns.

shape within a group, it is compared with its predecessor from the previous group. This demonstrates that the deformation occurs only in the vertices corresponding to the specific part, while the vertices of other parts remain unchanged. It is exemplified in the 'Nose' and 'Eyes' columns of Figure 4.15. For example, in the second row of the 'Nose' column,we observe the nose becoming wider, and in the first row of the 'Eyes' column, the eyelids appear thicker. In both instances, the other parts maintain their original shape.

In our analysis, we calculate the average per-vertex distance between the original face and generated face within specific part regions, as well as across all remaining parts, in the FaceScape dataset. To investigate the effectiveness of our parts-based latent representations, we systematically vary a single part latent variable within the first three dimensions from -0.01 to 0.01, while maintaining zero values in all other dimensions of that specific part, as results shown in Figure 4.18.

By examining the average vertex distance between the two shapes, we focus particularly on the distance from the corresponding part, which is expected to be large, and compare it with the distance from all other remaining parts, which is

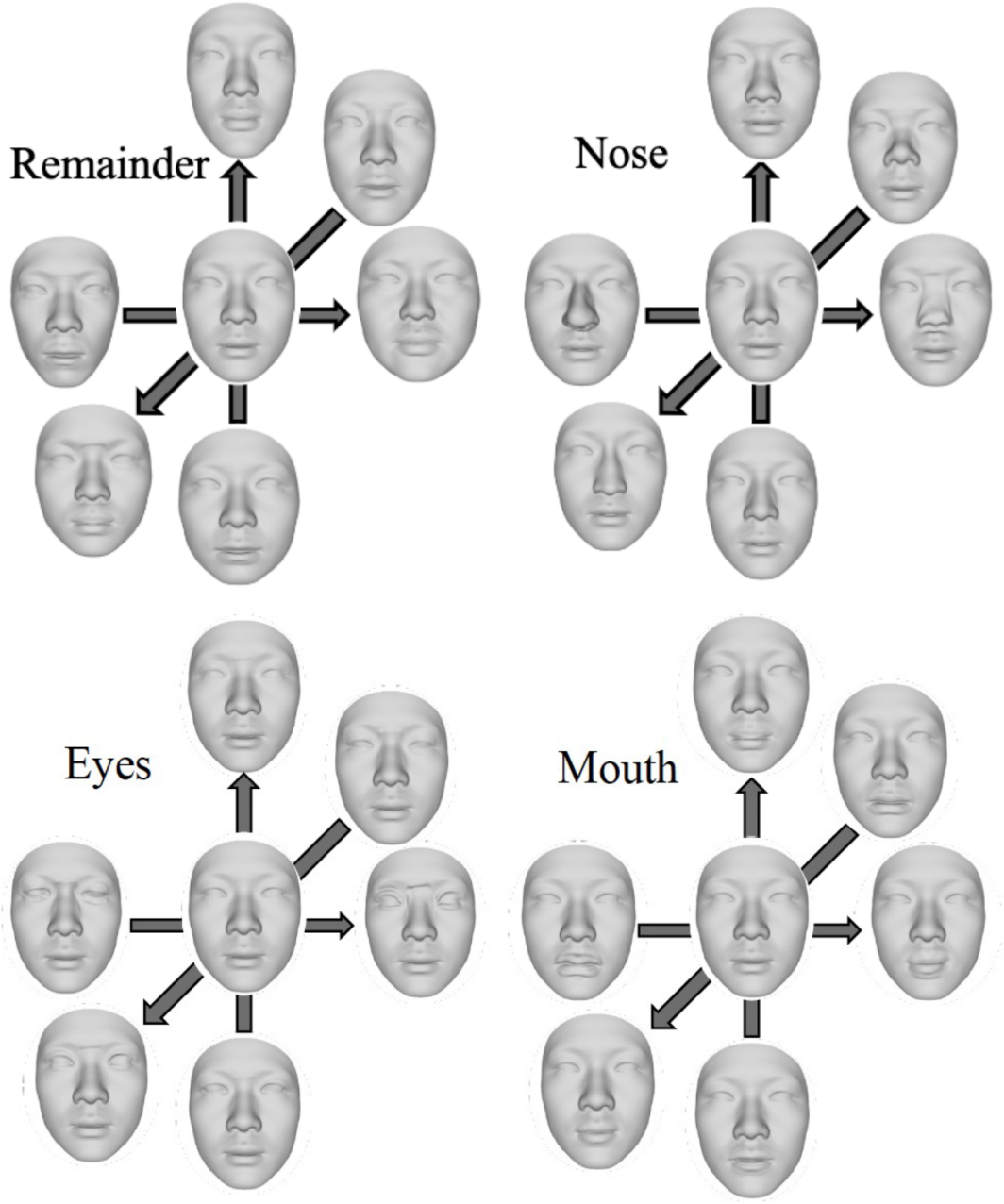

Figure 4.17: Independent control of four facial regions. Top Left: the 'Remainder' part of the face that excludes the nose/eyes/mouth is varied. Top Right: the nose region only is varied. Bottom Left: the eyes region only is varied. Bottom Right: the mouth region only is varied. To achieve this, these part-specific latent embeddings are varied ($\pm 3\sigma$) over their three principal components.

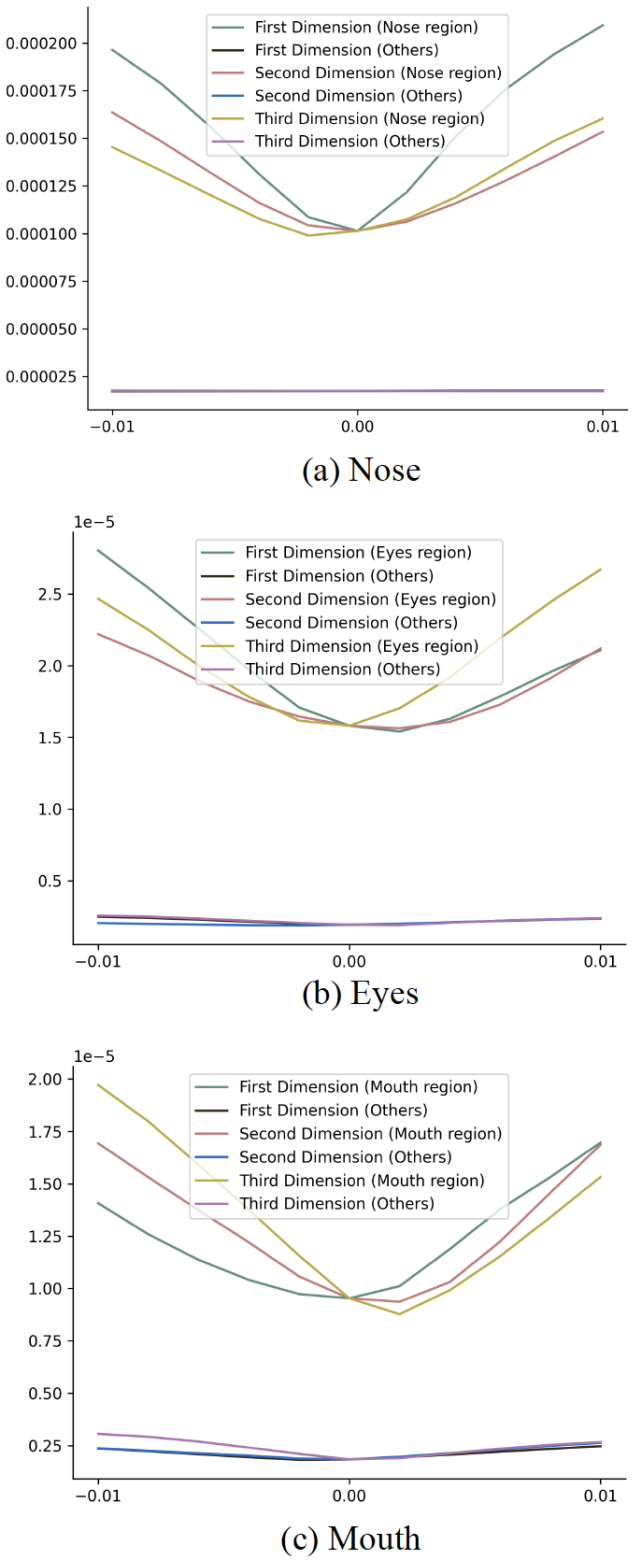

Figure 4.18: Average per-vertex distance during the exploration of each latent variable across the first three dimensions of the corresponding facial parts.

expected to be relatively small. We can discern the impact of altering the latent representation of one specific part while keeping other regions unchanged. This results also prove the efficacy of our parts-based latent representation learning approach. They demonstrate that modifying the latent variables of a particular facial feature leads to significant changes in that feature while leaving the rest of the face largely unaffected.

In addition, we explore new facial feature reconstructions in the FaceScape dataset by randomly sampling values from a standard normal distribution, $\mathcal{N}(0, 1)$, based on their corresponding latent representations, as illustrated in Figure 4.16. We also apply PCA to the latent space of each facial part to explore its primary variation. The first three components for each facial region, representing the most significant variations within the training set, are visualised in Figure 4.17.

4.2.7 Ablation Studies

We conduct experiments to compare landmark selection strategies, specifically contrasting the five original landmarks (nose tip, outer eye corners, and mouth corners) with those employed in our method, as predefined in Figure 4.4 and Figure 4.5. The reconstructed results of the full faces and individual facial parts, *i.e.*, nose, eyes, and mouth, are presented in Table 4.5 for the FaceScape dataset and in Table 4.6 for the Headspace dataset. Additionally, qualitative results are shown in Figure 4.19.

From these results, it is observable that our performance, using landmarks specifically defined for each facial part, surpasses that of the strategy using just five full-face landmarks. The predefined landmarks in our method enhance the ability to accurately capture the fine details of each facial part. As depicted in Figure 4.19, it is evident that the eyes and mouth are nearly disappearing in reconstructions that use only five basic landmarks, highlighting the importance of our comprehensive landmark strategy for detailed facial reconstruction.

We also conduct experiments specifically to evaluate the effectiveness of incorporating the Laplacian deformation step in our process of swapping key facial features, which is important for the smoothness of full-face reconstructions. In Figure 4.20 and Figure 4.21, we present comparisons of faces with swapped features between using only affine transformation and a combination of affine transformation and Laplacian deformation, for both FaceScape and Headspace datasets. It can be demonstrated that the use of Laplacian deformation effectively eliminates the seams at the swapped

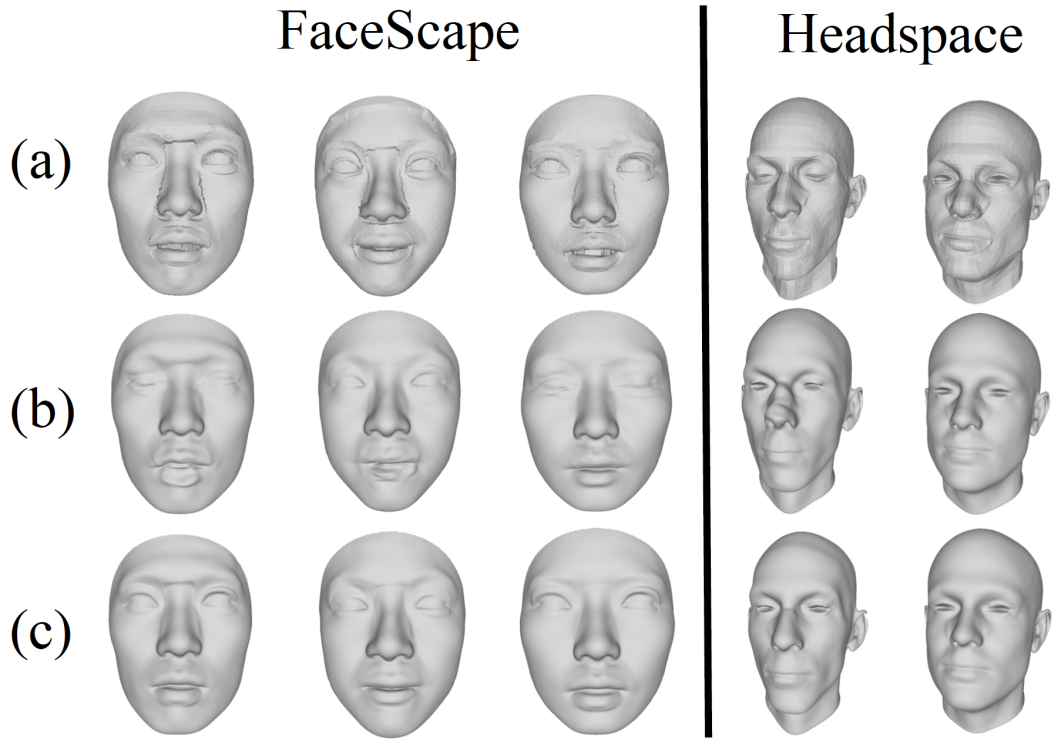

Figure 4.19: Results of 3D face reconstructions using different landmark selection strategies on both FaceScape and Headspace datasets, where different rows display distinct strategies. Row (a) displays the ground truths of the face shapes; row (b) displays the reconstructions with the basic strategy of only five landmarks; and row (c) displays results of our comprehensive landmark selection strategy.

Table 4.5: Results of all 3D face (including both original and those with swapped features) shape reconstruction with different landmark selection strategies on the FaceScape dataset [112, 120].

Part	SCD (mm) ↓		F-Score ↑	
	Ours	Five	Ours	Five
Full Face	**0.5639**	0.5731	**95.09**	94.21
Nose	**0.5919**	0.6133	**89.25**	87.71
Eyes	**0.6093**	0.6608	**87.20**	82.99
Mouth	**0.5887**	0.6525	**86.59**	82.10

Table 4.6: Results of all 3D face (including both original and those with swapped features) shape reconstructions with different landmark selection strategies on the Headspace dataset [24, 122].

Part	SCD (mm) ↓		F-Score ↑	
	Ours	Five	Ours	Five
Full Face	**0.7218**	0.7778	**81.66**	79.91
Nose	**0.6884**	0.7251	**82.89**	79.84
Eyes	**0.6395**	0.6538	**84.72**	83.64
Mouth	**0.5772**	0.5810	88.73	**88.94**

Table 4.7: Results of 3D original neutral face reconstruction with different swapping feature strategies on the FaceScape dataset [112, 120], 'W/o Lap' means without using Laplacian deformation and 'With Lap' means using Laplacian deformation.

Part	SCD (mm) ↓		F-Score ↑	
	W/o Lap	With Lap	W/o Lap	With Lap
Full Face	0.582	**0.533**	93.51	**96.19**
Nose	**0.585**	0.590	**89.99**	89.24
Eyes	0.627	**0.564**	86.40	**90.16**
Mouth	0.623	**0.584**	84.45	**86.69**
Rem	0.581	**0.489**	93.23	**97.37**

junctions. We also conduct a comprehensive comparison of both quantitative and qualitative results for generated neutral faces using these two different swapping feature strategies, in order to demonstrate how effective Laplacian deformation can significantly impact the final results, as presented in Table 4.7 and Figure 4.22 for the FaceScape dataset and in Table 4.8 and Figure 4.23 for the Headspace dataset. (We focus exclusively on reporting the results for neutral faces, as this is sufficient to illustrate the differences between the strategies.)

The PCA analyses, which explore variations within the latent spaces of individual facial parts across both datasets with face shapes processed using Laplacian deformation, are detailed in Appendix A.

4.2.8 LIMITATIONS

While our proposed method is capable of learning both global expression and separate part-based latent representations and this enables independent deformation on each predefined region, human-understandable shape editing and further explainability of

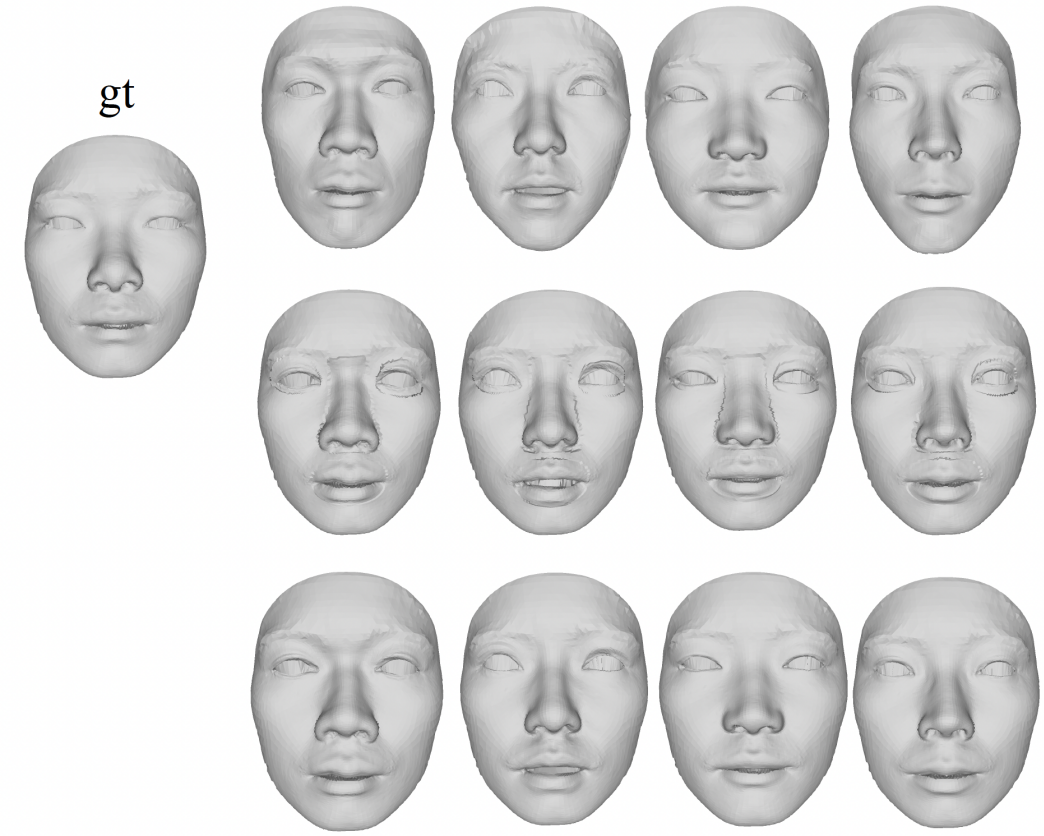

Figure 4.20: Comparison of face meshes with and without Laplacian deformation in the FaceScape dataset. In the leftmost column labelled 'gt', we display the ground truth face with three key facial features ready to be swapped. In the right 4×3 block, the first row shows the ground truth noses, eyes and mouths regions along with their corresponding remainder; the second row depicts the swapping process using only affine transformation; and the third row presents preprocessed faces of combining both affine transformation and Laplacian deformation.

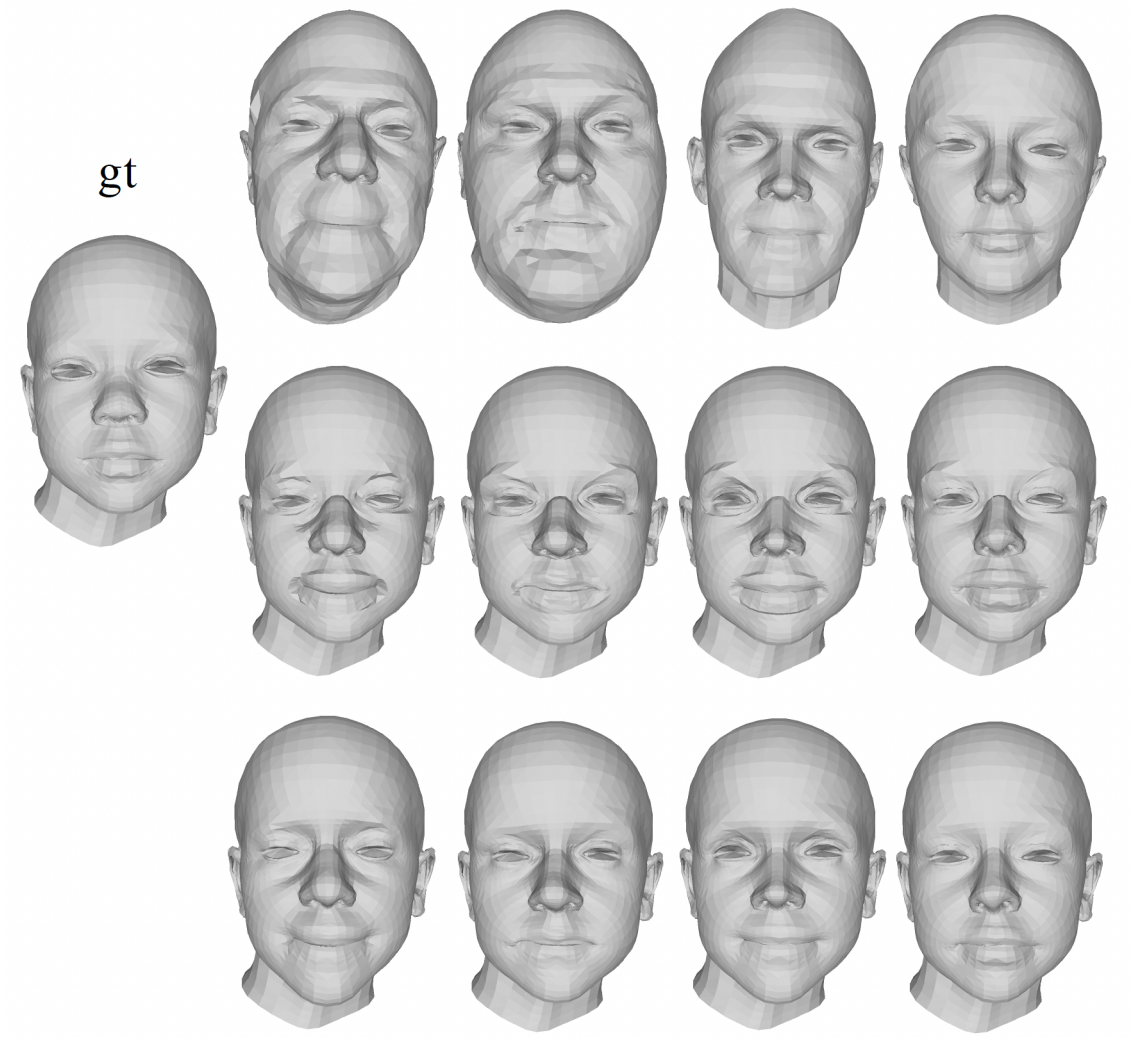

Figure 4.21: Comparison of face meshes with and without Laplacian deformation in the Headspace dataset. In the leftmost column labelled 'gt', we display the ground truth face with three key facial features ready to be swapped. In the right 4×3 block, the first row shows the ground truth noses, eyes and mouths regions along with their corresponding remainder; the second row depicts the swapping process using only affine transformation; and the third row presents preprocessed faces of combining both affine transformation and Laplacian deformation.

Figure 4.22: Generated faces from our model for the FaceScape dataset. The row 'gt' shows the ground truth 3D swapped faces; the row 'w/o' shows 3D generated faces where key facial features are swapped using only affine transformation; and the row 'with' shows 3D generated faces with key facial features swapped using both affine transformation and Laplacian deformation.

Table 4.8: Results of 3D original neutral face reconstruction with different swapping feature strategies on the Headspace dataset [24, 122], 'W/o Lap' means without using Laplacian deformation and 'With Lap' means using Laplacian deformation.

Part	SCD (mm) ↓		F-Score ↑	
	W/o Lap	With Lap	W/o Lap	With Lap
Full Face	0.718	**0.708**	82.03	**83.07**
Nose	0.700	**0.679**	82.47	**84.24**
Eyes	0.646	**0.611**	84.98	**86.98**
Mouth	0.580	**0.561**	88.01	**89.75**
Rem	0.722	**0.710**	81.97	**83.20**

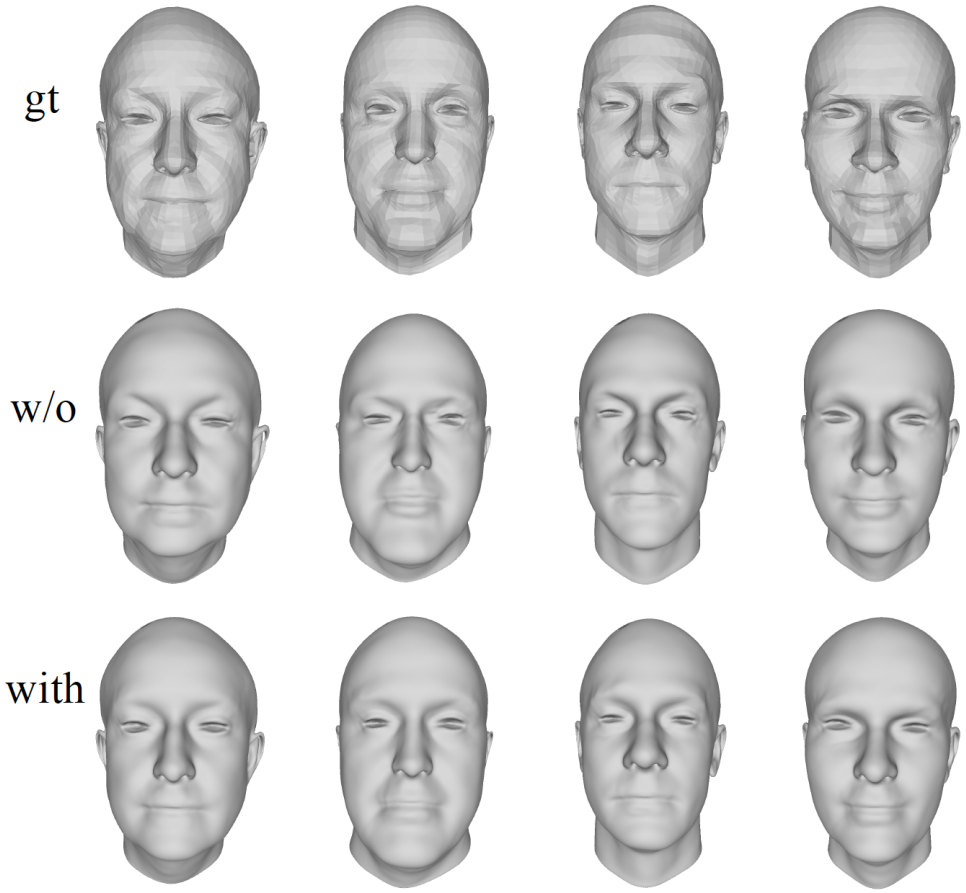

Figure 4.23: Generated faces from our model for the Headspace dataset. The row 'gt' shows the ground truth 3D swapped faces; the row 'w/o' shows 3D generated faces where key facial features are swapped using only affine transformation; and the row 'with' shows 3D generated faces with key facial features swapped using both affine transformation and Laplacian deformation.

the latent spaces requires further work.

Our method focuses on 3D parts-based facial generative modelling, which has the potential to generate new expressions and parts and enables individual modification of each facial part independently to subtly alter identities. We acknowledge that utilising our method may have the potential to maliciously alter digital biometric identities. Secure deployment of systems such as ours is necessary to mitigate these concerns.

4.3 SUMMARY

Unlike previous works that generated new faces based on full-identity latent representations, our approach enables independent control of different facial parts, *i.e.*, nose, mouth, eyes and also the remaining surface and yet generates new faces with high reconstruction quality.

To achieve this, we propose a new approach for facial feature swapping for data augmentation and a parts-based sequential deformation network to learn separate latent embeddings for separate parts. We predefined three key parts of a human face: nose, eyes and mouth - with the remainder of the facial structure (including forehead, chin, cheeks, cranium) grouped together as a fourth part - although, in principle, this 'remainder' part could be further subdivided.

To learn separate part representations, swapped facial features across pairs of subjects using 3D affine mappings to enable data augmentation by applying affine transforms to existing facial part shapes. We then trained a sequence of four submodules - one for each part deformation. All three part features (nose, eyes, mouth) belong to one subject, while the 'remaining' part is from a second subject. To the best of our knowledge, our method is the first to propose latent 3D shape representation learning that is both parts-based and implicit. Our approach fits complex head shapes by part-specific deformation to generate locally-controllable, high-resolution shapes, which demonstrates state-of-the-art performance in face reconstruction and particularly in facial part reconstruction on both FaceScape and Headspace datasets. We further improve our full shape reconstruction results by employing a smoother feature swapping procedure, Laplacian deformation, which effectively reduces curvature discontinuities at the swapped junctions.

In summary, the main contributions are:

- introduction of a parts-based face/head representations that enables separate, localised deformations;
- the ability to generate new facial parts/faces/heads;
- state-of-the-art performance in face reconstruction (cf recent non-parts based approaches).

— 5 —

Critical Comparison of Contributions

Three-dimensional face modelling is an important topic of study for both shape synthesis (3D model parameterisation from 2D images and 3D data, such as point clouds and meshes) and subsequent analysis (face shape understanding) and has the ability to generalise to unseen faces. In Chapters 3 and 4, we presented two methods for 3D face modelling: the first method enables reconstruction based on latent representations of identity and expression, while the second method facilitates face reconstruction by modelling expressions and identities, which can be divided into smaller specific regions, *i.e.*, the nose, eyes, mouth and the other facial features. In this chapter, we provide a comprehensive comparison of these two systems across five main aspects:

1. quality of 3D face reconstruction;
2. quality of 3D face disentanglement, mainly focussing on identity and expression;
3. ability to generalise to new faces;
4. suitability for various face modelling applications;
5. training time and memory requirements.

The structure of this chapter is organised as follows. First, in Section 5.1, we summarise the architectures of our networks, including the PointNet-based VAE-GAN model using point clouds as representations of 3D faces, specialising in 3D facial identity and expression disentanglement, and the SIREN-based deformation networks using continuous signed distance functions (SDFs) as representations of 3D faces, specialising in both 3D identity and expression, as well as parts-based identity disentanglement. Then, in Section 5.2, we compare these two methods in terms of

the above five aspects. Finally, in Section 5.3, we summarise our comparison results, discussing the relative advantages and disadvantages of each system.

5.1 OVERVIEW OF THE TWO SYSTEMS

Our work concentrates on 3D face shape modelling, with a particular emphasis on exploring nonlinear models using deep networks. To enhance the representation of facial components and analyse face shapes more effectively, decoupled latent representations of faces have garnered increasing attention. We proposed two completely different approaches to achieve functionally similar models. The first employs a VAE-GAN architecture, which utilised latent variables to model the probability of observed data based on explicit shape representations, *i.e.*, point clouds. The second involves deformation networks that learn latent variables to drive observed data towards a template shape, which is represented using implicit shape representations, *i.e.*, SDFs.

The simplified network architectures of these two methods are illustrated in Figure 5.1 and Figure 5.2, respectively. Since the first VAE-GAN network focusses only on the disentanglement of facial identity and expression, we omitted the further disentanglement for facial regions in the second deformation networks in our comparison. Instead, we base our comparison on the effectiveness of the latent variables for identities and expressions. Nevertheless, we will discuss the applications of both networks in their complete architectures.

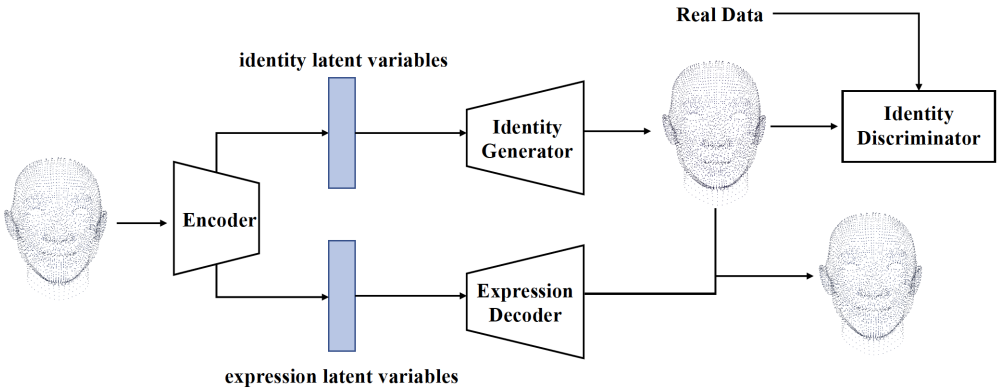

Figure 5.1: The simplified overview of our VAE-GAN model for face identity-expression disentanglement.

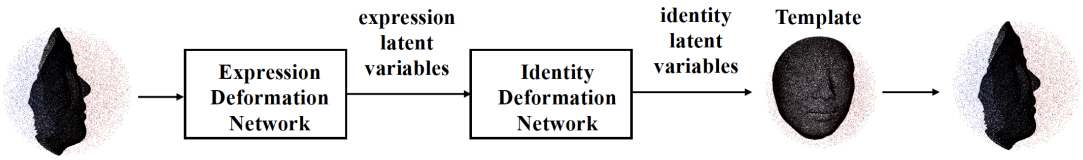

Figure 5.2: The simplified overview of our deformation models (DeforModels) for face identity-expression disentanglement.

As demonstrated in Figure 5.1, the VAE-GAN model employs one encoder and two decoders for identity (generator) and expression respectively. This VAE part of the framework aims to initially disentangle these two components of 3D faces. To ensure that expressions are not included in the identities, the key idea is to employ adversarial learning. This involves constructing a set of real data pairs (sharing same identities) and generated pairs (one generated and one from the real data) to train the identity discriminator.

Conversely, Figure 5.2 introduced a completely different network to achieve the disentanglement of identity and expressions. This network processes the input as SDFs of expressive face shapes. Through an expression deformation network, expressions are neutralised to better align with their corresponding identities. During this expression deformation process, the expressive latent representations are learnt. Similarly to the identity deformation network, the deformed (expression removed *i.e.*, identity) shapes undergo further deformation to align with a template face. During this process, their corresponding identity latent representations are learnt.

Both of these two methods are designed to achieve face identity and expression disentanglement and we utilise a common dataset for evaluation, *i.e.*, the FaceScape dataset [112, 120]. The VAE-GAN model processes point cloud representations of full heads from the FaceScape dataset, whereas DeforModels employs SDFs on cropped, pseudo-watertight faces. We conducted comparative analyses on this dataset to evaluate their performance in 3D face reconstruction, disentanglement, and their ability to generalise to new faces. In the original experiments conducted on these two methods, introduced in Chapter 3 and Chapter 4, respectively, our DeforModels was evaluated using 10 individuals (200 samples), while our VAE-GAN network was tested on a significantly larger scale, involving 5055 samples. However, for our comparative analysis, we adopted 10 subjects from both test sets, choosing those that

are permitted for publication, thus complying with the guidelines of the FaceScape dataset. Note that all comparisons were conducted under the same conditions for both methods.

5.2 COMPARISON RESULTS

5.2.1 3D Face Reconstruction from 3D inputs

In the analysis of 3D face modelling, the evaluation of 3D face reconstruction performance is an important metric. Therefore, to facilitate a thorough comparative analysis between the two methods, we assess their reconstruction abilities using the same test set, following the data preprocessing approach of the DeforModels that involves cropping to a predefined unit sphere. We compare the capability of reconstruction of generated face shapes from 10 individuals, each with 20 expressions, through both methods based on their corresponding learnt latent representations. For the first method (explicit, VAE-GAN), we employed point clouds on full heads and the original reconstructed shapes were full heads, whereas for the second method (implicit, DeforModels) we used cropped faces. Therefore, to ensure a fair comparison in reconstruction performance, we need both methods to employ the same cropped face data. To do this, we crop the full FaceScape head, as described in Chapter 4. To recap, we normalise the data to a unit sphere as follows. First, we set the coordinate origin to a point 4mm behind the nose tip, when the head is pose aligned. We crop to a sphere of 1cm radius and then then downscale this to a radius of 1cm.

To maintain the consistency in our evaluation, we apply the same metrics, *i.e.*, Symmetric Chamfer Distance (SCD) and F-Score, as those used for the DeforModels, as defined in Equation (4.13), since they do not require the same topologies between these cropped faces and those generated by the DeforModels. Unlike previous evaluations, we opted not to sample 150,000 vertices from the generated shapes of these two methods and their ground truths, as our aim is to compare performance between our methods rather than aligning with other literature reviews. Moreover, the number of vertices in the simplified 3D faces used in the first network does not exceed 5000, which was a design decision in order to make the network training times feasible. Therefore, for a fair comparison, we now sample 50,000 vertices from the face meshes generated by both methods and align them with their own ground truths. We present the results in Table 5.1. Further, we illustrate the 3D face shapes of two

Table 5.1: Comparison of the 3D face reconstruction performance in the FaceScape dataset using the first VAE-GAN with explicit representations and the second DeforModels with implicit representations, evaluated based on metrics: SCD and F-Scores.

Model	SCD ↓	F-Score ↑
VAE-GAN	0.9382	64.96
DeforModels	**0.7251**	**88.68**

subjects with seven different expressions (neutral, smile, mouth stretch, anger, chin raiser, lip puckerer and lip funneler), generated using these two methods, as shown in Figure 5.4 and Figure 5.5. For each subject, four expressions are demonstrated, facilitating a side-by-side comparison between both methods for the same set of expressions.

From the analysis presented in Table 5.1, Figure 5.4 and Figure 5.5, we can observe both quantitatively and qualitatively that deformation models based on implicit functions outperform the VAE-GAN architecture using point clouds. There may be two main reasons for this. Firstly, the flexibility in reconstruction resolution using the implicit method results in high resolutions, where there are many vertices on the face meshes, thus enhancing their smoothness, as shown in Figure 5.3. Secondly, in our first network, we implemented a simplification strategy detailed in Chapter 3. To recap, originally, each face mesh comprises 26,317 vertices and 52,261 faces. We apply a quadric-based edge collapse strategy [35], aiming to reduce the mesh to approximately 9000 faces. After the simplification, the face mesh contains 4547 vertices (no more than 5000) and 8999 faces. This preprocessing strategy causes the discontinuity in the faces due to removed vertices and triangles. Although this approach might not yield reconstructions as smooth as those may be achieved with complete data, it significantly reduces memory usage and computational time.

5.2.2 3D Face Disentanglement

In addition to 3D face reconstruction, the capability to effectively disentangle 3D facial identity and expression represents another crucial metric for assessing 3D face modelling. Consequently, we extend our comparative analysis to include the disentanglement performance of our two methods, utilising the same cropped face shapes as in the 3D face reconstruction analysis.

For the disentanglement evaluation detailed in Chapter 3, we employed two

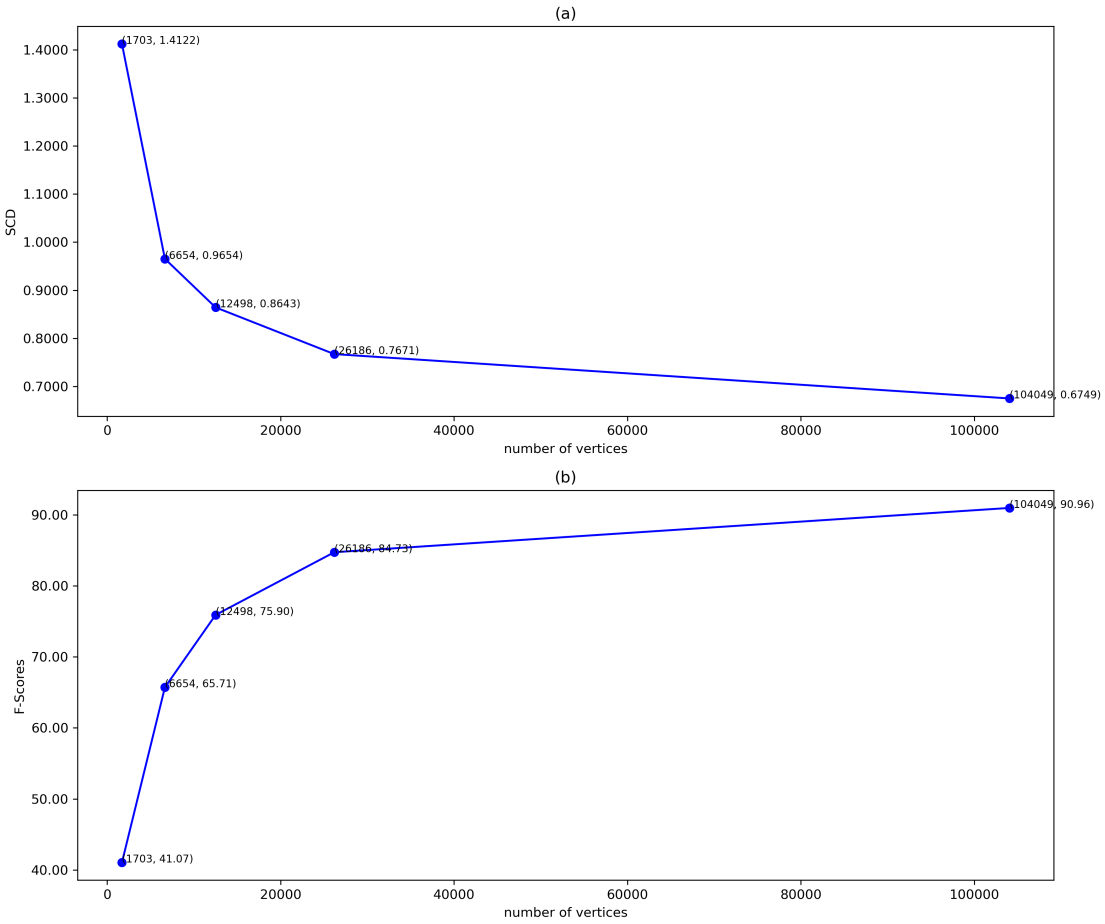

Figure 5.3: Analysis of different resolutions for SCD and F-Scores of one predicted face shape. Sub-figure (a) illustrates the relationship between SCD and the number of vertices (resolution) in one predicted face shape. Sub-figure (b) illustrates the relationship between F-Scores and the number of vertices (resolution) in one predicted face shape.

quantitative metrics. However, due to the variance in topology between the cropped generated facial identity shapes and those reconstructed identity shapes from the DeforModels, even within their respective datasets, directly comparing them using the same metrics poses a challenge. Consequently, we opt for a qualitative comparison. In alignment with the quantitative metrics employed for 3D face disentanglement, we will demonstrate the reconstructed facial identity shapes for each expression of the same individual. This approach enables us to assess the similarity between the generated identity shapes and the ground truth identity, while ensuring that no expressions are presented in each predicted identity shape. Moreover, it facilitates

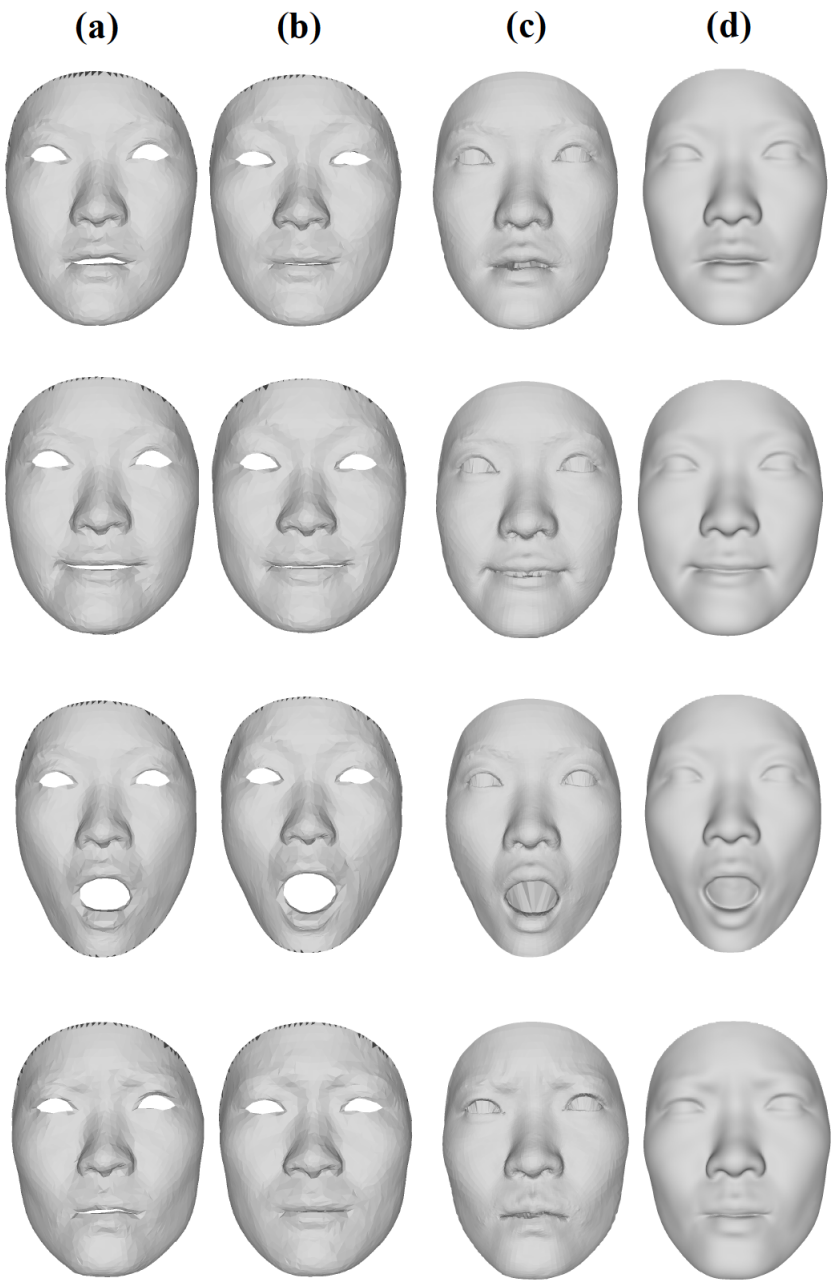

Figure 5.4: 3D expressive face reconstruction comparing the VAE-GAN model with DeforModels for Subject 1. Columns (a) and (b) represent ground truths and generated face shapes from the VAE-GAN model, respectively. Columns (c) and (d) represent ground truths and reconstructions from DeforModels. The rows from first to fourth sequentially represent the expressions: neutral, smile, mouth stretch and anger.

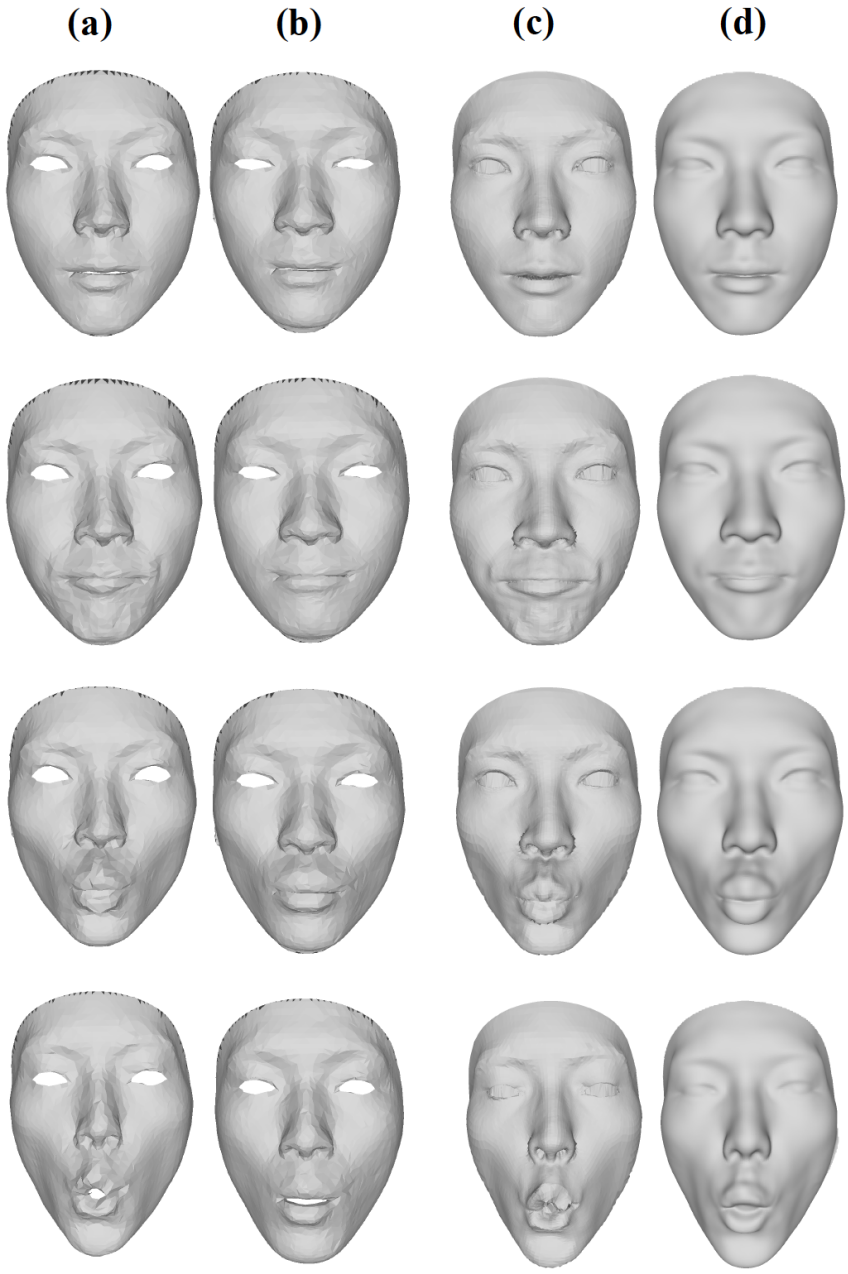

Figure 5.5: 3D expressive face reconstruction comparing the VAE-GAN model with DeforModels for Subject 2. Columns (a) and (b) represent ground truths and generated face shapes from the VAE-GAN model, respectively. Columns (c) and (d) represent ground truths and reconstructions from DeforModels. The rows from first to fourth sequentially represent the expressions: neutral, chin raiser, lip puckerer and lip funneler.

observations of the stability and consistency of the identity across different expressive face shapes from the same individual, comparing these aspects between the two methods.

We select nine distinctly representative expressions from one subject and display their corresponding generated identities. The reconstructed identity shapes using the VAE-GAN model and DeforModels are presented in Figure 5.6.

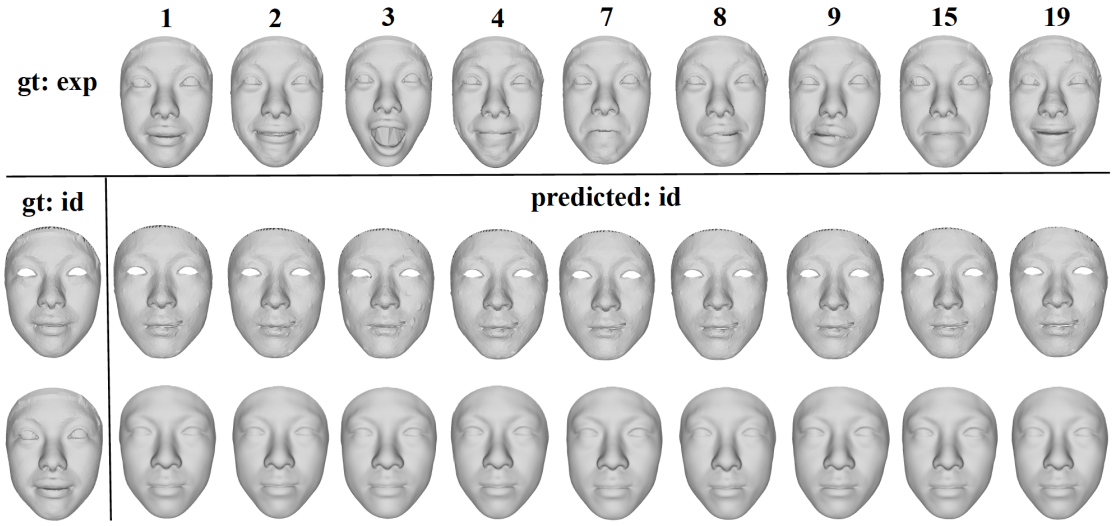

Figure 5.6: Comparison of generated facial identity shapes from one subject using the VAE-GAN network and DeforModels. The first row displays the original expressive ground truths along with their corresponding expression indices for references. The second row presents the disentangled facial identity shapes predicted by the VAE-GAN network, while the third row presents those predicted by DeforModels. Identity ground truths are labelled as 'gt: id'. The generated identity shapes derived from the decoupled identity latent vectors corresponding to various expressive shapes, each labelled as '1', '2', and so forth.

We also illustrate the three principal components of the learnt identity and expression latent representations obtained through the explicit and the implicit methods in Figure 5.7 and Figure 5.8, respectively. These visualisations demonstrate the effectiveness with which each model captures the variations in identity and expression. It is noteworthy that in the PCA analysis conducted with the VAE-GAN, uncropped head shapes are used to present the variations in identity and expression latent representations, as the variations observed in the identity latent embeddings specifically refer to variations in head shape.

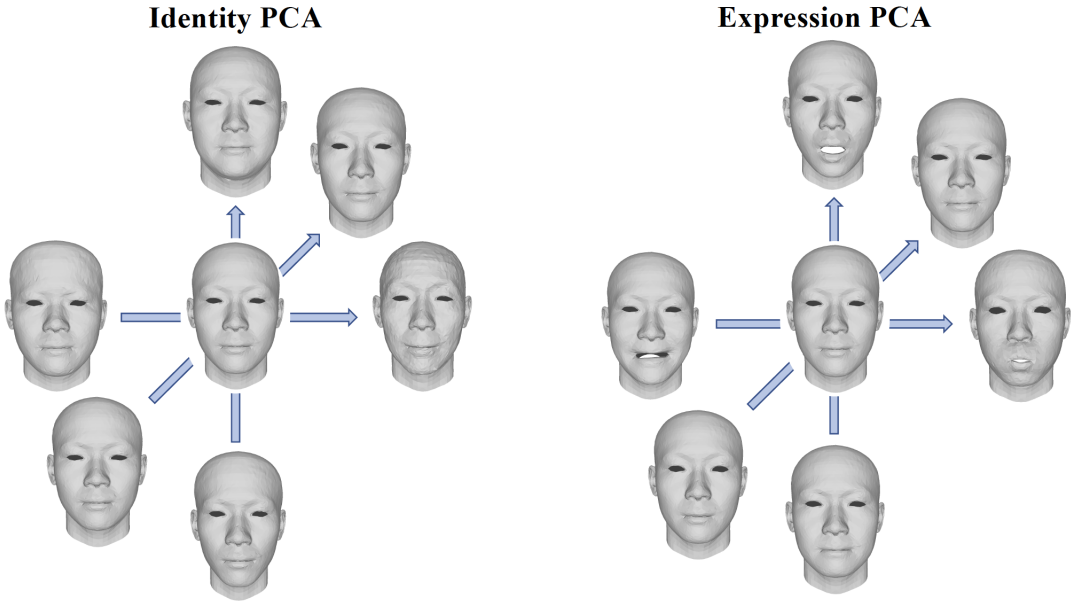

Figure 5.7: Independent variations of identity and expression representations obtained through the VAE-GAN network. These latent representations are varied ($\pm 3\sigma$) over their corresponding three principal components.

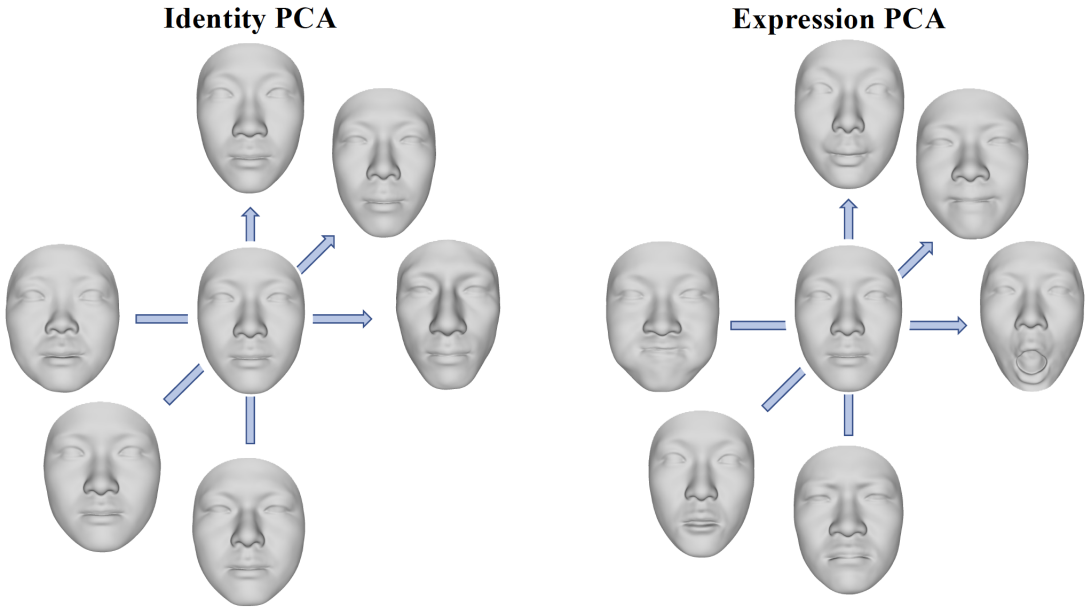

Figure 5.8: Independent variations of identity and expression representations obtained through the DeforModels. These latent representations are varied ($\pm 3\sigma$) over their corresponding three principal components.

From Figure 5.6, it is evident that the identity shapes generated by both of our methods are non-expressive. Observing the reconstructed identity shapes of the same subject, derived from identity latent representations associated with various expressions depicted in these two figures, we note a high degree of similarity among them. This highlights the robustness of our methods in maintaining consistent identity features despite varying expressions.

Simultaneously, from Figure 5.7 and Figure 5.8, which depict the most significant variations along identity and expression latent vectors for both methods, we observe a similar conclusion. When identity varies, the expression remains neutral, consistent with the mean face. Conversely, when expression varies, the identity remains unchanged, also consistent with the mean face. These findings further demonstrate the strong performance of our methods in 3D face disentanglement between identity and expressions.

5.2.3 Ability to Generalise to New Faces

The ability to generalise to unseen face shapes is an important aspect in the evaluation of 3D face models. In our comparative analysis, we particularly focus on the ability of latent representations of facial identity and expressions, learnt by our two methods, to generate unseen faces.

We randomly sample values from a normal distribution, $\mathcal{N}(0, \sigma)$, corresponding to the identity and expression latent embeddings from both methods. The results are shown in Figure 5.9 for the VAE-GAN architecture and in Figure 5.10 for the deformation models. For the VAE-GAN method, we employ full head shapes, as we did for PCA in Section 5.2.2, instead of the cropped face shapes. This is because the learnt latent embeddings are based on 3D full head shapes, thus capturing variations not only in the facial features but also in the overall head shape. In Figure 5.9 and Figure 5.10, we display the mean face on the left side, with other varied face shapes (eight for identity variations and eight for the expression variations) presented in two rows. From these two figures, we observe the varied identities and expressions, which demonstrated that our methods are able to generate new unseen faces across both identity and expression components. However, in faces generated by the VAE-GAN architecture, bumps are observed in the facial identity shapes, but not in the expressive shapes. We will further explore this issue in our future research. The unseen face shapes, generated by sampling within PCA spaces of

latent representations, are depicted in Appendix A.

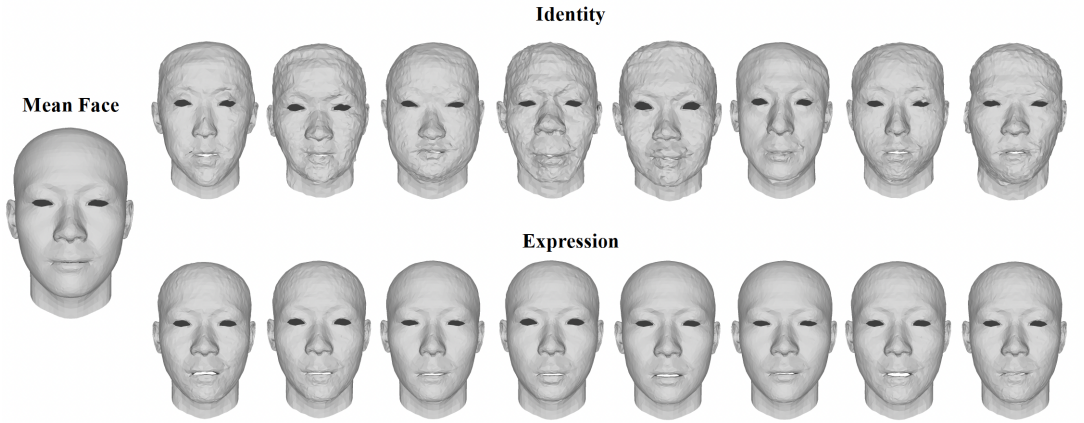

Figure 5.9: Examples of randomly generated facial identity shapes and expressive face shapes from the VAE-GAN model. Facial identities and expressions are generated through random Gaussian sampling applied to their corresponding latent representations, as shown as 'Identity' and 'Expression'. The left column is the mean face shape.

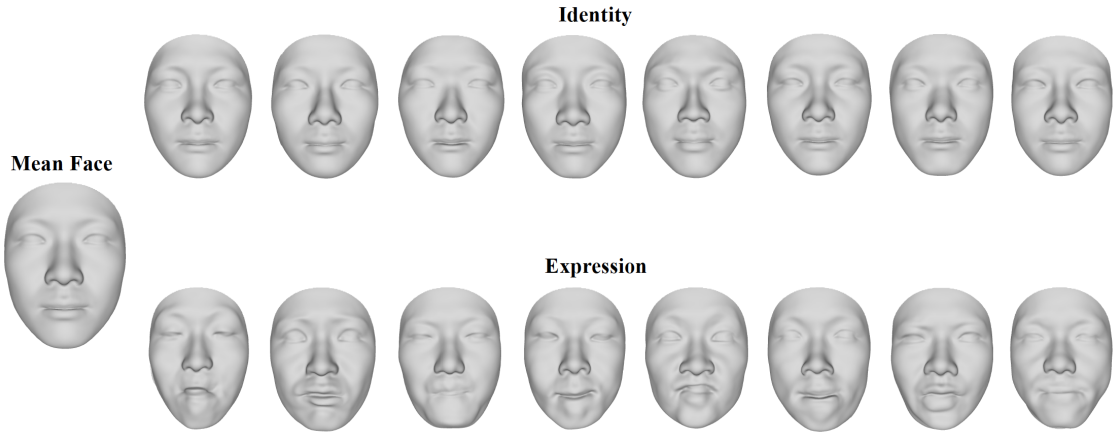

Figure 5.10: Examples of randomly generated facial identity shapes and full expressive face shapes from the DeforModels. Facial identities and expressions are generated through random Gaussian sampling applied to their corresponding latent representations, as shown as 'Identity' and 'Expression'. The left column is the mean face shape.

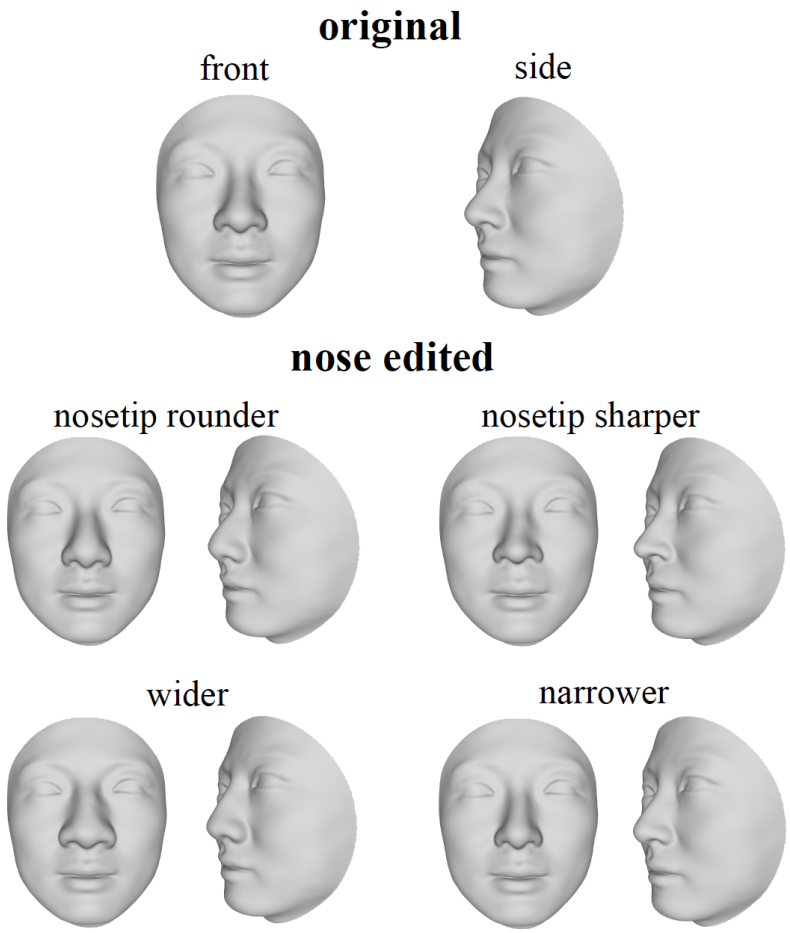

Figure 5.11: Face editing using DeforModels - Nose Region. Two views, frontal and side, are provided to compare the edited faces with the original face. Four specific edited deformations: making the nose tip area rounder or sharper, and adjusting the nose to be wider or narrower.

5.2.4 APPLICATIONS

In Sections 5.2.1, 5.2.2 and 5.2.3, we conducted a comparative analysis focussing on 3D face reconstruction, disentanglement and the ability to generalise to new data for both methods. Given the versatility of robust latent representations can be applied in many applications, such as 3D facial expression transfer and identity/expression latent representations interpolation, this section demonstrates specific applications based on our proposed methods.

Having successfully decoupled identity and expression latent variables, our methods facilitate both expression transfer and interpolation through the learnt latent

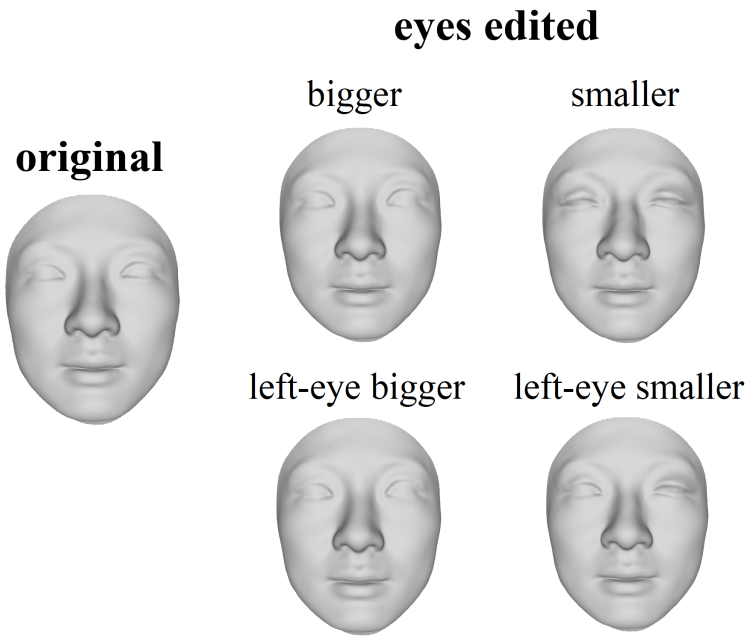

Figure 5.12: Face editing using DeforModels - Eyes Region. Four specific edited deformations are depicted: making both eyes bigger or smaller, or specifically making the left eye bigger or smaller (the left eye remains unchanged).

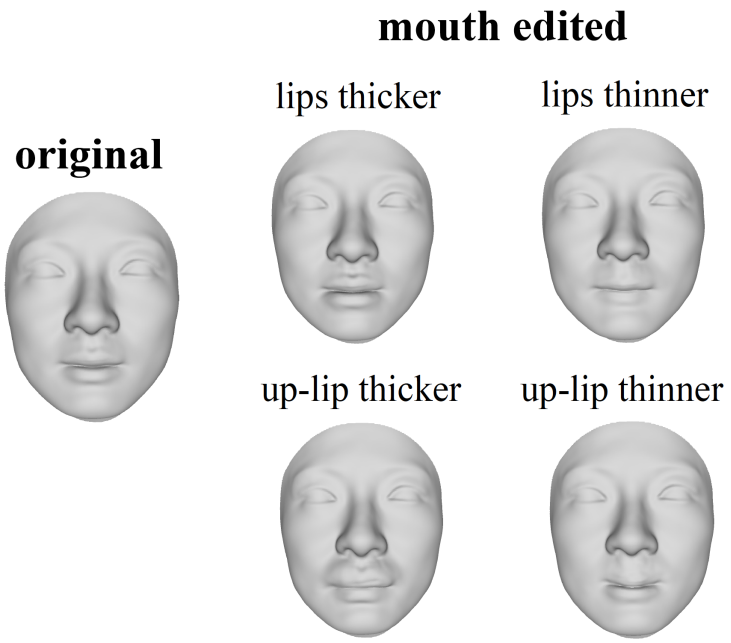

Figure 5.13: Face editing using DeforModels - Mouth Region. Four specific edited deformations are depicted: making mouth lips thicker or thinner, or specifically making the up lip thicker or thinner.

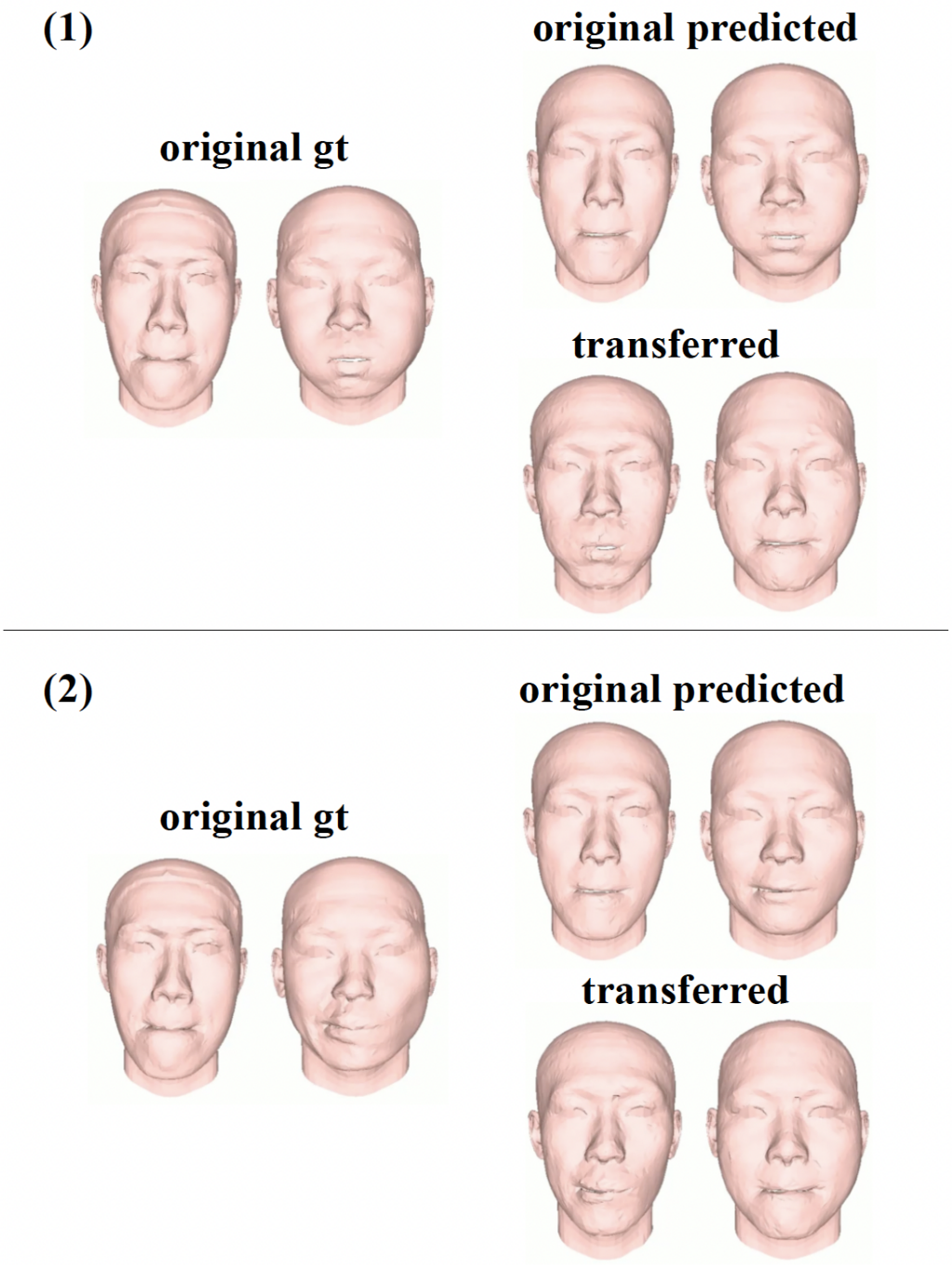

Figure 5.14: Expression transfer using latent representation from the VAE-GAN model. Two sets of expression transfers are shown. The left column shows the original facial ground truths before the expression transfer, and the right column shows predicted original faces and the corresponding faces after the transfer.

embeddings. Although our analysis primarily concentrated on the learning of identity and expression latent variables, our DeforModels also facilitates disentanglement for specific facial regions, *i.e.*, the nose, eyes, mouth and the remainder. This capability for finer region latent representation offers significant potential for face editing, highlighting a promising and necessary application in the field of 3D face modelling. Building upon this, we demonstrate expression transfer using latent representations from the VAE-GAN model, shown in Figure 5.14 and Figure A.5, and editing of specific facial regions based on the learnt facial feature representations from the DeforModels in Figures 5.11, 5.12 and 5.13.

Through our DeforModels, we achieved the disentanglement of facial parts-based identity latent representations, enabling their application in face editing. We predefined certain regions for editing, such as the nose tip, eye corners and lips, along with thresholds to precisely control the directions and magnitude of changes. In our experiment, we opted to edit only one facial feature at a time for the specific face since it simplifies the observation of changes in the edited region while ensuring that other facial region remains unaffected. Once the edited parameters, *e.g.* the direction, area and threshold, are established, we use the original latent representation as a starting point. From there, we optimise it to achieve the desired modifications in the facial features. Examples of the face editing for the nose, eyes, mouth regions are illustrated in Figure 5.11, Figure 5.12 and Figure 5.13, respectively.

5.2.5 Resource Requirements

For our VAE-GAN model, we initially pretrained the identity discriminator for 100 epochs with a batch size of 32 on the FaceScape dataset. The pretraining was implemented with PyTorch and executed on an NVIDIA GeForce RTX 3090 GPU, taking approximately 2 hours. Following this, the joint end-to-end network was trained for 280 epochs, also with a batch size of 32. We conducted the end-to-end network using PyTorch on an NVIDIA A40 system. This end-to-end training process was completed in around 10 hours.

For our DeforModels, we implemented the deformation networks using PyTorch and deployed them on two NVIDIA A40 GPUs. The model was trained using a batch size of 36 and 850 epochs for the FaceScape dataset, and 1000 epochs to fit latent representations. We ran training process for approximately 124 hours on the FaceScape dataset.

The VAE-GAN model indicates efficient training times and memory usage during training and inferences, while DeforModels demonstrates superior accuracy and detail in 3D face reconstruction, albeit with higher resource demands.

5.3 SUMMARY

In this chapter, we present a thorough comparison of our two generative models, *i.e.*, VAE-GAN and DeforModels, for 3D face modelling. This comparative analysis focusses comprehensively on five aspects: quality of 3D face reconstruction, quality of 3D face disentanglement of identity and expression, ability to generalise to new faces and resource requirements, including training time and memory. These are summarised in Table 5.2. Furthermore, we explore their applications, *e.g.* expression transfer and face editing, which significantly contribute to the human understanding and practical utility of 3D face modelling. The applications that can be achieved are listed in Table 5.3.

Table 5.2: Comparison of VAE-GAN and DeforModels across different perspectives. The symbol ✓denotes a superior method.

Perspective	VAE-GAN	DeforModels
3D face reconstruction		✓
3D face disentanglement		✓
Ability to generalise to new faces		✓
Resource efficiency	✓	

Table 5.3: Comparison of Applications between VAE-GAN and DeforModels.

Application	VAE-GAN	DeforModels
Interpolations in facial identity latent space	✓	✓
Interpolations in expression latent space	✓	✓
Interpolations in facial parts latent space	✗	✓
Expression transfer	✓	✓
Face recognition	✓	✓
Face editing	✗	✓

The 3D face shapes reconstructed by our first explicit network, which utilises the VAE-GAN architecture, show less smoothness in comparison to the reconstructed 3D

face shapes from the second implicit network. However, it is more resource-efficient, requiring less time and memory during training on the FaceScape dataset. Both generative models demonstrate strong performance in 3D face disentanglement, with their learnt latent spaces capable of generating new data. This effectiveness proves that the designs for adversarial learning and deformation from expressive to non-expressive shapes are both successful in decoupling latent embeddings.

Our second network, *i.e.*, DeforModels, performs well on more diverse applications. It is specifically designed for the disentanglement of not only identity and expression but also different facial region features. This approach significantly improves the controllability of face generation. Given that implicit representations provide greater flexibility for generating 3D shapes, the reconstructed shapes from DeforModels using SDFs are smoother and preserve finer details, as evidenced.

— 6 —

Conclusions

In this chapter, we summarise the key research contributions in 3D face modelling, particular in improving the controllability and explainability of models. We will highlight the significance of our research and its potential applications. Additionally, we will discuss the limitations of our studies and explore directions for future work.

6.1 SUMMARY OF CONTRIBUTIONS

In the introduction, we explained the traditional and recent techniques used in constructing 3DMMs and proposed our two primary research objectives and questions. The first objective was to achieve the disentanglement of 3D face identity and expressions using explicit point cloud representation, and the second was to learn parts-based latent representations for 3D face shapes with a core use of implicit shape representation (SDF) in the model. Both objectives aimed to model 3D faces and to enhance both the explainability and controllability of face models.

In the literature review chapter, we methodically introduced relevant research work across four perspectives. We started with representations of 3D shapes, the foundation of our work. Subsequently, we introduced the core of our research, *i.e.*, 3DMMs, the deep learning algorithms employed to build 3DMMs, and recent studies focussing on the development of explainable latent spaces for 3D face modelling.

Based on our two research questions, we presented two main technical chapters, as well as a shorter chapter for comparisons on both 3D face models. The main contributions will be detailed in the following subsections.

6.1.1 Disentangling Identity and Expressions

We have proposed an end-to-end deep learning network that disentangles 3D face identity and expressions, which can be applied in different scenarios. In detail, we employed a VAE-GAN framework that employs one encoder and two decoders to separately learn latent representations for face identity and expressions, as our initial step toward disentanglement. To ensure the independence and separation of these two latent representations, we introduced an adversarial learning approach that incorporates an identity discriminator. This discriminator is effective in scenarios with both known or unknown identity ground truths (neutral expressions). The principal concept behind our identity discriminator differs from traditional designs that categorise predicted identities as real or fake, based on reference identity ground truths. Instead, our identity discriminator is pretrained without requiring these ground truths, using pairs of expressive shapes as inputs. It determines whether the input pairs share the same identity. In this context, face pairs with the same identity are classified as real samples, whereas those with different identities are determined as fake. During the end-to-end training process, in combination with the generator, our identity discriminator ensures the predicted identity shapes preserve common information between face pairs. Specifically, for pairs sharing the same identity, this shared feature is regarded as the neutral, inherent to their shared identity. Our method was evaluated on three datasets, *i.e.*, the CoMA, BU3DFE and FaceScape dataset, under the same experimental conditions. These included scenarios both with and without available identity ground truths to ensure a fair comparison, showing comparative performance. We also conducted ablation studies to demonstrate the effectiveness of our identity discriminator, particularly in cases where facial identity ground truths are unavailable.

6.1.2 Parts-based face Modelling

We proposed a deformation network designed for modelling 3D face shapes by independently learning facial parts-based latent representations. Our architecture is implemented using implicit representation of 3D face shapes, *i.e.*, SDFs, distinguishing our network from other methods focussed on learning 3D facial parts. We predefined three semantic facial regions, *i.e.*, the nose, eyes and mouth, and employed a facial feature swapping strategy for data augmentation. To address the

issue of discontinuities at the border of swapped regions that arose during the data augmentation, we further explored the implementation of the Laplacian blending to seamlessly integrate these parts. In our deformation model, we employed a series of sequential neural networks to facilitate deformations, including a nose net, eyes net, mouth net, remaining part net and a template net. A template face, derived from the training data, is regarded as the reference target for deformation. For each facial part net, the input face shape is transformed to align with the corresponding part of the template face. In other words, after processing through a specific facial part net, the unique nature of that part on the input face is modified to stay consistent with the template part. During this process, the latent representations corresponding to each facial part are estimated. Moreover, we utilised facial landmarks to capture finer details in the learning of facial parts and facilitate adaptive blending of facial regions for a more natural face. Notably, the template net in our deformation models computes the SDFs for each deformed template face. Our sequential deformation model is designed for the independent learning of facial parts under their neutral expressions, as well as for the learning of latent representations for facial expressions through a expression net. This network successfully achieves the disentanglement of expression from identity. Our experimental evaluation was carried out on both 3D face shapes in the FaceScape dataset and 3D head shapes in the Headspace dataset. The strong performance observed in both datasets validates our model's efficacy in learning latent representations for expressions and facial parts.

6.1.3 Comparative Analysis of 3D Face Models

We conducted a comparative analysis of the two 3D face models outlined in our contributions, focussing on their capabilities to disentangle face identity and expressions. This comparison was structured across five key aspects: 3D face reconstruction, 3D face identity and expression disentanglement, the ability to generalise to unseen 3D face shapes, applications of 3D face models, and training time and memory requirements. Both face models demonstrate exhibit strong performance in reconstruction, disentanglement and generalising to new faces. Particularly, the face model based on implicit representations outperforms the explicit one due to its flexible resolution selection, resulting in reconstructed face shapes with superior high-level details and smoothness. Furthermore, both learnt latent representations that can be applied in various applications, enhancing the explainability and controllability of the generated

face shapes by modifying the relative latent representations of facial identity parts or expressions.

6.2 LIMITATIONS

We introduced two deep learning networks for 3D face modelling, significantly enhancing the models' explainability and controllability. To the best of our knowledge, our research uniquely addresses the scenario where facial identity ground truths are not available. Furthermore, we are the first to employ SDFs as representations for 3D face modelling, where the learning of parts-based latent representations is enabled. Moreover, unlike prior work using explicit representations for disentangling facial parts, our framework is the first to achieve the disentanglement of both facial identity parts and facial identity and expressions.

Despite these advancements, a common challenge in our initial work with a VAE-GAN framework is maintaining balanced training between the identity and expression generators and the identity discriminator. It is non-trivial to determine the optimal parameters for balancing the adversarial process. Our identity discriminator is primarily designed for scenarios lacking identity ground truths. Thus, when identity ground truths are available, the identity decoder's generation process is strongly supervised through the loss functions between the identity ground truths and the predicted facial identities. This strong supervision can potentially mitigate the need for an additional discriminator, which slightly affects the reconstruction results.

Furthermore, in our second research work, the incorporation of a sequential deformation network significantly increases the required training time. Additionally, the data preprocessing for implicit representations necessitates the watertightness of 3D shapes. This requirement presents a challenge and extends the preprocessing time needed to close all holes in the raw shapes.

6.3 FUTURE WORK

In this section, we discuss potential directions of our future work aimed at improving the explainability and controllability of 3D face modelling. These directions are further extensions and investigations based on our current 3D face models.

Parallel Deformation Networks

To significantly reduce the training time of our deformation network, one promising direction for 3D parts-based face modelling is the exploration and implementation of parallel deformation networks. In this approach, neural networks dedicated to different facial regions could be configured in parallel within a single framework. This means the network simultaneously deforms each part of input face. Additionally, a novel blending field network could be proposed to selectively integrate only the deformed parts that match their corresponding template regions, thereby fusing a natural template face. While the core functionality of the deformed neural networks remains unchanged, their manner shifts from sequential to parallel processing. Moreover, the potential to develop a new blending network for the efficient fusion of facial parts can be further investigated.

Explainable Face Editing

Although we independently learn latent representations for 3D facial parts and features (identity and expression), achieving human-understandable latent representations remains an open challenge. We could design a network that facilitates a bi-directional mapping between our learnt latent space and the PCA space. This would enable direct face editing through manipulation of the most significant variances within the PCA space, thereby transforming the non-linearly learned latent space in a more interpretable manner. Alternatively, we could define human-understandable measurement metrics and develop a mapping network that connects our latent spaces with these metrics. It facilitates a more intuitive manipulation process.

Two Models Combination

We could explore the integration of the networks from two face models. This combination aims to leverage the strengths of each model. For example, the GAN architecture achieves disentanglement of identity and expressions under different scenarios. Once we successfully disentangle the facial identities, we could then employ the deformation network to further achieve disentanglement based on parts-based latent codes. This combination may decrease the overall training time, offering an efficiency improvement in modelling.

— A —

Appendix

A.1 PCA ANALYSES ON FACIAL PARTS

Similar to the analysis presented in Figure 4.17, we perform PCA on the latent spaces of facial parts, derived from swapping facial features using Laplacian deformation techniques, to observe their principal variations. As illustrated in Figure A.1 for the FaceScape dataset, no seams and discontinuities are apparent on the the principally varied face shapes, especially in the nose and eyes regions when compared with those in Figure 4.17. Moreover, variations are confined to specific facial parts, as shown in Figure A.1 and Figure A.2 for the FaceScape and Headspace dataset, respectively. This not only demonstrates the effectiveness of our proposed method in learning independent latent representations for facial parts but also highlights how the Laplacian deformation technique contributes to reconstructing smooth faces.

A.2 GENERALISATION TO UNSEEN FACE SHAPES

In Section 5.2.3, we illustrate the ability to generalise new faces by randomly sampling values from a normal distribution, $\mathcal{N}(0, \sigma)$, for the identity and expression latent embeddings learnt from both VAE-GAN model and deforModels. We further perform randomly sampling within the PCA spaces of latent embeddings for identity and expression, following a normal distribution, $\mathcal{N}(0, \sigma)$, derived from these two methods. The PCA components utilised for this analysis are detailed in Section 5.2.2. The generated unseen face shapes are depicted in Figure A.3 for the VAE-GAN architecture and in Figure A.4 for the deforModels.

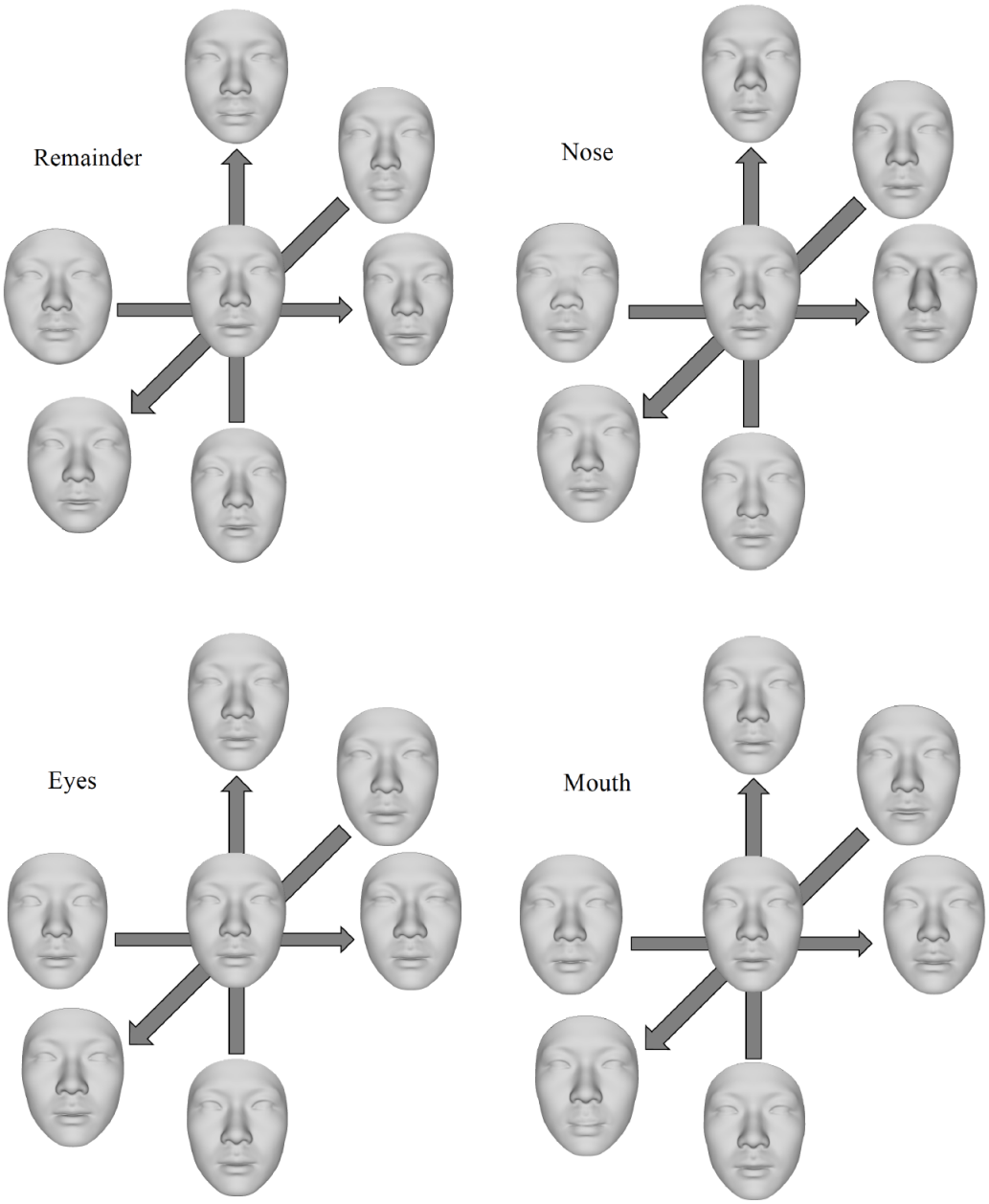

Figure A.1: Independent control of four facial regions for the FaceScape dataset. Top Left: the 'Remainder' part of the face that excludes the nose/eyes/mouth is varied. Top Right: the nose region only is varied. Bottom Left: the eyes region only is varied. Bottom Right: the mouth region only is varied. To achieve this, these part-specific latent embeddings are varied ($\pm 3\sigma$) over their three principal components.

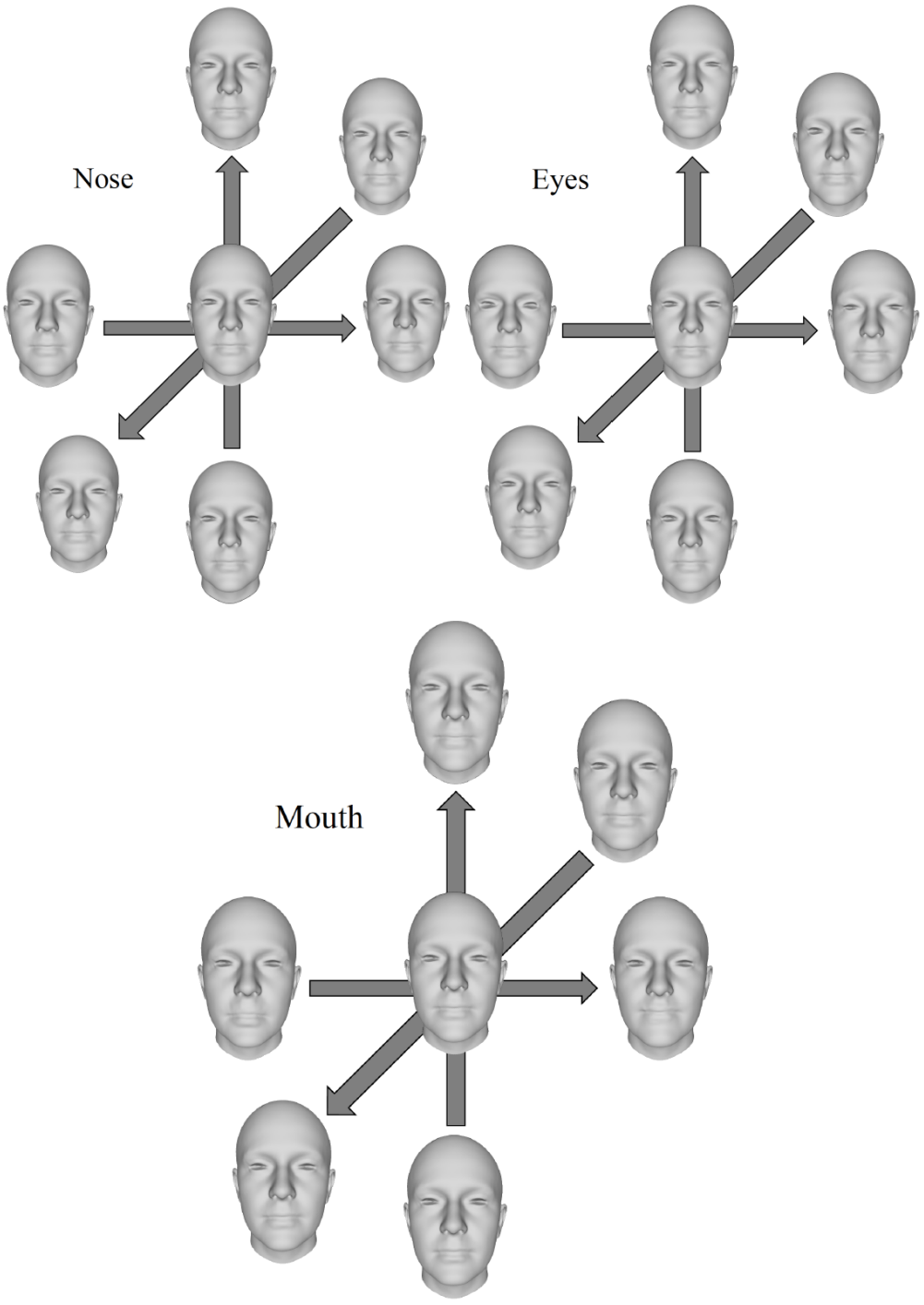

Figure A.2: Independent control of three facial regions for the Headspace dataset. The 'Nose' region only is varied. The 'Eyes' region only is varied. The 'Mouth' region only is varied. To achieve this, these part-specific latent embeddings are varied ($\pm 3\sigma$) over their three principal components.

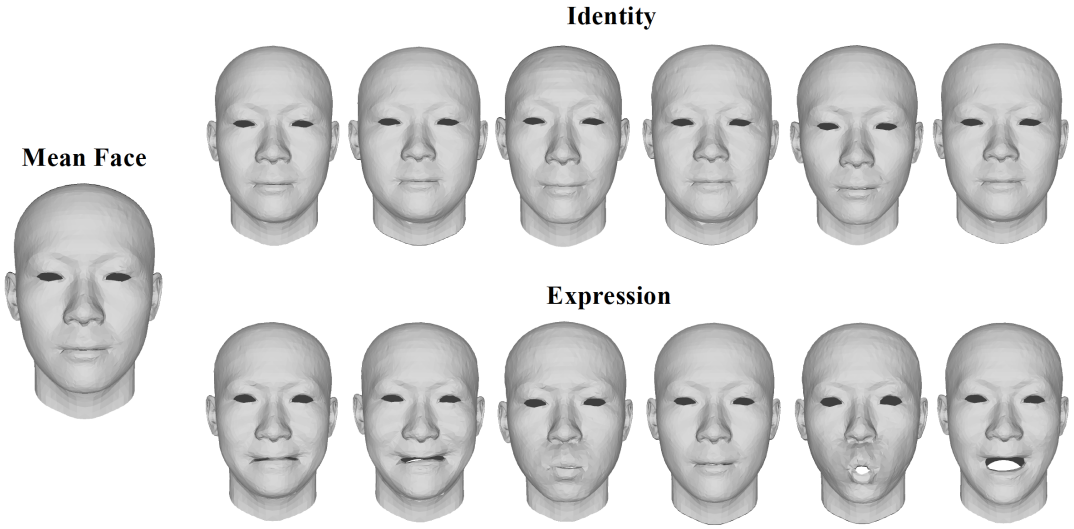

Figure A.3: Examples of facial identity and expressive shapes generated by random sampling within PCA spaces of their respective latent representations using the VAE-GAN model.

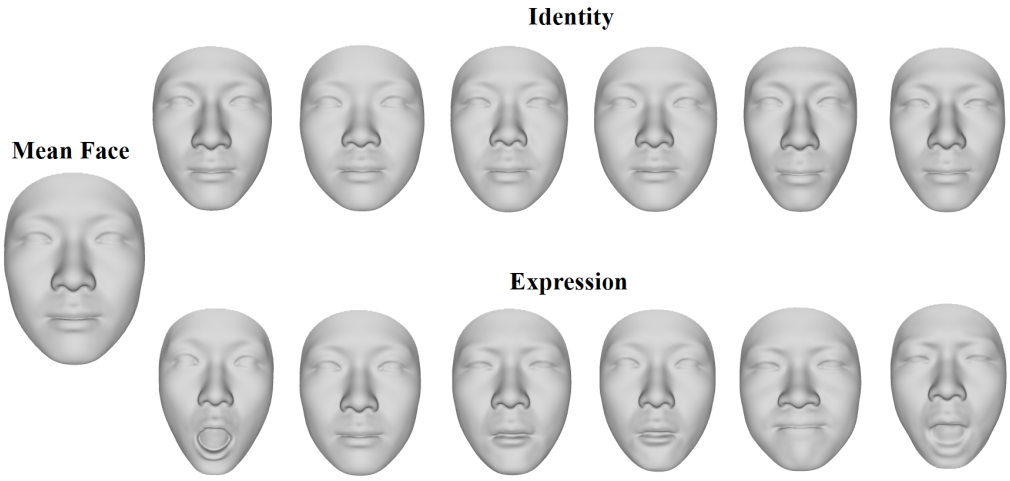

Figure A.4: Examples of facial identity and expressive shapes generated by random sampling within PCA spaces of their respective latent representations using the deforModels.

A.3 ADDITIONAL EXAMPLES OF APPLICATIONS

In Section 5.2.4, we demonstrate the application, *i.e.*, expression transfer, achieved by decoupled identity and expression latent representations from our VAE-GAN model. We illustrate additional examples of expression transfer in Figure A.5 as the application of the VAE-GAN model.

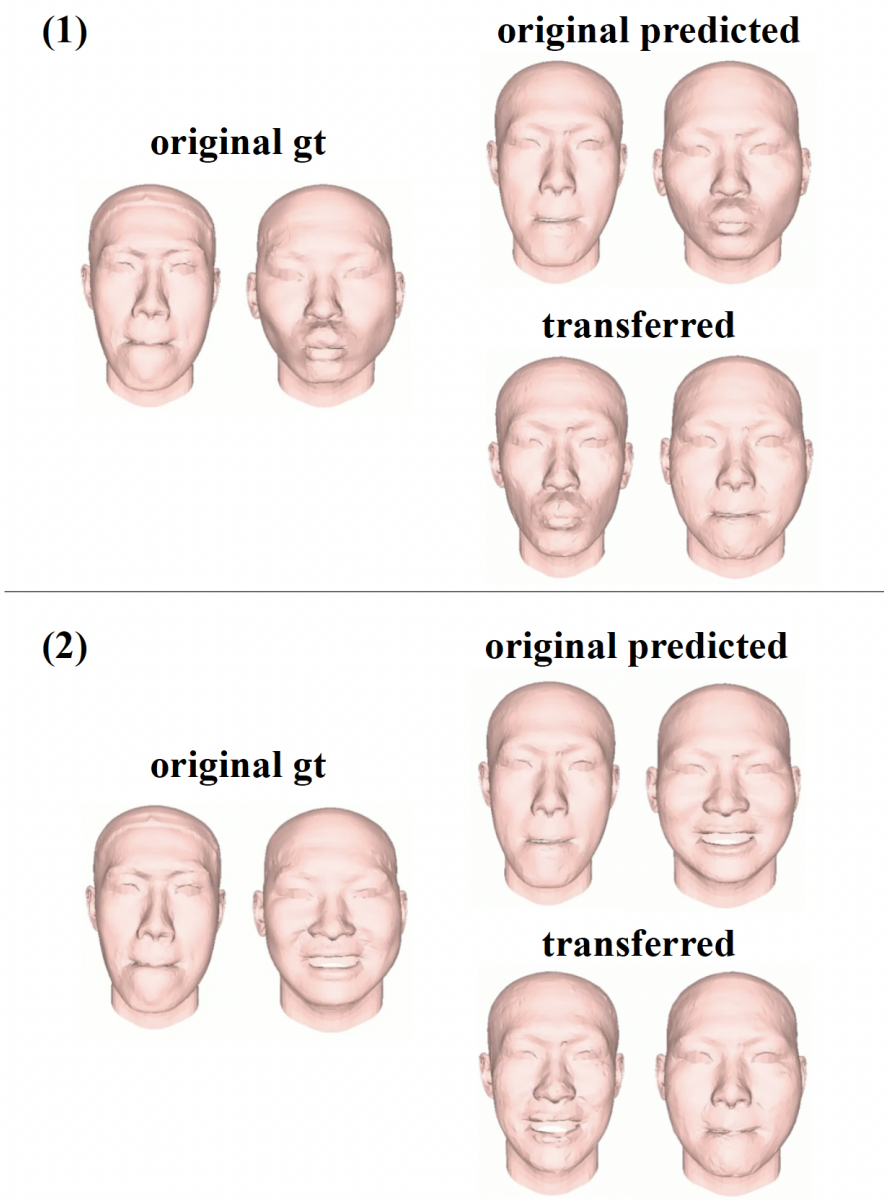

Figure A.5: Additional examples of expression transfer using latent representation from the VAE-GAN model.

— B —

Appendix: Network Architectures

The detailed architecture for our VAE model, introduced in Chapter 3, is shown in Figure B.1. This VAE, inspired by PointNet [86], is designed to learn the distributions of identity and expression, and to sample their latent representations $\mathbf{z}_{\text{id}}$ and $\mathbf{z}_{\text{exp}}$ respectively. Two decoders are employed separately: one to reconstruct the identity shape and the other to reconstruct the expression deformation, based on their corresponding latent vectors.

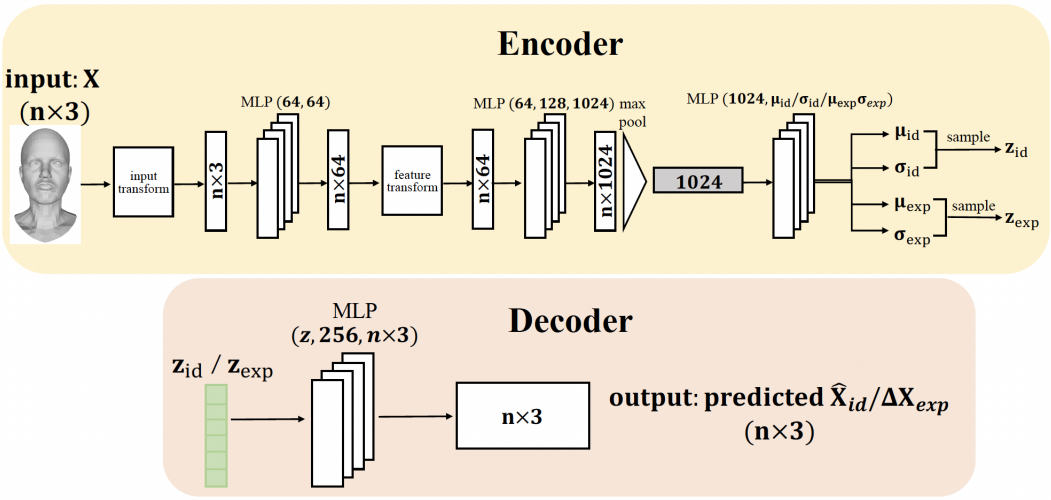

Figure B.1: Detailed pipeline of the VAE model. The encoder follows a PointNet-based architecture, outputting distributions for sampling latent vectors representing facial identity and expression deformation. The decoder, shared between the identity and expression branches, utilises an MLP with 256 hidden neurons to generate predicted face shapes.

Table B.1: Architecture of the Deform-Nets and SDFNet in the Deformation Networks.

Module	Layer	In-Feature	Out-Feature
Expression Deform-Net	Linear Sine	15	128
	Linear Sine	128	128
	Linear Sine	128	128
	Linear Sine	128	128
	Linear	128	15
Nose/Eyes/Mouth Deform-Net	Linear Sine	15	48
	Linear Sine	48	48
	Linear Sine	48	48
	Linear Sine	48	48
	Linear	48	15
Remainder Deform-Net	Linear Sine	15	112
	Linear Sine	112	112
	Linear Sine	112	112
	Linear Sine	112	112
	Linear	112	20
SDFNet	Linear Sine	3	112
	Linear Sine	112	112
	Linear Sine	112	112
	Linear Sine	112	112
	Linear	112	1

The detailed architectures for the implementation of our deformation networks, as previously discussed in Chapter 4, are presented below.

Our networks are composed of five deformation modules and one module specifically designed for predicting SDFs of a template face. The architecture for both the deformation module (Deform-Net) and the SDFs prediction module (SDFNet) consists of five fully connected layers. Except for the output layer, each of these layers is followed by a sine activation function. We base our design on the SIREN architecture [91] that leverages hyperparameters networks to optimise latent representations and predict weights for each deformation module. For the landmark generation network, we utilise the model proposed in ImFace [117]. Additionally, we detail the network for blending facial parts into a whole face. The comprehensive architectures of our networks are listed in Tables B.1 to B.4.

Table B.2: Architecture of the Hyper-Net in the Deformation Networks. $N_{w_{Exp-Deform}}$, $N_{w_{N/E/M-Deform}}$ and $N_{w_{Rem-Deform}}$ represent the numbers of weights in Expression Deform-Net, Nose/Eyes/Mouth Deform-Net and Remainder Deform-Net, respectively.

Module	Layer	In-Feature	Out-Feature
Expression Hyper-Net	Linear ReLU	128	128
	Linear ReLU	128	128
	Linear	128	$N_{w_{Exp-Deform}}$
Nose/Eyes/Mouth Hyper-Net	Linear ReLU	48	48
	Linear ReLU	48	48
	Linear	48	$N_{w_{N/E/M-Deform}}$
Remainder Hyper-Net	Linear ReLU	112	112
	Linear ReLU	112	112
	Linear	112	$N_{w_{Rem-Deform}}$

Table B.3: Architecture of the Landmarks-Net (LMs-Net) in the Deformation Networks.

Module	Layer	In-Feature	Out-Feature
Expression LMs-Net	Linear LeakyReLU	128	256
	Linear LeakyReLU	256	256
	Linear	256	15
Nose/Eyes/Mouth LMs-Net	Linear LeakyReLU	48	256
	Linear LeakyReLU	256	256
	Linear	256	15
Remainder LMs-Net	Linear LeakyReLU	112	256
	Linear LeakyReLU	256	256
	Linear	256	15

Table B.4: Architecture of the Blending WeightsNet in the Deformation Networks.

Module	Layer	In-Feature	Out-Feature
Blending PartsNet	Linear LeakyReLU	3	128
	Linear LeakyReLU	128	128
	Linear Softmax	128	5

References

[1] V. F. Abrevaya, A. Boukhayma, S. Wuhrer, and E. Boyer. “A generative 3D facial model by adversarial training”. In: (2019).

[2] M. A. Aliari, A. Beauchamp, T. Popa, and E. Paquette. “Face Editing Using Part-Based Optimization of the Latent Space”. In: *Computer Graphics Forum*. Vol. 42. 2. Wiley Online Library. 2023, pp. 269–279.

[3] T. Bagautdinov, C. Wu, J. Saragih, P. Fua, and Y. Sheikh. “Modeling facial geometry using compositional vaes”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018, pp. 3877–3886.

[4] V. Blanz, C. Basso, T. Poggio, and T. Vetter. “Reanimating faces in images and video”. In: *Computer graphics forum*. Vol. 22. 3. Wiley Online Library. 2003, pp. 641–650.

[5] V. Blanz and T. Vetter. “A morphable model for the synthesis of 3D faces”. In: *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*. 1999, pp. 187–194.

[6] J. Booth, E. Antonakos, S. Ploumpis, G. Trigeorgis, Y. Panagakis, and S. Zafeiriou. “3d face morphable models" in-the-wild"”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 48–57.

[7] J. Booth, A. Roussos, A. Ponniah, D. Dunaway, and S. Zafeiriou. “Large Scale 3D Morphable Models”. In: *International Journal of Computer Vision* 126 (2018), pp. 233–254.

[8] J. Booth, A. Roussos, S. Zafeiriou, A. Ponniah, and D. Dunaway. "A 3d morphable model learnt from 10,000 faces". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2016, pp. 5543–5552.

[9] S. Bouaziz, Y. Wang, and M. Pauly. "Online modeling for realtime facial animation". In: *ACM Transactions on Graphics (ToG)* 32.4 (2013), pp. 1–10.

[10] A. Brunton, T. Bolkart, and S. Wuhrer. "Multilinear wavelets: A statistical shape space for human faces". In: *European Conference on Computer Vision*. Springer. 2014, pp. 297–312.

[11] C. Cao, Q. Hou, and K. Zhou. "Displaced dynamic expression regression for real-time facial tracking and animation". In: *ACM Transactions on graphics (TOG)* 33.4 (2014), pp. 1–10.

[12] C. Cao, Y. Weng, S. Lin, and K. Zhou. "3D shape regression for real-time facial animation". In: *ACM Transactions on Graphics (TOG)* 32.4 (2013), pp. 1–10.

[13] Z. Chai, H. Zhang, J. Ren, D. Kang, Z. Xu, X. Zhe, C. Yuan, and L. Bao. "REALY: Rethinking the Evaluation of 3D Face Reconstruction". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2022.

[14] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, et al. "Shapenet: An information-rich 3d model repository". In: *arXiv preprint arXiv:1512.03012* (2015).

[15] Y. Chang, C. Hu, R. Feris, and M. Turk. "Manifold based analysis of facial expression". In: *Image and Vision Computing* 24.6 (2006), pp. 605–614.

[16] Y. Chang, C. Hu, and M. A. Turk. "Manifold of facial expression." In: *AMFG*. 2003, pp. 28–35.

[17] Z. Chen, V. G. Kim, M. Fisher, N. Aigerman, H. Zhang, and S. Chaudhuri. "Decor-gan: 3d shape detailization by conditional refinement". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2021, pp. 15740–15749.

[18] Z. Chen and H. Zhang. "Learning implicit fields for generative shape modeling". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 5939–5948.

[19] S. Cheng, M. Bronstein, Y. Zhou, I. Kotsia, M. Pantic, and S. Zafeiriou. "Meshgan: Non-linear 3d morphable models of faces". In: *arXiv preprint arXiv:1903.10384* (2019).

[20] J. Chibane, T. Alldieck, and G. Pons-Moll. "Implicit Functions in Feature Space for 3D Shape Reconstruction and Completion". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE. June 2020.

[21] J. Chibane and G. Pons-Moll. "Implicit feature networks for texture completion from partial 3d data". In: *Computer Vision–ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*. Springer. 2020, pp. 717–725.

[22] J. Chibane, G. Pons-Moll, et al. "Neural unsigned distance fields for implicit function learning". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 21638–21652.

[23] G. Chou, I. Chugunov, and F. Heide. "GenSDF: Two-Stage Learning of Generalizable Signed Distance Functions". In: *arXiv preprint arXiv:2206.02780* (2022).

[24] H. Dai, N. Pears, W. Smith, and C. Duncan. "Statistical Modeling of Craniofacial Shape and Texture". In: *International Journal of Computer Vision* 128.2 (2019), pp. 547–571.

[25] H. Dai, N. Pears, W. A. Smith, and C. Duncan. "A 3d morphable model of craniofacial shape and texture variation". In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 3085–3093.

[26] M. Defferrard, X. Bresson, and P. Vandergheynst. "Convolutional neural networks on graphs with fast localized spectral filtering". In: *Advances in neural information processing systems* 29 (2016).

[27] Y. Deng, J. Yang, and X. Tong. "Deformed implicit field: Modeling 3d shapes with learned dense correspondence". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 10286–10296.

[28] H. Drira, B. B. Amor, A. Srivastava, M. Daoudi, and R. Slama. "3D face recognition under expressions, occlusions, and pose variations". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.9 (2013), pp. 2270–2283.

[29] Y. Duan, H. Zhu, H. Wang, L. Yi, R. Nevatia, and L. J. Guibas. "Curriculum deepsdf". In: *European Conference on Computer Vision*. Springer. 2020, pp. 51–67.

[30] B. Egger, W. A. Smith, A. Tewari, S. Wuhrer, M. Zollhoefer, T. Beeler, F. Bernard, T. Bolkart, A. Kortylewski, S. Romdhani, et al. "3d morphable face models—past, present, and future". In: 39.5 (2020), pp. 1–38.

[31] Y. Feng, H. Feng, M. J. Black, and T. Bolkart. "Learning an animatable detailed 3d face model from in-the-wild images". In: 40.4 (2021), pp. 1–13.

[32] C. Ferrari, S. Berretti, P. Pala, and A. Del Bimbo. "A sparse and locally coherent morphable face model for dense semantic correspondence across heterogeneous 3D faces". In: *IEEE transactions on pattern analysis and machine intelligence* 44.10 (2021), pp. 6667–6682.

[33] S. Foti, B. Koo, D. Stoyanov, and M. J. Clarkson. "3D Generative Model Latent Disentanglement via Local Eigenprojection". In: *Computer Graphics Forum*. Wiley Online Library. 2023.

[34] S. Foti, B. Koo, D. Stoyanov, and M. J. Clarkson. "3D Shape Variational Autoencoder Latent Disentanglement via Mini-Batch Feature Swapping for Bodies and Faces". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 18730–18739.

[35] M. Garland and P. S. Heckbert. "Surface simplification using quadric error metrics". In: *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*. 1997, pp. 209–216.

[36] D. Ghafourzadeh, S. Fallahdoust, C. Rahgoshay, A. Beauchamp, A. Aubame, T. Popa, and E. Paquette. "Local control editing paradigms for part-based 3D face morphable models". In: *Computer Animation and Virtual Worlds* 32.6 (), e2028.

[37] D. Ghafourzadeh, C. Rahgoshay, S. Fallahdoust, A. Aubame, A. Beauchamp, T. Popa, and E. Paquette. "Part-based 3D face morphable model with anthropometric local control". In: *Graphics Interface 2020*. 2019.

[38] S. Giebenhain, T. Kirschstein, M. Georgopoulos, M. Rünz, L. Agapito, and M. Nießner. "Learning Neural Parametric Head Models". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 21003–21012.

[39] 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* 27 (2014).

[40] S. Graßhof, H. Ackermann, S. S. Brandt, and J. Ostermann. "Apathy Is the Root of All Expressions". In: *2017 12th IEEE International Conference on Automatic Face Gesture Recognition (FG 2017)*. 2017, pp. 658–665.

[41] S. Grasshof, H. Ackermann, S. S. Brandt, and J. Ostermann. "Multilinear Modelling of Faces and Expressions". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43.10 (2021), pp. 3540–3554.

[42] Y. Gu and N. Pears. "Parts-based Implicit 3D Face Modeling". In: *19th International Conference on Computer Vision Theory and Applications (VISAPP 2024): VISAPP 2024*. SciTePress. 2023.

[43] Y. Gu, N. Pears, and H. Sun. "Adversarial 3D Face Disentanglement of Identity and Expression". In: *2023 IEEE 17th International Conference on Automatic Face and Gesture Recognition (FG)*. IEEE. 2023, pp. 1–7.

[44] M.-H. Guo, J.-X. Cai, Z.-N. Liu, T.-J. Mu, R. R. Martin, and S.-M. Hu. "Pct: Point cloud transformer". In: *Computational Visual Media* 7 (2021), pp. 187–199.

[45] J. Ho, A. Jain, and P. Abbeel. "Denoising diffusion probabilistic models". In: *Advances in neural information processing systems* 33 (2020), pp. 6840–6851.

[46] Z.-H. Jiang, Q. Wu, K. Chen, and J. Zhang. "Disentangled representation learning for 3D face shape". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 11957–11966.

[47] Y. Jung, W. Jang, S. Kim, J. Yang, X. Tong, and S. Lee. "Deep deformable 3d caricatures with learned shape control". In: *ACM SIGGRAPH 2022 Conference Proceedings*. 2022, pp. 1–9.

[48] A. Kacem, K. Cherenkova, and D. Aouada. "Disentangled Face Identity Representations for joint 3D Face Recognition and Expression Neutralisation". In: *arXiv preprint arXiv:2104.10273* (2021).

[49] A. Kanazawa, S. Tulsiani, A. A. Efros, and J. Malik. "Learning category-specific mesh reconstruction from image collections". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018, pp. 371–386.

[50] H. Kim and A. Mnih. "Disentangling by factorising". In: *International Conference on Machine Learning*. PMLR. 2018, pp. 2649–2658.

[51] J. Kim, J. Yoo, J. Lee, and S. Hong. "Setvae: Learning hierarchical composition for generative modeling of set-structured data". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 15059–15068.

[52] D. P. Kingma and J. Ba. "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980* (2014).

[53] D. P. Kingma, M. Welling, et al. "An introduction to variational autoencoders". In: *Foundations and Trends® in Machine Learning* 12.4 (2019), pp. 307–392.

[54] D. P. Kingma and M. Welling. "Auto-encoding variational bayes". In: *arXiv preprint arXiv:1312.6114* (2013).

[55] D.-T. Lee and B. J. Schachter. "Two algorithms for constructing a Delaunay triangulation". In: *International Journal of Computer & Information Sciences* 9.3 (1980), pp. 219–242.

[56] Y. Lei, M. Bennamoun, M. Hayat, and Y. Guo. "An efficient 3D face recognition approach using local geometrical signatures". In: *Pattern Recognition* 47.2 (2014), pp. 509–524.

[57] J. P. Lewis, M. Cordner, and N. Fong. "Pose space deformation: a unified approach to shape interpolation and skeleton-driven deformation". In: *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*. 2000, pp. 165–172.

[58] H. Li, D. Huang, J.-M. Morvan, Y. Wang, and L. Chen. "Towards 3D face recognition in the real: a registration-free approach using fine-grained matching of 3D keypoint descriptors". In: *International Journal of Computer Vision* 113.2 (2015), pp. 128–142.

[59] R. Li, K. Bladin, Y. Zhao, C. Chinara, O. Ingraham, P. Xiang, X. Ren, P. Prasad, B. Kishore, J. Xing, et al. "Learning formation of physically-based face attributes". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 3410–3419.

[60] S. Li, M. Liu, and C. Walder. "EditVAE: Unsupervised parts-aware controllable 3D point cloud shape generation". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 36. 2. 2022, pp. 1386–1394.

[61] T. Li, T. Bolkart, M. J. Black, H. Li, and J. Romero. "Learning a model of facial shape and expression from 4D scans." In: *ACM Trans. Graph.* 36.6 (2017), pp. 194–1.

[62] T. Li, T. Bolkart, M. J. Black, H. Li, and J. Romero. "Learning a model of facial shape and expression from 4D scans". In: *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)* 36.6 (2017), 194:1–194:17.

[63] Y. Lipman. "Phase transitions, distance functions, and implicit neural representations". In: *arXiv preprint arXiv:2106.07689* (2021).

[64] F. Liu, L. Tran, and X. Liu. "3d face modeling from diverse raw scan data". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 9408–9418.

[65] F. Liu, R. Zhu, D. Zeng, Q. Zhao, and X. Liu. "Disentangling features in 3D face shapes for joint face reconstruction and recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 5216–5225.

[66] S. Liu, S. Saito, W. Chen, and H. Li. "Learning to infer implicit surfaces without 3d supervision". In: *Advances in Neural Information Processing Systems* 32 (2019).

[67] S. Liu, L. Giles, and A. Ororbia. "Learning a hierarchical latent-variable model of 3d shapes". In: *2018 international conference on 3D vision (3DV)*. IEEE. 2018, pp. 542–551.

[68] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song. "Sphereface: Deep hypersphere embedding for face recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 212–220.

[69] Z. Liu, Y. Feng, M. J. Black, D. Nowrouzezahrai, L. Paull, and W. Liu. "Meshdiffusion: Score-based generative 3d mesh modeling". In: *arXiv preprint arXiv:2303.08133* (2023).

[70] M. Lüthi, T. Gerig, C. Jud, and T. Vetter. "Gaussian process morphable models". In: *IEEE transactions on pattern analysis and machine intelligence* 40.8 (2017), pp. 1860–1873.

[71] Marian Kleineberg. *mesh-to-sdf*. `https://github.com/marian42/mesh_to_sdf`. 2021.

[72] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, and A. Geiger. "Occupancy networks: Learning 3d reconstruction in function space". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 4460–4470.

[73] A. Mollahosseini, B. Hasani, and M. H. Mahoor. "Affectnet: A database for facial expression, valence, and arousal computing in the wild". In: *IEEE Transactions on Affective Computing* 10.1 (2017), pp. 18–31.

[74] T. Möller and B. Trumbore. "Fast, minimum storage ray/triangle intersection". In: *ACM SIGGRAPH 2005 Courses*. 2005, 7–es.

[75] S. Moschoglou, S. Ploumpis, M. A. Nicolaou, A. Papaioannou, and S. Zafeiriou. "3dfacegan: Adversarial nets for 3d face representation, generation, and translation". In: *International Journal of Computer Vision* 128 (2020), pp. 2534–2551.

[76] A. Muntoni and P. Cignoni. *PyMeshLab*. Jan. 2021.

[77] N. Olivier, K. Baert, F. Danieau, F. Multon, and Q. Avril. "Facetunegan: Face autoencoder for convolutional expression transfer using neural generative adversarial networks". In: *Computers & Graphics* 110 (2023), pp. 69–85.

[78] H. Otroshi Shahreza and S. Marcel. "Face Reconstruction from Facial Templates by Learning Latent Space of a Generator Network". In: *Advances in Neural Information Processing Systems* 36 (2024).

[79] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove. "Deepsdf: Learning continuous signed distance functions for shape representation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 165–174.

[80] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al. "Pytorch: An imperative style, high-performance deep learning library". In: *Advances in neural information processing systems* 32 (2019), pp. 8026–8037.

[81] *A 3D Face Model for Pose and Illumination Invariant Face Recognition*. IEEE. Genova, Italy, 2009.

[82] N. Pears, H. Dai, W. Smith, and H. Sun. "Laplacian ICP for Progressive Registration of 3D Human Head Meshes". In: *2023 IEEE 17th International Conference on Automatic Face and Gesture Recognition (FG)*. IEEE. 2023, pp. 1–7.

[83] S. Peng, J. Dong, Q. Wang, S. Zhang, Q. Shuai, X. Zhou, and H. Bao. "Animatable neural radiance fields for modeling dynamic human bodies". In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 14314–14323.

[84] S. Ploumpis, E. Ververas, E. O'Sullivan, S. Moschoglou, H. Wang, N. Pears, W. Smith, B. Gecer, and S. P. Zafeiriou. "Towards a complete 3D morphable model of the human head". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2020).

[85] B. Poole, A. Jain, J. T. Barron, and B. Mildenhall. "Dreamfusion: Text-to-3d using 2d diffusion". In: *arXiv preprint arXiv:2209.14988* (2022).

[86] C. R. Qi, H. Su, K. Mo, and L. J. Guibas. "Pointnet: Deep learning on point sets for 3d classification and segmentation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 652–660.

[87] C. R. Qi, L. Yi, H. Su, and L. J. Guibas. "Pointnet++: Deep hierarchical feature learning on point sets in a metric space". In: *arXiv preprint arXiv:1706.02413* (2017).

[88] A. Ranjan, T. Bolkart, S. Sanyal, and M. J. Black. "Generating 3D faces using Convolutional Mesh Autoencoders". In: *European Conference on Computer Vision (ECCV)*. 2018, pp. 725–741.

[89] S. Saito, Z. Huang, R. Natsume, S. Morishima, A. Kanazawa, and H. Li. "Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization". In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 2304–2314.

[90] C. Shan, S. Gong, and P. W. McOwan. "Appearance manifold of facial expression". In: *International Workshop on Human-Computer Interaction*. Springer. 2005, pp. 221–230.

[91] V. Sitzmann, J. N. Martel, A. W. Bergman, D. B. Lindell, and G. Wetzstein. "Implicit Neural Representations with Periodic Activation Functions". In: *arXiv*. 2020.

[92] O. Sorkine and M. Alexa. "As-rigid-as-possible surface modeling". In: *Symposium on Geometry processing*. Vol. 4. Citeseer. 2007, pp. 109–116.

[93] O. Sorkine, D. Cohen-Or, Y. Lipman, M. Alexa, C. Rössl, and H.-P. Seidel. "Laplacian surface editing". In: *Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing*. 2004, pp. 175–184.

[94] H. Sun, N. Pears, and Y. Gu. "Information bottlenecked variational autoencoder for disentangled 3d facial expression modelling". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2022, pp. 157–166.

[95] R. Sundararaman, G. Pai, and M. Ovsjanikov. *Implicit field supervision for robust non-rigid shape matching*. 2022. arXiv: 2203.07694 [cs.CV].

[96] O. Taheri, Y. Zhou, D. Tzionas, Y. Zhou, D. Ceylan, S. Pirk, and M. J. Black. "GRIP: Generating interaction poses using spatial cues and latent consistency". In: *International conference on 3D vision (3DV)*. 2024.

[97] F. Taherkhani, A. Rai, Q. Gao, S. Srivastava, X. Chen, F. de la Torre, S. Song, A. Prakash, and D. Kim. "Controllable 3D Generative Adversarial Face Model via Disentangling Shape and Appearance". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2023, pp. 826–836.

[98] T. Takikawa, J. Litalien, K. Yin, K. Kreis, C. Loop, D. Nowrouzezahrai, A. Jacobson, M. McGuire, and S. Fidler. "Neural geometric level of detail: Real-time rendering with implicit 3d shapes". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 11358–11367.

[99] Q. Tan, L. Gao, Y.-K. Lai, and S. Xia. "Variational autoencoders for deforming 3d mesh models". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 5841–5850.

[100] Y. Tang, Y. Qian, Q. Zhang, Y. Zeng, J. Hou, and X. Zhe. "WarpingGAN: Warping multiple uniform priors for adversarial 3D point cloud generation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 6397–6405.

[101] M. Tarasiou, R. A. Potamias, E. O'Sullivan, S. Ploumpis, and S. Zafeiriou. "Locally Adaptive Neural 3D Morphable Models". In: *arXiv preprint arXiv:2401.02937* (2024).

[102] A. Tewari, F. Bernard, P. Garrido, G. Bharaj, M. Elgharib, H.-P. Seidel, P. Pérez, M. Zollhofer, and C. Theobalt. "Fml: Face model learning from videos". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 10812–10822.

[103] A. Tewari, H.-P. Seidel, M. Elgharib, C. Theobalt, et al. "Learning complete 3d morphable face models from images and videos". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 3361–3371.

[104] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Nießner. "Face2face: Real-time face capture and reenactment of rgb videos". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 2387–2395.

[105] J. Thies, M. Zollhöfer, M. Nießner, L. Valgaerts, M. Stamminger, and C. Theobalt. "Real-time expression transfer for facial reenactment." In: *ACM Trans. Graph.* 34.6 (2015), pp. 183–1.

[106] L. Tran and X. Liu. "Nonlinear 3d face morphable model". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 7346–7355.

[107] J. Wu, C. Zhang, T. Xue, B. Freeman, and J. Tenenbaum. "Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling". In: *Advances in neural information processing systems* 29 (2016).

[108] Y.-P. Xiao, Y.-K. Lai, F.-L. Zhang, C. Li, and L. Gao. "A survey on deep geometry learning: From a representation perspective". In: *Computational Visual Media* 6 (2020), pp. 113–133.

[109] Q.-C. Xu, T.-J. Mu, and Y.-L. Yang. "A survey of deep learning-based 3D shape generation". In: *Computational Visual Media* 9.3 (2023), pp. 407–442.

[110] P. Yan, J. Gregson, Q. Tang, R. Ward, Z. Xu, and S. Du. "NEO-3DF: Novel Editing-Oriented 3D Face Creation and Reconstruction". In: *Proceedings of the Asian Conference on Computer Vision*. 2022, pp. 486–502.

[111] X. Yan, J. Yang, E. Yumer, Y. Guo, and H. Lee. "Perspective transformer nets: Learning single-view 3d object reconstruction without 3d supervision". In: *Advances in neural information processing systems* 29 (2016).

[112] H. Yang, H. Zhu, Y. Wang, M. Huang, Q. Shen, R. Yang, and X. Cao. "FaceScape: a Large-scale High Quality 3D Face Dataset and Detailed Riggable 3D Face Prediction". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020.

[113] T. Yenamandra, A. Tewari, F. Bernard, H.-P. Seidel, M. Elgharib, D. Cremers, and C. Theobalt. "i3dmm: Deep implicit 3d morphable model of human heads". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 12803–12813.

[114] L. Yin, X. Wei, Y. Sun, J. Wang, and M. J. Rosato. "A 3D facial expression database for facial behavior research". In: *7th international conference on automatic face and gesture recognition (FGR06)*. IEEE. 2006, pp. 211–216.

[115] W. Zhang, X. Ji, K. Chen, Y. Ding, and C. Fan. "Learning a Facial Expression Embedding Disentangled From Identity". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 6759–6768.

[116] Z. Zhang, C. Yu, H. Li, J. Sun, and F. Liu. "Learning Distribution Independent Latent Representation for 3D Face Disentanglement". In: *2020 International Conference on 3D Vision (3DV)*. IEEE. 2020, pp. 848–857.

[117] M. Zheng, H. Yang, D. Huang, and L. Chen. "Imface: A nonlinear 3d morphable face model with implicit neural representations". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 20343–20352.

[118] Z. Zheng, T. Yu, Q. Dai, and Y. Liu. "Deep implicit templates for 3d shape representation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 1429–1439.

[119] L. Zhou, Y. Du, and J. Wu. "3d shape generation and completion through point-voxel diffusion". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 5826–5835.

[120] H. Zhu, H. Yang, L. Guo, Y. Zhang, Y. Wang, M. Huang, M. Wu, Q. Shen, R. Yang, and X. Cao. "FaceScape: 3D Facial Dataset and Benchmark for Single-View 3D Face Reconstruction". In: *IEEE transactions on pattern analysis and machine intelligence* (2023).

[121] J.-Y. Zhu, Z. Zhang, C. Zhang, J. Wu, A. Torralba, J. Tenenbaum, and B. Freeman. "Visual object networks: Image generation with disentangled 3D representations". In: *Advances in neural information processing systems* 31 (2018).

[122] W. Zielonka, T. Bolkart, and J. Thies. "Towards Metrical Reconstruction of Human Faces". In: *European Conference on Computer Vision* (2022).