Imperial College London

Department of Computing

Machine Learning Methods for Facial Attribute Editing

Evangelos Ververas

April 2022

Supervised by Prof. Stefanos Zafeiriou

Submitted in part fulfilment of the requirements for the degree of PhD in Computing and the Diploma of Imperial College London. This thesis is entirely my own work, and, except where otherwise indicated, describes my own research.

Abstract

In this thesis, we explore the problem of facial attribute analysis and editing in images, from the scope of two major fields of Machine Learning, those of Components Analysis (CA) and Deep Learning (DL). First, we present a CA method for analysing and editing facial data. Then, we present a DL algorithm for animating facial images according to expressions and speech. Finally, we present a method for improving gaze estimation generalisation to unseen image domains and showcase applications to eye gaze editing.

Although CA methods are able to capture only linear relationships in data, they can still be useful with well-aligned data, such as UV maps of facial texture. In this Thesis, we propose robust extensions of the Joint and Individual Variance Explained (JIVE) method, for the recovery of joint and individual components in visual facial data, captured in unconstrained conditions and possibly containing sparse non-Gaussian errors and missing data. We demonstrate the effectiveness of the proposed methods to several computer vision applications, namely facial expression synthesis and 2D and 3D face age progression in-the-wild.

CA methods usually fall short in image generation, as they fail to generate details. On the contrary, Image-to-image (i2i) translation, which is the problem of translating images between image domains, has recently seen remarkable progress since the advent of DL and Generative Adversarial Networks (GANs). In this Thesis, we study the problem of i2i translation, under a set of continuous parameters that correspond to statistical blendshape models of facial motion. We show that it is possible to edit facial images according to expression and speech blendshapes using "sliders", which are more flexible than discrete expressions or action units.

Lastly, realistically animating gaze is crucial for achieving high quality facial animations. To this end, large datasets of faces with gaze annotations are required for training. In this Thesis, we present a weakly-supervised method for improving gaze estimation generalization to unseen domains, by harnessing arbitrary unlabelled "in-the-wild" face images. Unlike previous methods, we tackle gaze estimation as end-to-end, dense 3D reconstruction of eyes and experimentally validate the benefits of this choice. Particularly, we show improvements in semi-supervised and cross-dataset gaze estimation. Finally, we showcase how our methods can be employed for training efficient models for gaze editing.

Acknowledgements

I feel very grateful to have had the opportunity to undertake the research degree of Doctor of Philosophy in the Department of Computing at Imperial College London. It is really challenging to put into words how exciting these five years have been. For that, I would like to thank the department of Computing at Imperial College London for the financial support of the project through my Teaching Scholarship. Without this, none of this work would have been possible.

It goes without saying that my deepest thanks goes to my supervisor, Professor Stefanos Zafeiriou. I could really not thank him enough for his vital role in this project. It is truly outstanding how supportive he has been, from the very beginning. His generosity with his time, even during the busiest periods, as well as his willingness to transfer his knowledge have really been impressive. What strikes me most is how his own passion for this field can be transferred to his students and therefore be reflected at the quality of the work. Of course, I could never forget those times when his role as a supervisor changed to that of a good friend. He truly is an inspiration and I feel extremely grateful to have met him.

A special thanks goes to my colleagues, Stelios M., Stelios P., Thanos, Jiankang, Grigoris, Markos, George, Christos, Alex, Rolandos, Dimitris and Panagiotis for their support and for the numerous fruitful discussions we had, which really took the project a step further. I feel very grateful for the excellent collaboration we had and for all the knowledge I gained from them.

I would also like to express my gratitude to Professor Ioannis Theoharis for supporting my decision to pursue a PhD in the field during the last year of my undergraduate studies. He had been very encouraging and always willing to provide his support. Similarly, I would like to thank Dr Amani El-Kholy, from the department of Computing at Imperial College London, for her continuous and immediate support for any requests related to my PhD studies. I also feel very grateful for my tutors in the Academic Teaching and Learning Programme related to my scholarship, for the knowledge and skills provided regarding teaching in Higher Education.

Lastly, it is difficult to express in words how thankful I am for my family and friends and the support I have received by them from the beginning of my studies. Mum and Dad and my beloved sister, Katerina, you have never doubted my professional endeavours. My dear partner, Katerina, I couldn’t thank you enough for the emotional support you kept offering all these years, when facing various PhD challenges. Thank you all so much for always being there for me. My deepest love to you all.

Overall, I wouldn’t change anything about this adventure. It has filled me with knowledge and great memories. Thinking of these times, I now feel overwhelmed with gratitude and blessed to have been able to undertake this degree at Imperial College London.

Contents

List of Tables

List of Figures

List of Algorithms

List of Abbreviations

3DMM	3D Morphable Model
4DFAB	4D Face Database for Expression Analysis and Biometric Applications
ADMM	Alternating-Directions Method of Multipliers
AFW	Annotated Faces In-The-Wild (Dataset)
AP	Average Precision
ATSS	Adaptive Training Sample Selection
ATW	Age In-The-Wild (Dataset)
AU	Action Unit
AUC	Area Under Curve
AVA	Atomic Visual Actions
BKRRR	Bilinear Kernel Reduced Rank Regression
BN	Batch Normalisation
CA	Component Analysis
CCA	Canonical Correlation Analysis
CDL	Coupled Dictionary Learning
CG	Craniofacial Growth
CK+	Extended Cohn-Kanade (Dataset)
CNN	Convolutional Neural Network
COBE	Common Orthogonal Basis Extraction
CoMA	Convolutional Mesh Autoencoder
DARB	Deep Ageing with Restricted Boltzmann Machines
DCNN	Deep Convolutional Neural Network
DIAT	Deep Identity-Aware Transfer of Facial Attributes
DL	Deep Learning
DNR	Deferred Neural Rendering

EAP	Exemplar-based Age Progression
FACS	Facial Action Coding System
FES	Facial Expression Synthesis
FLAME	Faces Learned with an Articulated Model and Expression
FT	Face Transformer
GAN	Generative Adversarial Network
GFL	Generalised Focal Loss
i2i	Image-to-image
IAAP	Illumination Aware Age Progression
IcGAN	Invertible Conditional Generative Adversarial Network
IED	Image Euclidean Distance
ITW	In-The-Wild
ITWG	In-The-Wild Gaze (Dataset)
JIVE	Joint and Individual Variation Explained
LAEO	Looking at Each Other
LFW	Labelled Faces in the wild (Dataset)
LRW	Lip Reading In-The-Wild (Dataset)
LRW-3D	Lip Reading Words in 3D (Dataset)
LSFM	Large Scale Facial Model
LSGAN	Least Squares Generative Adversarial Network
LYHM	Liverpool-York Head Model
MLP	Multi-Layer Perceptron
MPIE	Multi- Pose, Illumination, Expressions (Dataset)
PAFPN	Path-Aggregation Feature Pyramid Network
PCA	Principal Component Analysis
RaGAN	Relativistic Average Generative Adversarial Network
RCICA	Robust Correlated and Individual Component Analysis
RFA	Recurrent Face Aging
RGAN	Relativistic Generative Adversarial Network
RJIVE	Robust Joint and Individual Variation Explained
RJIVE-M	Robust Joint and Individual Variation Explained with Missing Values
ROC	Receiver Operating Characteristics
RPCA	Robust Principal Component Analysis
RRE	Relative Reconstruction Error

SCRFD	Sample and Computation Redistribution for Efficient Face Detection
SGD	Stochastic Gradient Descent
SliderGAN-RaD	SliderGAN with Relativistic Average Discriminator
SliderGAN-WGP	SliderGAN with Wasserstein GAN with Gradient Penalty
SRJIVE	Scalable Robust Joint and Individual Variation Explained
SVD	Singular Value Decomposition
SVM	Support Vector Machines
SVT	Singular Value Thresholding
UHM	Unified Head Model
WGAN	Wasserstein Generative Adversarial Network
WGAN-GP	Wasserstein Generative Adversarial Network with Gradient Penalty

CHAPTER 1

Introduction

Contents

1.1 Problem Scope

In recent years, it has been more popular than ever to use digital media in various aspects of human life such as entertainment, communication and education. This is mainly due to the interactivity they offer in comparison to more traditional means, as well as the possibilities they offer for new and exciting applications or products. For example, video conferences have eliminated the consideration of distance from effective human communication and collaboration, while modern digital animation tools have long enabled the production of high quality films and video games which would otherwise be impossible.

The human face is one of the most expressive parts of the human body, while at the same time it is one of the most challenging to parameterise and reproduce by digital means. Undoubtedly, facial representation and animation of very high quality have already been achieved and showcased by modern movies and video games. However, these results are mostly the outcomes of the significant manual effort of tens or hundreds of 3D character creators and animators working for their production. Therefore, to decrease the cost and time required to produce such works, as well as to be able to incorporate the human face in more common interactive digital applications, automatic methods to support face analysis, editing and animation have to be developed.

Face attribute analysis and editing is a prominent problem in the field of Computer Vision, which refers to the process of automatically identifying and manipulating characteristics of the human face such as age, expression, gender, identity, gaze direction, hair style, hair colour, etc. Significant advancements have been recently made in this field, mainly due to the advent of Deep Learning (DL) and the large amounts of available data. In particular, generative modelling has seen substantial improvements following the developments of Generative Adversarial Networks (GANs) [1], leading to unprecedented levels of detail achieved in 2D face generation and animation [2, 3].

Through the years, multiple approaches have been adopted for analysing and editing facial features in images and 3D data, including Component Analysis (CA) methods and Deep Learning (DL) ones. In more detail, Component Analysis (CA) methods have been employed to discover underlying structures in facial data, which have been useful for performing both discriminative and generative tasks, such as face identification [4, 5, 6], facial expression recognition [7, 8, 9] and image imputation [10, 11]. Especially, methods based on CA can still achieve high performance in 3D shape modelling of the human face and head [12, 13].

Joint and individual variation among different views of facial data is a type of underlying structure with particular importance for identifying and manipulating facial attributes. Several CA methods that attempt to address this problem have been proposed in literature [14, 15, 16]. However, gross, sparse, non-Gaussian errors in facial data, such as the salt-and-pepper noise in imaging devices, occlusions in facial images, etc., can cause the above methods to be unstable. In the first part of the Thesis we present a CA algorithm for discovering joint and individual variation in facial data, while also handling data contaminated by errors. We demonstrate the merits of our technique in 2D and 3D face attribute manipulation and identity verification.

Even though CA methods have been successfully applied to various face analysis and editing tasks as mentioned above, they are still restricted by their ability to recover only linear relationships between data. In contrast, DL-based approaches can deal with more complex and demanding tasks, such as high-quality image generation where GANs have exhibited significant improvements.

Facial expression synthesis in images is an application which has greatly benefited by the development of GANs and DL. Recent methods, have been able to learn translations between two or more distinct image domains (e.g. collections of images of people in two or more different facial expressions), capturing high frequency details, such as wrinkles and details of eyes and teeth [17, 18, 19, 20]. However, none of the above techniques allow continuous editing of faces regarding arbitrary facial motion. In this Thesis, we present an approach to address this problem by utilising 3D statistical blendshape models of expression as a more intuitive means to control facial motion and demonstrate its benefits for face editing in images regarding both expression and speech.

Among facial attributes discussed above, such as facial expressions, age, identity, etc., gaze is one of the most important characteristics in achieving realistic facial animations. Gaze constitutes an especially informative visual cue for understanding someone's emotional state or level of attention, thus, even slight mistakes can cause animated faces to seem unrealistic or robotic. Additionally, manually animating gaze direction can be particularly demanding due to saccadic movements of eyes, i.e fast movements that abruptly change the point of someone's attention. The above reasons pose the requirement for developing automatic techniques for accurate gaze following and editing.

In order to train effective methods for gaze editing, large datasets of faces with gaze annotations are required. Employing pseudo-annotations using off-the-shelf gaze estimation networks, is an attractive but sub-optimal option, especially due to the cross-domain generalisation problems of most gaze retrieval methods [21, 22, 23, 24, 25]. In this Thesis, we present a method for improving generalisation of automatic monocular gaze estimation models that operate on RGB images, harnessing arbitrary unlabelled face images which are abundantly available in the internet, through our weak supervision training framework. Moreover, we employ our models to extract robust gaze annotation from "in-the-wild" data, which we employ to train efficient gaze editing systems.

1.2 Motivation and Objectives

In this Thesis, we explore the problem of face attribute analysis and editing, proposing and experimenting with novel algorithms from two different areas of Machine Learning (ML), namely Component Analysis (CA) and Deep Learning (DL). First, we present linear CA methods for analysing and editing face data according to specific attributes. Then, we report an algorithm for editing face expressions in images, based on DL and Convolutional Neural Networks (CNNs). Finally, by specifically modeling the human eyes in 3D, we propose to recover gaze in a weakly-supervised fashion and develop an algorithm for gaze manipulation on facial images. A common aspect of all methods presented here is that 3D representations and codes of the face are imposed in the structure of the algorithms or data. Here, we present the motivation behind each of the three main research questions we attempt to answer in this Thesis, before progressing into defining the objectives of this work. In particular, the problems we try to solve are the following:

- **Problem 1** Face images consist of hidden components largely connected to the inherent structure and properties of human faces, as well as components related to explicit attributes such as age, gender and facial expression. Discovering those over collections of face images annotated with regards to single or multiple attributes, could be crucial for the tasks of face image synthesis and editing. However, images captured in unconstrained conditions pose significant challenges due to occlusions and missing values, rendering common CA techniques unsuitable for the task. Thus, we pose the following question: Can we develop robust alternatives to known CA algorithms in our attempt to discover useful components from "in-the-wild" face data and employ them to improve performance in attribute transfer and face editing tasks?
- **Problem 2** DL techniques and particularly Generative Adversarial Networks (GANs) have revolutionised image-to-image (i2i) translation tasks, outperforming previous solutions by large margins. Particularly, face expression synthesis has been greatly benefited by recent i2i translation methods. However, early techniques were limited to handling single basic expressions and specific expression intensities. Since expressions cause continuous deformations to the human face and are not limited to a particular set of labeled expressions, is it possible to develop algorithms that control expressions in images based on continuous, generalised and easily perceivable codes, such as the weights of statistical 3D expression blendshapes? Therefore,

is it possible to develop i2i translation techniques, based on Generative Adversarial Networks (GANs), that can be used to produce continuous and smooth animations from single input faces?

- **Problem 3** Gaze estimation has been one of the tasks that have been benefited from the advent of DL algorithms and novel related datasets. Nevertheless, difficulties in collecting diverse data both in terms of the number of different subjects and capturing environments, have led to gaze estimation techniques that do not generalise well to unseen conditions. Recent attempts to improve generalisation are still limited by their approach to inferring gaze as angle or vector regression, ignoring the fact that gaze is inherently a 3D space problem. Based on both previous observations, would it be possible to build gaze estimation methods that consider the 3D structure of eyes and at the same time employ "in-the-wild" face data to improve cross-domain gaze estimation? Moreover, being able to extract robust gaze labels from "in-the-wild" images, could we propose an i2i translation method for accurate gaze manipulation in facial images, further demonstrating the possible applications of facial attribute editing?

Having presented the main challenges we have identified and considered in this work regarding facial attribute editing using Machine Learning methods, we summarise the objectives of this Thesis as follows:

- **Objective 1** Having a collection of "in-the-wild" face images annotated for a single attribute, such as expression or age, we aim to develop robust algorithms to discover joint and individual components, referring to face and attribute-specific components respectively. Particularly, we are able to extend the so-called Joint and Individual Variance Explained (JIVE) method, and propose a robust alternative that can handle occlusions and missing data. We demonstrate the effectiveness of the proposed method on several computer vision applications, namely facial expression synthesis and 2D and 3D face age progression "in-the-wild".

- **Objective 2** Most face expression synthesis methods operate under a limited number of condition labels corresponding to distinct expressions. Other methods, rely on AUs to model expression with continuous codes, however, AUs cannot be easily employed for annotating images or generating expressions, unless these tasks are done by experts. Therefore, we aim to develop i2i translation algorithms to overcome both previous limitations, by relying on continuous and easily perceivable 3D expression blendshapes and expression pseudo-labels extrcted by 3D Morphable Model (3DMM) fitting. In that way, we demonstrate continuous control of face deformation in images not only regarding facial expressions but also speech, indicating that blendshape coding can be an efficient generalised approach to i2i translation-based face image editing.

- **Objective 3** Most gaze estimation algorithms rely on existing annotated datasets and infer gaze directly as angles, vectors or points on screen. However, predicting dense geometry instead of few parameters has been beneficial for tasks such as body and face pose estimation. Moreover, weak-supervision has been already successfully employed for training body pose estimation algorithms without any annotated data. In this work, we aim to combine the previous observations and develop a gaze estimation system which is based on 3D eye reconstruction and is not limited by the available gaze datasets. Instead, it can learn gaze from any "in-the-wild" face dataset through a weakly-supervised training framework. As we demonstrate, incorporating "in-the-wild" face images in training, can significantly improve gaze estimation generalisation to unseen domains. Especially, robust models for gaze estimation can be trained without any gaze supervision. Lastly, we demonstrate the quality of robust gaze pseudo-labels extracted by the above algorithms, by proposing a single-shot, multi-face gaze estimation method and an i2i translation application for gaze editing on faces "in-the-wild".

1.3 Contributions and Thesis Overview

In this Section, we provide a concise overview of the structure of this Thesis, delving into more details about the exact contributions achieved by completing the objectives described in Section 1.2. In particular:

- In Chapter 2, we provide technical background knowledge which is necessary to comprehend the content of the following chapters, as well as we present methods from literature that inspired and challenged our developed methodologies. Particularly, we first provide a short review, as well as details on methods for 3D reconstruction of faces "in-the-wild". Especially, We focus on 3D Morphable Models (3DMMs) of the face, as they are employed by the work in Chapter 3 for obtaining aligned unwrapped textures (UV maps) from images and the work in Chapter 4 as a method to encode expression deformations in face images. Then, we detail the CA methods JIVE [14], COBE [15] and RCICA [16], which are all prior work to our RJIVE method presented in Chapter 3. Lastly, we provide background knowledge on GANs [1, 26] and present the fundamental image-to-image (i2i) translation methods pix2pix [17], CycleGAN [18] and StarGAN [19] which have constituted the basis for developing SliderGAN in Chapter 4.

- In Chapter 3, we present our work "Recovering Joint and Individual Components in Facial Dat" [27], in which we introduce a robust extension to the JIVE algorithm [14], able to handle sparse, gross errors and missing information in facial data. The proposed RJIVE decomposes the data into three terms: a low-rank matrix that captures the joint variation across views, low-rank matrices accounting for structured variation individual to each view, and a sparse matrix collecting the sparse gross errors. Different alternatives of RJIVE are introduced regarding the automatic estimation of the rank of the recovered components, as well as their orthogonality. Additionally, two optimisation problems to reconstruct test samples based on the extracted components are proposed along with algorithms based on the Alternating-Directions Method of Multipliers (ADMM) [28] to tackle them. The proposed methods are applied in three challenging computer vision tasks, namely facial expression synthesis and face age progression in 2D images and 3D data captured "in-the-wild".

- In Chapter 4, we present our work "SliderGAN: Synthesising Expressive Face Images by Sliding 3D Blendshape Parameter" [29], in which we introduce SliderGAN, an algorithm to create synthetic expressions from face images by controlling the weights of a statistical model. Particularly, we are motivated by the successes in 3D face reconstruction methodologies from "in-the-wild" images [30, 31, 32, 13, 33], which make use of a statistical models of 3D facial motion, and propose a methodology for facial image translation using GANs driven by the continuous parameters of the linear blendshapes. Contrary to StarGAN [19] which requires discrete annotations and GANimation [20] which requires annotations of AUs, SliderGAN's training is solely based on pseudo-annotations provided by fitting a 3D Morphable Model (3DMM) to images [13] (for expression deformations) or by aligning audio signals [34] (for speech deformations). We support training by the use of synthetic data leveraging the reconstruction capabilities of statistical shape models and demonstrate that SliderGAN-RaD, a variation of SliderGAN trained within a Relativistic GAN framework [35], is able to produce textures of higher quality than when trained with the standard Wasserstein GAN with gradient penalty [36]. Finally, we showcase that SliderGAN is able to synthesise smooth deformations of expression and speech for on-demand animation, as well as for expression and speech transfer from source images and videos.

- In Chapter 5, we provide details on our work "Weakly-Supervised Gaze Estimation from Synthetic Views" [37], in which we propose a method to tackle gaze estimation as 3D reconstruction of eyes and exploit the abundant "in-the-wild" face images to improve gaze estimation generalisation. Based on a unified 3D representation of eyes, i.e. a 3D eyeball template, we obtain 3D pseudo-ground truth from existing gaze datasets and enforce multiple geometric constraints during training to learn gaze. Inspired by work on self-supervised 3D body pose estimation [38, 39, 40], we propose to train robust cross-domain gaze estimators from unlabeled images of faces "in-the-wild", by designing a weakly-supervised framework in which we enforce multi-view geometric constraints that encourage consistent eye geometry across synthetic views of the same subject. We demonstrate the benefits of our methodology in cross-domain gaze estimation and social activity detection scenarios. Lastly, we demonstrate the validity of our method's results in two tasks a) single-shot, multi-face gaze estimation, which performs gaze estimation in O(1) with regards to the number of faces in the scene and b) gaze re-targeting in images, based on robust gaze pseudo-labels extracted without any gaze supervision.

- In Chapter 6, we provide a summary of the contents of this Thesis, draw conclusions and discuss about possible extensions and future works that could follow the research presented in the previous Chapters.

1.4 Impact and Applications

The Chapters of this Thesis cover CA and DL methodologies developed to tackle specific tasks regarding face attribute editing. Even thought the main target of these chapters is to present effective generative models, discriminative aspects exists in all our approaches. Combining the advantages of both, our methods can be used in applications of various fields, such as:

Biometrics All methods proposed in this Thesis can be employed for improving the accuracy of biometric applications. For example, face identification systems can benefit from low-rank representations extracted by our CA method, from expression neutralisation performed by our DL expression editing algorithm and from gaze frontalisation performed by our gaze editing system. Besides, our "in-the-wild" gaze estimation can be employed to ensure liveliness and attention of the user in user authentication scenarios.

Health and safety The human face, as well as the eye gaze constitute visual cues for the overall health, attention, mental state and tiredness of individuals. Thus, our face analysis and editing and gaze tracking methods could be employed to improve applications in which knowledge of the above information is crucial, such as driver state monitoring systems in vehicles or remote health monitoring systems.

Entertainment Automatic editing and animation methods have been extremely popular recently as they have been deployed in mobile phones, video and photo editing software. All face editing techniques presented in this thesis can be utilised for editing photographs of faces regarding various attributes or creating facial animations. Moreover, our gaze tracking could be utilised for foveated rendering in VR/AR games and applications.

Human-Computer Interaction Understanding human emotions and intentions is a crucial aspect of making human-computer interaction interfaces feel organic. Virtual assistants and automated customer service systems, are some examples of applications which could benefit from the face attribute analysis and editing methodologies of this Thesis. Moreover, accessibility features could be enhanced in interactive applications, by offering control via gaze.

Dataset Augmentation Training robust DL systems extensively relies on acquiring large amounts of training data, covering as much as possible of the variation of the test sets. Our face attribute editing methodologies can be directly employed for dataset augmentation supporting training of human perception tasks such as emotion recognition, face landmarks localisation and gaze estimation.

1.5 Publications

In this Section, I provide the list of publications resulted from work I either lead or contributed in, during the years I have been working as a PhD student at Imperial College London. In particular, I first list the publications which are directly relevant to the work presented in this Thesis and then, the publications which I have co-authored but are not covered in detail in the following Chapters.

1.5.1 Relevant Publications

- C. Sagonas, **E. Ververas**, Y. Panagakis, and S. Zafeiriou, "Recovering joint and individual components in facial data," IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), vol. 40, no. 11, pp. 2668–2681, 2018.

  (Sections 3.2.1, 3.2.2 and 3.2.3 do not constitute personal contributions.)

- **E. Ververas** and S. Zafeiriou, "Slidergan: Synthesizing expressive face images by sliding 3d blendshape parameters," International Journal of Computer Vision (IJCV), vol. 128, no. 10, pp. 2629–2650, 2020.

- **E. Ververas**, P. Gkagkos, J. Deng, J. Guo, M. Doukas, and S. Zafeiriou. "Generalizing Gaze Estimation with Weak-Supervision from Synthetic Views," arXiv preprint arXiv:2212.02997, 2022.

  (Section 5.2.4 does not constitute personal contribution.)

1.5.2 Other Publications

- S. Moschoglou, **E. Ververas**, Y. Panagakis, M. A. Nicolaou, and S. Zafeiriou, "Multi-attribute robust component analysis for facial uv maps," IEEE Journal of Selected Topics in Signal Processing, vol. 12, no. 6, pp. 1324–1337, 2018.

- S. Ploumpis, **E. Ververas**, E. O'Sullivan, S. Moschoglou, H. Wang, N. Pears, W. A. Smith, B. Gecer, and S. Zafeiriou, "Towards a complete 3d morphable model of the human head," IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), vol. 43, no. 11, pp. 4142–4160, 2020.

- J. Booth, A. Roussos, **E. Ververas**, E. Antonakos, S. Ploumpis, Y. Panagakis, and S. Zafeiriou, "3d reconstruction of "in-the-wild" faces in images and videos," IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), vol. 40, no. 11, pp. 2638–2652, 2018.

- J. Deng, J. Guo, **E. Ververas**, I. Kotsia, and S. Zafeiriou, “Retinaface: Single-shot multi-level face localisation in the wild,” in Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR), pp. 5203–5212, 2020.

- D. Kollias, S. Cheng, **E. Ververas**, I. Kotsia, and S. Zafeiriou, “Deep neural network augmentation: Generating faces for affect analysis,” International Journal of Computer Vision (IJCV), vol. 128, pp. 1455–1484, 2020.

- R. A. Potamias, J. Zheng, S. Ploumpis, G. Bouritsas, **E. Ververas**, and S. Zafeiriou, “Learning to generate customized dynamic 3d facial expressions,” in Proceedings of European Conference on Computer Vision (ECCV), pp. 278–294, Springer, 2020.

- J. Deng, A. Roussos, G. G. Chrysos, **E. Ververas**, I. Kotsia, J. Shen, and S. Zafeiriou, “The menpo benchmark for multi-pose 2d and 3d facial landmark localisation and tracking,” International Journal of Computer Vision (IJCV), vol. 127, pp. 599–624, 2018.

CHAPTER 2

Related Work

Contents

In this Chapter, we provide technical information, crucial for understanding the content of the following Chapters. In particular, we provide elaborate details on methods and subjects employed by the works of this Thesis, as well as fundamental and related methodologies which have motivated and challenged the development of our works. First, we discuss the subject of acquiring 3D representations of faces from images and how these can be utilised as the main data representation by 2D face attribute editing algorithms. Then, we present three CA methods, whose limitations we attempt to overcome by the RJIVE algorithms of Chapter 3. The chapter continues with a discussion on some fundamental works on generative modelling and i2i translation, which, since their inception, have inspired numerous face attribute editing and animation techniques including ours, SliderGAN of Chapter 4 and our gaze editing application of Chapter 5. We close the chapter, with a review on the more recent methods of generative face image generation and editing which have recently revolutionize the field with the image fidelity they provide, including 2D and 3D-aware GAN methods, volumetric avatar models and diffusion-based approaches.

2.1 3D Reconstruction of Faces In-The-Wild

3D reconstructions of faces from natural 2D images refers to the process of recovering 3D models of the shape and texture of the face, as well as models of the camera projection and illumination conditions of the scene. Blanz and Vetter [41, 42] where the first to show that it is possible to reconstruct the facial shape and texture from images by fitting on them statistical models of the facial shape and texture, called 3D Morphable Models (3DMMs). Fitting 3DMMs on images involved solving an analysis-by-synthesis problem, expressed as a nonlinear optimisation problem, constrained by the statistical models of shape and texture as well as projection and illumination models reconstructing the scene.

Among face 3DMMs, the Basel model [43], build from scans of 200 people, has been the most popular, while more models of larger scale have been made available through the years. To name a few, the Large-Scale-Facial-Model (LSFM) [44] is a model of facial shape and texture built from 10K face scans, the Liverpool-York-head-Model [45] has been built from 1.2K head scans, modelling the shape and texture of the whole head, while the Unified-Head-Model (UHM) [46] combines the LSFM and LYHM and explicitly models the ears and eyes. Moreover, the FLAME (Faces Learned with an Articulated Model and Expressions) model [47] is built from 3.3K head scans and additional articulation of pose and expression and the CoMA (Convolutional Mesh Autoencoder) model [48] is built from 12.5K head scans from 12 subjects and models shape using mesh convolutions.

Since the work of Balnz and Vetter [41, 42], numerous methods for 3D face reconstruction based on 3DMMs have been proposed [49, 50, 51]. In particular, in [13] the authors proposed to employ statistical texture models of the face, based on image features such as Scale Invariant Feature Transform (SIFT) and Histogram Of Gradients (HOG), avoiding in this way to explicitly model scene illumination. Even thought [13] does not reconstruct facial texture, it can still achieve state-of-the-art results in 3D facial shape reconstruction from “in-the-wild” images. Recently, the focus has been shifted from traditional 3DMM fitting towards approaches that leverage Deep Convolutional Neural Networks (DCNNs). According to these, DCNNs can be employed to learn a regression from images to the parameters of a 3DMM [52] in a supervised fashion, or unsupervised by harnessing differentiable renderers or multiple images of the same person [53, 54, 30, 55, 56]. To obtain higher quality textures and facial shape, some methods such as [31, 32] employ additional networks, called correctives, that operate on top of those regressed by a 3DMM, while others utilise Generative Adversarial Models (GANs) to model texture and fit it on images based on robust identity features [12].

Our aim is to employ 3D reconstruction strategies to recover 3D representations of faces from images, focusing on reconstruction of the facial shape rather than texture. In particular, we aim to employ 3D reconstructed faces from images to create data representations and codes useful for developing 2D facial attribute editing algorithms. That is, in Chapter 3 we develop CA methods based on unwrapped 3D textures (UV maps) which are extracted by 3DMM fitting on images and provide a more elaborate alignment space than 2D alignment. Moreover, in Chapter 4 we utilise 3D expression blendshapes as a method to encode facial expressions in images and train DCNNs to edit expressions based on that encoding. For both our goals described above, we have employed the 3DMM fitting strategy of [13], as its feature-based texture model significantly helps recovering robust reconstructions from faces in arbitrary conditions. To model identity and expression variation we utilise the LSFM and 4DFAB models respectively. In the rest of Section 2.1, we provide details on expression blendshapes, the 3DMM fitting we use and how UV texture maps are extracted from 2D images.

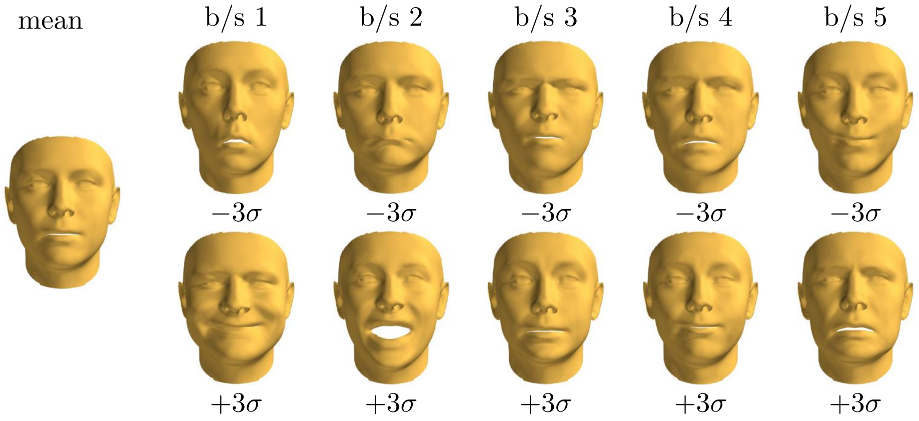


Figure 2.1: Visualisation of the 5 most significant components of the blendshape model of 4DFAB [59] according to their eigenvalue magnitude. The 3D faces of this figure have been generated by applying the components multiplied by weights equal to three times the standard deviation (3σ) to a mean face. The image has been taken from [29]. The figure shows how each individual expression component controls specific parts of the 3D human face.

2.1.1 Expression Blendshape Models

Traditionally 3DMMs of the face or head are built from 3D scans of subjects in neutral poses, allowing them to model shape variation with regards to identity, ethnicity, gender and age. To jointly model expression, those shape models are usually fused with datasets focusing on facial deformation due to expression. Such datasets are the Dynamic 3D FACS Dataset (D3DFACS) [57] which includes 10 subjects performing between 19 and 97 different motions specifically mapped to the Facial Action Coding System (FACS) of Action Units, the FaceWarehouse [58] dataset which includes 3D scans of 20 expressions from each one of 150 participants and the 4DFAB [59] dataset which consists of 4D videos of 180 subjects resulting in almost 2 million 3D face scans. Among those datsets, 4DFAB includes the largest expression variation resulting in more powerful expression models [59], known also as blendshape models.

Blendshape models are frequently used in computer vision tasks as they constitute an effective parametric approach for modelling facial motion. The localised blendshape model [60] proposed a method to localise sparse deformation modes with intuitive visual interpretation. The model was built by sequences of manually collected expressive 3D face meshes. In more detail, a variant of sparse Principal Component Analysis (PCA) was applied to a matrix $\mathbf{D} = [\mathbf{d}_1, ..., \mathbf{d}_m] \in \mathbb{R}^{3n \times m}$, which includes m difference vectors $\mathbf{d}_i \in \mathbb{R}^{3n}$, produced by subtracting each expressive mesh from the neutral mesh of each corresponding sequence. Therefore, the sparse blendshape components $\mathbf{C} \in \mathbb{R}^{h\times 1}$ where recovered by the following minimisation problem:

$$\operatorname{argmin} \|\mathbf{D} - \mathbf{BC}\|_F^2 + \Omega(\mathbf{C}) \quad \text{s.t. } \mathcal{V}\left(\mathbf{B}\right), \tag{2.1}$$

where, the constraint $\mathcal{V}$ can either be $\max\left(|\mathbf{B}_k|\right) = 1, \ \forall k$ or $\max\left(\mathbf{B}_k\right) = 1, \ \mathbf{B} \geq 1, \ \forall k$, with $\mathbf{B}_k \in \mathbb{R}^{3n\times 1}$ denoting the k^{th} component of the sparse weight matrix $\mathbf{B} = [\mathbf{B}_1, \cdots, \mathbf{B}_h]$. According to [60], the selection of the constraints mainly controls whether face deformations will take place towards both negative and positive direction of the axes of the model's parameters or not, which is useful for describing shapes like muscle bulges. The regularisation of sparse components $\mathbf{C}$ was performed with $L1/L2$ norm [61, 62], while to compute optimal $\mathbf{C}$ and $\mathbf{B}$, an iterative alternating optimisation was employed. The exact same approach was employed by [59], in the construction of the 4DFAB blendshape model exploited in this work. The 5 most significant deformation components of the 4DFAB expression model are depicted in Fig. 2.1.

The blendshape models discussed above are most commonly 3D representations of the Facial Action Coding System (FACS) as specific blendshapes control specific parts of the face as they have been defined by this encoding. FACS-based blendshapes have been a popular tool the past few decades for face animation employed for film and video game production, however significant skill is required from artists to synthesize realistic animations. Other disadvantages include limited muscle separation, redundancy and difficulty in localization [63]. To overcome these issues, a common solution for animators is to employ specific additional corrective deformers which increase the total number of involved components.

Inspired by Mimic [64], an anatomically grounded language for facial deformation, authors in [65] presented Animatomy, a novel representation of 3D human face modelling and animation, aiming to solve the above problems. Animatomy models facial muscles as fiber curves and facial deformations as the contraction and relaxation of those fibers. The model is built similarly to FLAME [47] as a 3D shape decoder trained, according to the authors, with a significant amount of curated ground truth data. While FLAME is associated with PCA parameters of some principal components which have limited physical interpretation, Animatomy provide a more anatomically intuitive method to control animation based on the muscle contraction via a set of facial fibers. Their end-to-end system can be employed for automatic fitting on dynamic facial scans of a particular subject, face animation using expression transfer from an acor's performance and interactive face manipulation similarly to FACS-based blendshapes. Experiments in [65] demonstrate superior reconstruction performance of Animatomy compared to FACS-based blendshape models, while the feedback they have received in a conducted survey with professional animators is mostly positive, highlighting its representation power and ease of use.

Lastly, Animatomy is a recently published method and has not been employed by the face modelling community yet as a facial motion representation for 3D or 2D animation. FACS-based systems have been well established in the past decades with more new models emerging based on the same encoding. Nevertheless, they do not come without limitations, as for example redundancy and limited muscle disentanglement. Animatomy could provide a more fine-grained representation for animation.

2.1.2 3D Morphable Models

A 3DMM consists of three parametric models: the *shape*, *camera* and *texture* models. Here we focus on the components of the 3DMM of [13] which we have employed in our works, but the basic techniques are the same across most 3DMMs.

Shape Modelling

The shape model must be able to represent faces of different individuals and with various expressions. This is achieved by combining an identity shape model $\mathcal{S}_{id}$ which models face variation across different individuals in neutral expression, and an expression shape model $\mathcal{S}_{exp}$ which models expression as offsets from a given identity shape.

An identity shape model is built based on a collection of 3D face scans which are first brought into dense correspondence with a common shape template by an ICP-based algorithm [66, 67, 68, 69], then aligned by applying Generalised Procrustes Analysis before performing Principal Component Analysis (PCA) which results in $\{\bar{\mathbf{s}}_{id}, \mathbf{U}_{id}, \mathbf{\Sigma}_{id}\}$, where $\bar{\mathbf{s}}_{id} \in \mathbb{R}^{3N}$ is the mean shape vector, $\mathbf{U}_{id} \in \mathbb{R}^{3N \times n_{id}}$ is the orthonormal basis after keeping the first n_p principal components and $\mathbf{\Sigma}_{id} \in \mathbb{R}^{n_{id} \times n_{id}}$ is a diagonal matrix with the standard deviations of the corresponding principal components. Letting standard deviation to be absorbed by the principal components as $\widetilde{\mathbf{U}}_{id} = \mathbf{U}_{id}\mathbf{\Sigma}_{id}$, a 3D shape instance can be expressed as function of the parameters $\mathbf{p}_{id}$ as:

$$\mathcal{S}_{id}(\mathbf{p}_{id}) = \bar{\mathbf{s}}_{id} + \widetilde{\mathbf{U}}_{id}\mathbf{p}_{id}. \tag{2.2}$$

An expression blendshape shape model is built by a collection of expressive 3D face scans as described in Section 2.1.1. Similarly to $\mathcal{S}_{id}$, the expression model consists of $\{\bar{\mathbf{s}}_{exp}, \mathbf{U}_{exp}, \mathbf{\Sigma}_{exp}\}$, where $\bar{\mathbf{s}}_{exp} \in \mathbb{R}^{3N}$ is the mean expression offset, $\mathbf{U}_{exp} \in \mathbb{R}^{3N \times n_{exp}}$ is the expression basis, (which is not necessarily orthonormal) having n_{exp} principal components and $\mathbf{\Sigma}_{exp} \in \mathbb{R}^{n_{exp} \times n_{exp}}$ is the diagonal matrix with the corresponding standard deviations. Again, standard deviation is absorbed by the principal components as $\widetilde{\mathbf{U}}_{exp} = \mathbf{U}_{exp}\mathbf{\Sigma}_{exp}$. Then a 3D expression instance can be expressed as function of the parameters $\mathbf{p}_{exp}$ as:

$$\mathcal{S}_{exp}(\mathbf{p}_{exp}) = \bar{\mathbf{s}}_{exp} + \widetilde{\mathbf{U}}_{exp}\mathbf{p}_{exp}. \tag{2.3}$$

Combining the models requires them to be in correspondence with a common reference template. Assuming this is the case, the combined facial shape model of identity and expression is written as follows:

$$\begin{aligned} \mathcal{S}(\mathbf{p}) = \mathcal{S}(\mathbf{p}_{id}, \mathbf{p}_{exp}) &= \bar{\mathbf{s}} + \widetilde{\mathbf{U}}_{id}\mathbf{p}_{id} + \widetilde{\mathbf{U}}_{exp}\mathbf{p}_{exp} \\ &= \bar{\mathbf{s}} + [\widetilde{\mathbf{U}}_{id}, \widetilde{\mathbf{U}}_{exp}][\mathbf{p}_{id}^{\mathsf{T}}, \mathbf{p}_{exp}^{\mathsf{T}}]^{\mathsf{T}} \\ &= \bar{\mathbf{s}} + \widetilde{\mathbf{U}}\mathbf{p}, \end{aligned} \tag{2.4}$$

where $\bar{\mathbf{s}} = \bar{\mathbf{s}}_{id} + \bar{\mathbf{s}}_{exp}$ is the overall mean shape, $\mathbf{p}_{id}$ is the vector with the identity parameters, $\mathbf{p}_{exp}$ is the vector with the expression parameters, $\widetilde{\mathbf{U}}$ is the matrix of combined identity and expression components and $\mathbf{p}$ the vector of the respective combined parameters.

Camera Modelling

The camera model projects a 3D mesh instance $\mathbf{s}$ from the 3D Cartesian coordinates into the 2D Cartesian coordinates on an image plane. In more detail, a 3D point $\mathbf{x} = [x, y, z]^{\mathsf{T}}$ is transformed to a 2D location $\mathbf{x}' = [x', y']^{\mathsf{T}}$ in the image plane, by first applying a view transformation such that $\mathbf{v} = [v_x, v_y, v_z]^{\mathsf{T}} = \mathbf{R}_v\mathbf{x} + \mathbf{t}_v$, where $\mathbf{R}_v \in \mathbb{R}^{3\times3}$ and $\mathbf{t}_v = [t_x, t_y, t_z]^{\mathsf{T}}$ are the camera's 3D rotation and translation components, respectively. Then a camera projection is applied as $\mathbf{x}' = \pi(\mathbf{c}_{\text{intr}}, \mathbf{v})$, where $\mathbf{c}_{\text{intr}}$ is a vector with the camera's intrinsic parameters and $\pi(\mathbf{c}_{\text{intr}}, \mathbf{v})$ is a perspective or an orthographic camera model. To ensure computational efficiency, robustness and simpler differentiation, rotation matrix $\mathbf{R}_v$ can be parameterised by quartenions instead of Euler angles. The projection operation performed on the points of a 3D mesh $\mathbf{s}$ by the camera model can be expressed as a function $\mathcal{P}(\mathbf{s},\mathbf{c}) : \mathbb{R}^{3N} \to \mathbb{R}^{2N}$, where $\mathbf{c} = \left[\mathbf{c}_{intr}^{\mathsf{T}}, \mathbf{q}^{\mathsf{T}}, \mathbf{t}^{\mathsf{T}}\right]^{\mathsf{T}}$ is the vector of camera parameters. When $\mathbf{s}$ is an instance $\mathcal{S}(\mathbf{p})$ of the 3D shape model of the face, the camera function can also be written as:

$$\mathcal{W}(\mathbf{p},\mathbf{c}) \equiv \mathcal{P}\left(\mathcal{S}(\mathbf{p}),\mathbf{c}\right). \tag{2.5}$$

Texture Modelling

Contrary to texture models built based on texture from 3D face scans captured in controlled environments [70, 44, 41], building feature-based texture models out of "in-the-wild" facial images has allowed to avoid the estimation of illumination parameters during fitting, while it has also made fitting robust to occlusions [13].

To build a feature-based texture model, first a dense feature extraction function is defined as $\mathcal{F} : \mathbb{R}^{H\times W\times N_{\text{colors}}} \to \mathbb{R}^{H\times W\times C}$, where C is the number of channels of the feature-based image, and applied on a collection of "in-the-wild" facial images $\{\mathbf{I}_i\}_1^M$. Then, assuming that shape and camera parameters $\{\mathbf{p},\mathbf{c}_i\}$ are available for each image by fitting the combined shape model on sparse face landmarks, texture samples can be extracted by sampling the feature-based image representations at the vertices of the projected 3D meshes, forming vectors $\mathbf{t}_i = \mathbf{F}_i\left(\mathcal{W}(\mathbf{p}_i,\mathbf{c}_i)\right) \in \mathbb{R}^{CN}$.

As texture samples $\mathbf{t}_i$ include missing information and gross but sparse non-Gaussian errors caused mainly by self-occlusions inherent in facial data, a texture reconstruction step needs to take place before building the final model. That is, the Principal Component Pursuit problem [71] is solved to recover a low-rank matrix $\mathbf{L} \in \mathbb{R}^{CN\times M}$ representing the clean facial texture and a sparse matrix $\mathbf{E} \in \mathbb{R}^{CN\times M}$ accounting for gross but sparse non-Gaussian noise such that $\mathbf{X} = \mathbf{L} + \mathbf{E}$, where $\mathbf{X} = [\mathbf{t}_1,\ldots,\mathbf{t}_M] \in \mathbb{R}^{CN\times M}$ is a matrix including the concatenated M feature-based texture vectors.

The final texture model is created by applying PCA on $\mathbf{L}$, the set of reconstructed feature-based textures. This results in $\{\bar{\mathbf{t}},\mathbf{U}_t\}$, where $\bar{\mathbf{t}} \in \mathbb{R}^{CN}$ is the mean texture vector and $\mathbf{U}_t \in \mathbb{R}^{CN\times n_t}$ is the orthonormal basis after keeping the first n_t principal components. This model can be used to generate novel 3D feature-based texture instances based on texture parameters $\boldsymbol{\lambda} = [\lambda_1,\ldots,\lambda_{n_t}]^{\mathsf{T}}$ with the function $\mathcal{T} : \mathbb{R}^{n_t} \to \mathbb{R}^{CN}$ as:

$$\mathcal{T}(\boldsymbol{\lambda}) = \bar{\mathbf{t}} + \mathbf{U}_t\boldsymbol{\lambda}. \tag{2.6}$$

3D Morphable Model fitting

3DMMs of the face are employed to reconstruct 3D facial meshes from input images. To do so, an optimisation problem needs to be formed and solved for each input image $\mathbf{I}$, in order to recover parameters $\{\mathbf{p}_{id}, \mathbf{p}_{exp}, \mathbf{c}\}$ which reconstruct a model instance that best represents the face in image $\mathbf{I}$. In particular, in [13] the authors, based on the extensive literature in Lucas-Kanade 2D image alignment [72, 73, 74, 75, 76, 77], form a Gauss-Newton-style minimisation problem as:

$$\min_{\mathbf{p},\mathbf{c}} \left\|\mathbf{F}\left(\mathcal{W}(\mathbf{p},\mathbf{c})\right) - \mathcal{T}(\boldsymbol{\lambda})\right\|^2 + c_l \left\|\mathcal{W}_l(\mathbf{p},\mathbf{c}) - \boldsymbol{\ell}\right\|^2 + c_p \left\|\mathbf{p}\right\|^2, \tag{2.7}$$

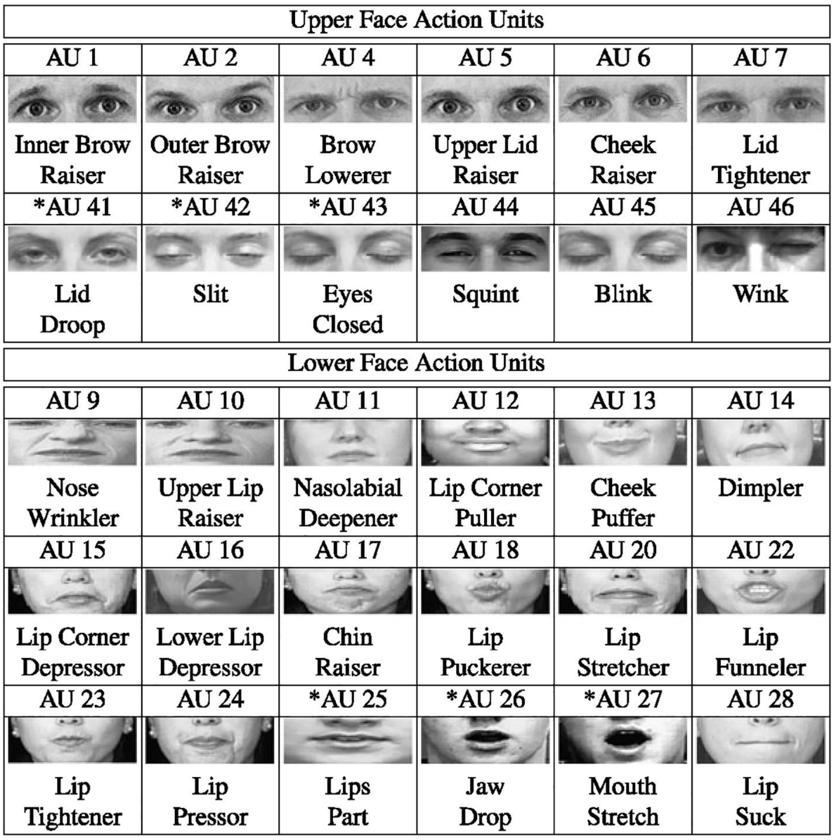
where the first term is a texture term depending on shape, texture and camera parameters and penalises the squared L^2 norm of the difference between the image feature-based texture that corresponds to the projected 2D locations of the 3D shape instance and the texture instance of the 3DMM, the second term is a sparse landmarks term defined on the image coordinate system and aims to drive the optimisation procedure using the selected sparse landmarks as anchors, and the last term is a parameter regularisation term employed to avoid over-fitting.

The problem of (2.7) is solved by adopting a project-out optimisation approach, which enables optimisation on the orthogonal complement of the texture subspace. This eliminates texture parameter increments at each iteration and makes the process faster than the more widely-used Simultaneous algorithm [75, 78, 33]. More details on the optimisation algorithm can be found in [13].

2.1.3 Facial Expression Coding with AUs and Expression Blendshapes

Facial Action Units (AUs) [79] coding constitutes a comprehensive approach for quantifying facial motion, which is based on identifying the activation of individual muscles of the human face. Particularly, each AU corresponds to a specific muscle of the human face, while the level of activation of each AU can also be measured. In total there exist 44 Action Units, some of which are depicted in Figure 2.2. The level of detail offered by AU coding makes it ideal for defining rigorous representations of facial expressions and emotions. Because of that, multiple works have employed AUs for training emotion and expression recognition systems [80, 81, 82, 83, 84].

However, identifying the activated AUs in a facial image is a tedious task which requires particular expertise. This makes acquiring AU annotations for large datasets of facial images costly and inefficient. Moreover, automatic Action Unit detection is currently an open problem both in controlled, as



Upper Face Action Units					
AU 1	AU 2	AU 4	AU 5	AU 6	AU 7
Inner Brow Raiser	Outer Brow Raiser	Brow Lowerer	Upper Lid Raiser	Cheek Raiser	Lid Tightener
*AU 41	*AU 42	*AU 43	AU 44	AU 45	AU 46
Lid Droop	Slit	Eyes Closed	Squint	Blink	Wink

Lower Face Action Units					
AU 9	AU 10	AU 11	AU 12	AU 13	AU 14
Nose Wrinkler	Upper Lip Raiser	Nasolabial Deepener	Lip Corner Puller	Cheek Puffer	Dimpler
AU 15	AU 16	AU 17	AU 18	AU 20	AU 22
Lip Corner Depressor	Lower Lip Depressor	Chin Raiser	Lip Puckerer	Lip Stretcher	Lip Funneler
AU 23	AU 24	*AU 25	*AU 26	*AU 27	AU 28
Lip Tightener	Lip Pressor	Lips Part	Jaw Drop	Mouth Stretch	Lip Suck

Figure 2.2: Action Units describing the motion of specific facial muscles in the human face. Individual AUs focus on the motion of specific muscles of the face, allowing the encoding of complex expressions using distinct and exact AU combinations. A drawback of AU coding is that expert knowledge is required to extract AU annotations. Image is taken from [88].

well as in unconstrained recording conditions. In particular, recent AU detection techniques achieve around 50% F1 in EmotioNet challenge and from our experiments OpenFace [85] achieves lower than 20-25% [86, 87]. Another drawback of automatic AU detection is that it can be performed for specific subsets of AUs only, not covering the full range of possible facial motion (e.g. motion of the lips).

On the contrary, simpler, more flexible, intuitive and automatically identifiable representations of facial motion would make a better fit for applications such as expression recognition, facial animation and expression editing. 3D expression blendshapes constitute such a representation of facial motion which is already a standard for 3D animation. They can be manually sculpted by 3D artists or built from collections of expressive 3D face scans as described in Section 2.1.1, which results in more realistic facial deformations. An example of a facial deformation blendshape model is the model of 4DFAB [59] which is presented in Figure 2.1. In comparison to AUs, blendshape components

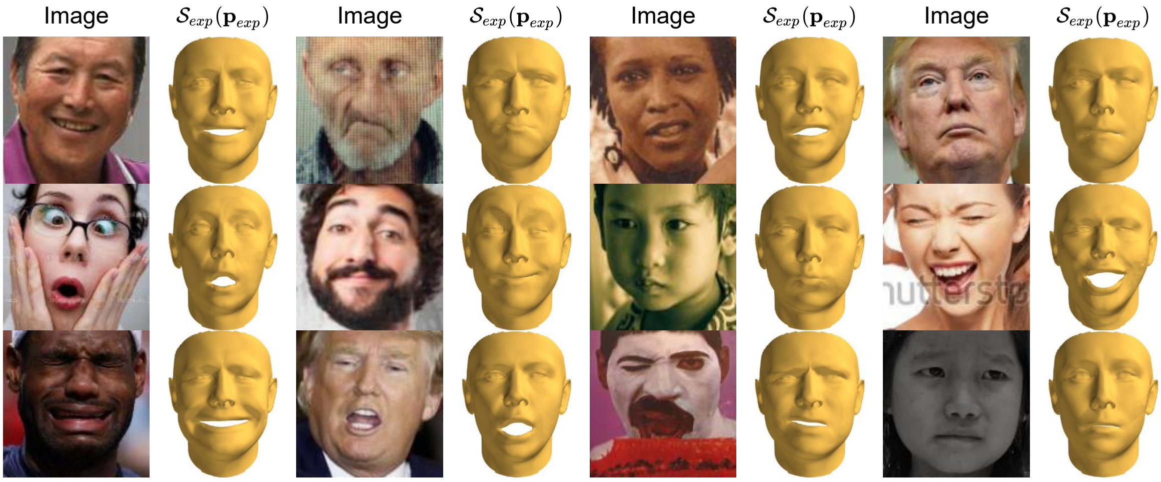
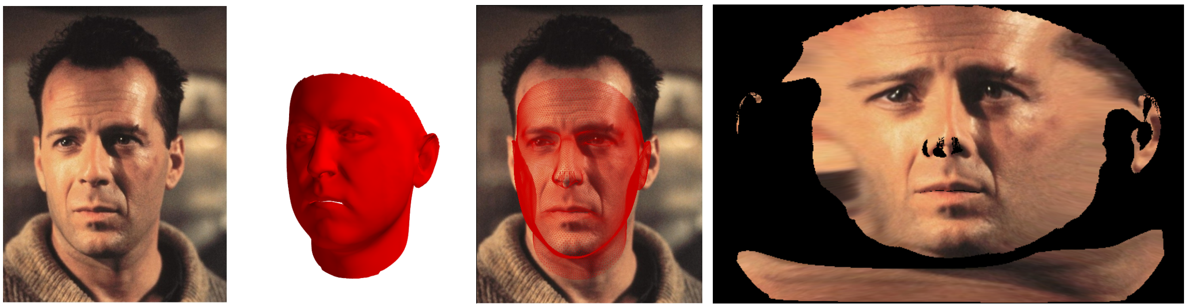


Image
$\mathcal{S}_{exp}(\mathbf{p}_{exp})$
Image
$\mathcal{S}_{exp}(\mathbf{p}_{exp})$
Image
$\mathcal{S}_{exp}(\mathbf{p}_{exp})$
Image
$\mathcal{S}_{exp}(\mathbf{p}_{exp})$

Figure 2.3: 3D representation of expression in images, given by the blendshape model $\mathcal{S}_{exp}$ and parameters $\mathbf{p}_{exp}$ recovered by fitting the 3DMM of [13] on the corresponding images. The 3D reconstructions demonstrate that 3D blendshape coding can be a useful expression embedding for face images. Images are taken from [29].

correspond to possible motions of the human face (for example, opening the mouth, squinting, raising eyebrows, moving mouth and nose together from side to side) which might involve multiple facial muscles, while AUs correspond to specific muscles of the face. One advantage of 3D blendshapes over AUs is that manipulating expressions or creating new ones does not require expert knowledge, as the edited 3D face instance can be directly observed within a 3D viewer. Moreover, automatic estimation of expression blendshapes from facial images has been an active research topic with significant reported advancements [13, 12, 53, 54].

In the works of Chapter 3 and Chapter 4 of this Thesis, we have adopted the method of [13] to recover representation from images based on 3D expression blendshape coding. In Chapter 3, the technique serves as a means to recover aligned UV map representations from expressive "in-the-wild" facial images, while in Chapter 4 as a method to extract pseudo-annotations about expression which are employed to train a system for expression editing in images. Particularly, by fitting the 3DMM of [13] in an input image $\mathbf{I}$, we can extract identity and expression parameters $\mathbf{p}_{id}$ and $\mathbf{p}_{exp}$ that instantiate the recovered 3D face mesh $\mathcal{S}(\mathbf{p}_{id}, \mathbf{p}_{exp})$, as described in Section 2.1.2. Based on the independent shape parameters for identity and expression, we exploit parameters $\mathbf{p}_{exp}$ to compose an annotated dataset of images and their corresponding vector of expression parameters $\{\mathbf{I}^i, \mathbf{p}^i_{exp}\}_{i=1}^{K}$, with no manual annotation cost. Examples of images and the 3D instances recovered by the above procedure and



(a) (b) (c) (d)

Figure 2.4: Generation of a UV texture map with missing values from a standard 2D image. Given an image (a), we recover the 3D facial shape by fitting a 3DMM on it (b) and sample texture at the locations of the projected vertices (c). In this way, we compute the UV texture map along with the occluded parts (d).

including only expression deformation are depicted in Figure 2.3.

2.1.4 Dataset Alignment in UV Spaces

A 3D mesh can be represented by a matrix $\mathbf{X} = [\mathbf{x_1}^\mathsf{T}, \mathbf{x_2}^\mathsf{T}, \ldots, \mathbf{x_N}^\mathsf{T}]^\mathsf{T} \in \mathbb{R}^{3N}$ of N vertices $\mathbf{x}_i = [x_x^i, x_y^i, x_z^i] \in \mathbb{R}^3$ and a list of triangles $\mathbf{T} = [\mathbf{t_1}^\mathsf{T}, \mathbf{t_2}^\mathsf{T}, \ldots, \mathbf{t_M}^\mathsf{T}]^\mathsf{T}$, where the triad $\mathbf{t}_i = [t_1^i, t_2^i, t_3^i]$, $t_j^i \in \{\mathbb{Z}^+ \mid t_j^i \leq N\}$ indicates the vertices forming triangle i. Moreover, a mesh's texture is given as a 2D image $\mathbf{I}$ from which colour is sampled using specific texture coordinates $\mathbf{C} = [\mathbf{c_1}^\mathsf{T}, \mathbf{c_2}^\mathsf{T}, \ldots, \mathbf{c_N}^\mathsf{T}]^\mathsf{T} \in \mathbb{R}^{2N}$, where $\mathbf{c}_i = [c_u^i, c_v^j] \in \mathbb{R}^2$, which map vertices of the mesh to specific 2D image locations.

Texture coordinates $\mathbf{C}$ essentially constitute 2D representations of 3D meshes. Forming continuous 2D representations of meshes enables us to process 3D meshes using image-based techniques. UV spaces $\mathbf{U} \subset \mathbb{R}^2$ are such continuous spaces for which a bijective mapping exists between $\mathbf{X}$ and $\mathbf{U}$ such that $f(\mathbf{x}) \mapsto \mathbf{u}$ and $f^{-1}(\mathbf{u}) \mapsto \mathbf{x}$ for all $\mathbf{u} \in \mathbf{U}$, meaning that exact correspondence exists between the 3D coordinates of a mesh and the 2D coordinates of a UV space. Cylindrical and spherical coordinate projections are commonly used to define UV spaces for facial meshes.

Having established a UV space mapping, texture of 3D meshes can also be projected into that space forming continuous UV maps [89], which are suitable for processing using standard image processing and parameterisation techniques. For example, aligning 3D face scans with specific mesh templates have been achieved by aligning the UV map images using 2D image alignment techniques instead

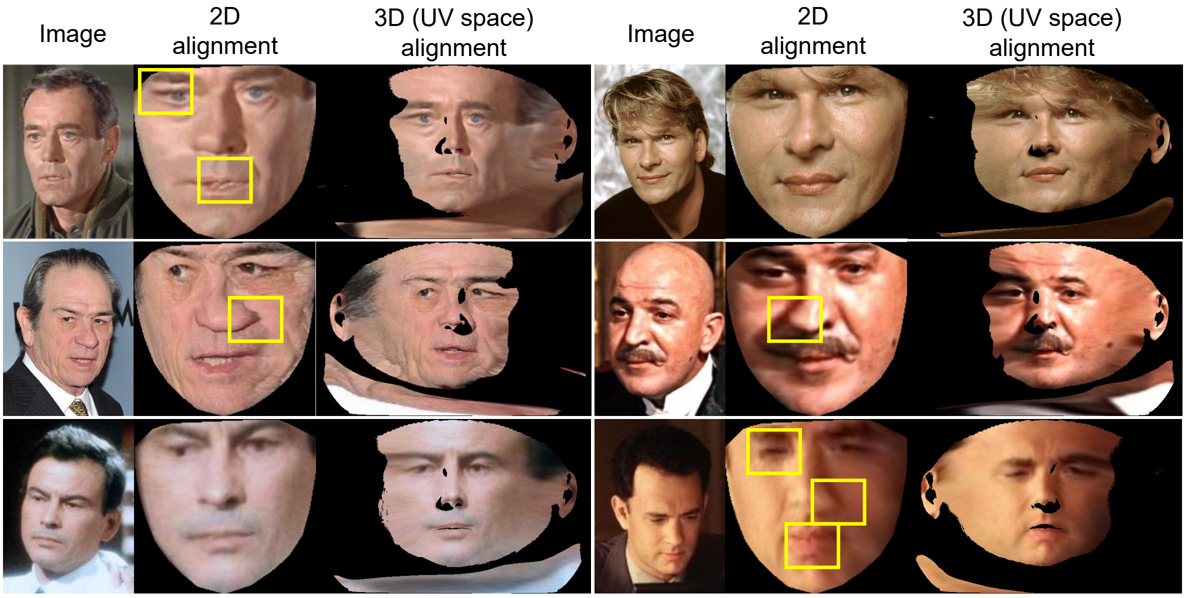


Figure 2.5: Images aligned in a common 2D shape defined by sparse 2D facial landmarks and in a common UV space defined by unwrapping the template of the LSFM [44] 3D face model. Even thought texture is sampled and distributed by the same mechanism of piece-wise affine transformations, 3D fitting of a dense 3D model on images results in aligned representations with fewer artifacts. In yellow rectangles are highlighted some artifacts of 2D alignment.

of directly working in 3D [44, 24]. Besides, when exact correspondence exists between a set of 3D meshes, the corresponding UV texture maps are also rigorously aligned, making them suitable for building statistical texture models by applying Component Analysis techniques.

Texture acquired from 3D scanning devices can be directly projected to UV spaces forming complete UV maps. However, aligning standard 2D face images in UV space is also possible by fitting a 3DMM on “in-the-wild” images and sampling them at the locations of the projected vertices. Then, the sampled textures are projected on UV space using the texture coordinates $\mathbf{C}$. The steps of this process are presented in Figure 2.4. As it can be seen, UV maps recovered from natural facial images include missing values (depicted with black) along with sparse, non-Gaussian noise caused by occlusions, accessories and capturing noise. On the contrary, aligning images in UV space provides the advantage of having less warping effects in comparison to 2D image alignment methods, as shown in Figure 2.5.

In our work presented in Chapter 3 of this Thesis, we employ UV texture maps that are precisely mapped to 3D models of human faces and particularly to instances of the LSFM face model [44], as for example the ones depicted in Figure 2.5. There, we develop robust, linear models of texture, which we employ for facial image reconstruction and editing, leveraging the benefits and overcoming the challenges posed by UV maps.

2.2 Component Analysis Methods for Face Attribute Analysis and Editing

Component Analysis (CA) methods are useful to discover important underlying structures in all sorts of data including datasets of facial images. Structures discovered through these methods can be employed for downstream tasks by discriminative models or for image generation and editing as in RJIVE [27]. Undoubtedly, Principal Component Analysis (PCA) [90, 91, 92], is the most widely used method for dimensionality reduction. PCA projects the data in a lower dimension space maintaining the maximum variance between them. Focusing on extracting joint components among data, the Canonical Correlation Analysis (CCA) [93] can be employed to extract linear correlated components among two or more sets of variables, while the inter-battery factor analysis [94] also determines the common factors among two sets of variables.

To discover both joint and individual structures in datasets, the Joint and Individual Variation Explained (JIVE) [14], the Common Orthogonal Basis Extraction (COBE) [15], and the Robust Correlated and Individual Component Analysis (RCICA) [16] were proposed. In this Section, we briefly review the three methods, as they are the most closely related methods to our CA method, the RJIVE, presented in Chapter 3.

2.2.1 Joint and Individual Variation Explained (JIVE)

The JIVE recovers the joint and individual components among $M \geq 2$ data-sets $\{\mathbf{X}^{(i)} \in \mathbb{R}^{d^{(i)} \times J}, i = 1, 2, \ldots, M\}$, where J is the number of samples of each data-set. In particular, each matrix is decomposed into two terms: a low-rank matrix $\mathbf{J}^{(i)} \in \mathbb{R}^{d^{(i)} \times J}$ capturing joint structure between data-sets and a low-rank matrix capturing individual structure $\mathbf{A}^{(i)} \in \mathbb{R}^{d^{(i)} \times J}$ to each data-set. That is,

$$\mathbf{X}^{(i)} = \mathbf{J}^{(i)} + \mathbf{A}^{(i)}, i = 1, 2, \ldots, M. \tag{2.8}$$

Let $\mathbf{X}$ and $\mathbf{J}$, be $\sum_{i=1}^{M} d^{(i)} \times J$ matrices constructed by the concatenation of the corresponding matrices i.e., $\mathbf{X} = [\mathbf{X}^{(1)^\mathsf{T}}, \mathbf{X}^{(2)^\mathsf{T}}, \ldots, \mathbf{X}^{(M)^\mathsf{T}}]^\mathsf{T}$, $\mathbf{J} = [\mathbf{J}^{(1)^\mathsf{T}}, \mathbf{J}^{(2)^\mathsf{T}}, \ldots, \mathbf{J}^{(M)^\mathsf{T}}]^\mathsf{T}$, the JIVE solves the rank-constrained least-squares problem [14]:

$$\begin{aligned} \min_{\mathbf{J}, \{\mathbf{A}^{(i)}\}_{i=1}^{M}} \quad & \frac{1}{2} \left\| \mathbf{X} - \mathbf{J} - \left[\mathbf{A}^{(1)^\mathsf{T}}, \cdots, \mathbf{A}^{(M)^\mathsf{T}} \right]^\mathsf{T} \right\|_F^2 . \\ \text{s.t.} \quad & \operatorname{rank}(\mathbf{J}) = r, \{ \operatorname{rank}(\mathbf{A}^{(i)}) = r^{(i)}, \mathbf{J}\mathbf{A}^{(i)^\mathsf{T}} = \mathbf{0} \}_{i=1}^{M} \end{aligned} \tag{2.9}$$

Problem (2.9) imposes rank constraints on joint and individual components and requires the rows of $\mathbf{J}$ and $\{\mathbf{A}^{(i)}\}_{i=1}^{M}$ to be orthogonal. The intuition behind the orthogonality constraint is that, sample patterns responsible for joint structure between data types are unrelated to sample patterns responsible for individual structure [14]. By adopting the least squares error, the JIVE assumes Gaussian distributions with small variance [95]. Such an assumption rarely holds in real word data, where gross non-Gaussian corruptions are in abundance. Consequently, the components obtained by employing the JIVE in the analysis of grossly corrupted data may be arbitrarily away from the true ones, degenerating their performance.

2.2.2 Common Orthogonal Basis Extraction (COBE)

A closely related method to the JIVE is the COBE which extracts the common and the individual components from M data-sets of the same dimensions by solving a set of least-squares minimisation problems [15]. More specifically, each data-set $\mathbf{X}^{(i)} \in \mathbb{R}^{J \times d^{(i)}}$ is factorised as $\mathbf{A}^{(i)}\mathbf{B}^{(i)^\mathsf{T}}$ where a column of $\mathbf{A}^{(i)}$ signifies a latent variable to be found and $\mathbf{B}^{(i)}$ signifies a matrix of weights. $\mathbf{A}^{(i)}$ is assumed to be decomposable in blocks as $\left[\bar{\mathbf{A}}\tilde{\mathbf{A}}^{(i)} \right]$ where $\bar{\mathbf{A}} \in \mathbb{R}^{n \times m}$, $\tilde{\mathbf{A}}^{(i)} \in \mathbb{R}^{n \times (d^{(i)} - m)}$ and $m \leq \min\{d^{(i)}, i = 1, \cdots, M\}$. In other words, $\bar{\mathbf{A}}$ is assumed to be common to all factorisations and hence it presents joint structure while $\tilde{\mathbf{A}}^{(i)}$ is assumed to represent individual structure. Similarly, $\mathbf{B}^{(i)}$ splits as $\bar{\mathbf{B}}^{(i)}$ and $\tilde{\mathbf{B}}^{(i)}$. The optimisation problem of the COBE takes the following form:

$$\begin{aligned} \min_{\bar{\mathbf{A}}, \tilde{\mathbf{A}}^{(i)}} \quad & \sum_{i=1}^{M} \left\| \mathbf{X}^{(i)} - \bar{\mathbf{A}}\bar{\mathbf{B}}^{(i)^\mathsf{T}} - \tilde{\mathbf{A}}^{(i)}\tilde{\mathbf{B}}^{(i)^\mathsf{T}} \right\|_F^2 . \\ \text{s.t.} \quad & \bar{\mathbf{A}}^\mathsf{T}\bar{\mathbf{A}} = \mathbf{I}, \{ \tilde{\mathbf{A}}^{(i)^\mathsf{T}}\tilde{\mathbf{A}}^{(i)} = \mathbf{I}, \bar{\mathbf{A}}^\mathsf{T}\tilde{\mathbf{A}}^{(i)^\mathsf{T}} = \mathbf{0} \}_{i=1}^{M} . \end{aligned} \tag{2.10}$$

Similarly to the JIVE, the usage of the least square error makes the COBE non-robust against sparse, non-Gaussian errors.

2.2.3 Robust Correlated and Individual Component Analysis (RCICA)

The goal of the RCICA [16] is to extract both the correlated and the individual components between two known high-dimensional datasets or views namely, $\{\mathbf{X}^{(i)} \in \mathbb{R}^{d^{(i)} \times J}\}_{i=1}^{2}$, in the presence of sparse noise (or errors). To this end, the RCICA seeks a decomposition of each data matrix $\{\mathbf{X}^{(i)}\}$ into three terms:

$$\mathbf{X}^{(i)} = \mathbf{C}^{(i)} + \mathbf{A}^{(i)} + \mathbf{E}^{(i)}, \quad i = 1, 2. \tag{2.11}$$

where $\mathbf{C}^{(i)} \in \mathbb{R}^{d^{(i)} \times J}$ and $\mathbf{A}^{(i)} \in \mathbb{R}^{d^{(i)} \times J}$ are low-rank matrices, with rank($\mathbf{C}^{(i)}$) $\leq k_c$ and rank($\mathbf{A}^{(i)}$) $\leq k^{(i)}$ and mutually independent columns, capturing the correlated and individual components, respectively and $\mathbf{E}^{(n)} \in \mathbb{R}^{d^{(i)} \times J}$ is a sparse matrix accounting for the sparse noise.

To find the correlated components $\mathbf{C}^{(i)} \in \mathbb{R}^{d^{(i)} \times J}$, the cost function of the Canonical Correlation Analysis (CCA) [93] is adopted. That is, by further decomposing the matrix $\{\mathbf{C}^{(i)}\}_{i=1}^{2}$ as: $\mathbf{C}^{(i)} = \mathbf{U}^{(i)}\ {\mathbf{V}^{(i)}}^{\mathsf{T}}\ \mathbf{X}^{(i)}$, the maximally correlated components are derived by minimising the CCA cost, namely $\frac{\lambda_c}{2}\|{\mathbf{V}^{(1)}}^{\mathsf{T}}\mathbf{X}^{(1)} - {\mathbf{V}^{(2)}}^{\mathsf{T}}\mathbf{X}^{(2)}\|_F^2$. Here, $\mathbf{U}^{(i)}$ are orthonormal basis, transforming the correlated components back to the observation space $\mathbf{X}^{(i)}$. Since, the column space of the individual components $\mathbf{A}^{(i)}$ is desired to be orthogonal to the one of the correlated components we have to enforce $\{{\mathbf{Q}^{(i)}}^{\mathsf{T}}\mathbf{U}^{(i)}\}_{i=1}^{2} = \mathbf{0}$, where $\mathbf{Q}^{(i)}$ are column orthonormal basis spanning the column space of the individual components $\mathbf{A}^{(i)}$, that is $\mathbf{A}^{(i)} = \mathbf{Q}^{(i)}\ \mathbf{H}^{(i)}$.

Consequently, a natural estimator accounting for the upper-bounded rank of the correlated and independent components and the sparsity of $\{\mathbf{E}^{(i)}\}_{i=1}^{2}$ is to minimise the objective function of CCA, i.e., $\frac{1}{2}\|{\mathbf{V}^{(1)}}^{\mathsf{T}}\mathbf{X}^{(1)} - {\mathbf{V}^{(2)}}^{\mathsf{T}}\mathbf{X}^{(2)}\|_F^2$ as well as the rank of $\{\mathbf{C}^{(i)} = \mathbf{U}^{(i)}\ {\mathbf{V}^{(i)}}^{\mathsf{T}}\ \mathbf{X}^{(i)}, \mathbf{A}^{(i)} = \mathbf{Q}^{(i)}\ \mathbf{H}^{(i)}\}_{i=1}^{2}$ and the number of nonzero entries of $\{\mathbf{E}^{(i)}\}_{i=1}^{2}$ measured by the ℓ_0-(quasi) norm, e.g., [96].

To avoid the NP-hardness of rank and ℓ_0-norm minimisation,the nuclear- and the ℓ_1- norms are typically adopted as surrogates to rank and ℓ_0- norm, respectively [97, 98]. By employing the unitary invariance of the nuclear norm e.g., $\|\mathbf{Q}^{(i)}{\mathbf{V}^{(i)}}^{\mathsf{T}}\|_* = \|{\mathbf{V}^{(i)}}^{\mathsf{T}}\|_*$ the optimisation problem of RCICA is formulated as the following constrained non-linear one:

$$
\begin{aligned}
\min_{\mathcal{V}} \quad & \sum_{i=1}^{2} \Big[\|\mathbf{V}^{(i)^{\mathsf{T}}}\|_* + \lambda_*^{(i)} \|\mathbf{H}^{(i)}\|_* + \lambda_1^{(i)} \ \|\mathbf{E}^{(i)}\|_1 \Big] \\
& \qquad + \frac{\lambda_c}{2} \|\mathbf{V}^{(1)^{\mathsf{T}}}\mathbf{X}^{(1)} - \mathbf{V}^{(2)^{\mathsf{T}}}\mathbf{X}^{(2)}\|_F^2, \\
\text{s.t.} \quad & (i) \ \ \mathbf{X}^{(i)} = \mathbf{U}^{(i)}\mathbf{V}^{(i)^{\mathsf{T}}}\mathbf{X}^{(i)} + \mathbf{Q}^{(i)}\mathbf{H}^{(i)} + \mathbf{E}^{(i)} \\
& (ii) \ \ \mathbf{V}^{(i)^{\mathsf{T}}}\mathbf{X}^{(i)}\mathbf{X}^{(i)^{\mathsf{T}}}\mathbf{V}^{(i)} = \mathbf{I}, \\
& (iii) \ \ \mathbf{U}^{(i)^{\mathsf{T}}}\mathbf{U}^{(i)} = \mathbf{I}, \ \ \mathbf{Q}^{(i)^{\mathsf{T}}}\mathbf{Q}^{(i)} = \mathbf{I}, \\
& (iv) \ \ \mathbf{Q}^{(i)^{\mathsf{T}}}\mathbf{U}^{(i)} = \mathbf{0}, \ \ i = 1, 2,
\end{aligned}
\tag{2.12}
$$

where the positive parameters λ_c, $\lambda_*^{(1)}$, $\lambda_*^{(2)}$, $\lambda_1^{(1)}$ and $\lambda_1^{(2)}$ control the correlation, rank and sparsity of the derived spaces and $\mathcal{V} = \{\mathbf{U}^{(i)}, \mathbf{V}^{(i)}, \mathbf{Q}^{(i)}, \mathbf{H}^{(i)}, \mathbf{E}^{(i)}\}_{i=1}^2$ collects the optimisation variables. The constraints (ii) in (2.12) have been adopted from the CCA [93] while the constraints (iii) and (iv) ensure that both the recovered correlated and individual components are linearly independent.

Although the RCICA is robust to sparse, non-Gaussian error, its extension to more than two datasets is not trivial due to the orthogonality among the correlated and individual components and column of orthonormality of the basis matrices $\mathbf{U}^{(i)}$ and $\mathbf{Q}^{(i)}$, $i = 1, 2, \dots M$, with M being the number of different views. This makes the resulting optimisation problem highly-nonlinear and hence difficult to solve.

2.3 Image-to-Image Translation for Face Attribute Analysis and Synthesis

Component Analysis methods identify low-rank spaces of certain variations in facial data, which can be employed for downstream tasks (e.g. age group classification, identity verification, etc.) or image reconstruction and facial attribute manipulation (e.g. age group, identity, expression transfer etc.). Nevertheless, due to the low-rank nature of the recovered structures, CA-based image reconstruction and editing cannot produce photorealistic results with high frequency details, such as face wrinkles, detailed eyes and teeth, etc.

Undoubtedly, Generative Adversarial Networks (GANs) [1] have recently revolutionised the field of Computer Vision and especially generative tasks including but not limited to image generation.

Unlike CA methods, GAN-generated images can achieve high levels of photorealism [2], influencing numerous image generation and editing applications, including image-to-image translation. Image-to-image (i2i) translation refers to the task of transferring a given image from a source domain X to a target domain Y. For example, i2i translation tasks include, but are not limited to, transferring sketches to images, black and white images to coloured ones, label maps to realistic images, paintings to images, neutral faces to expressive ones, etc.

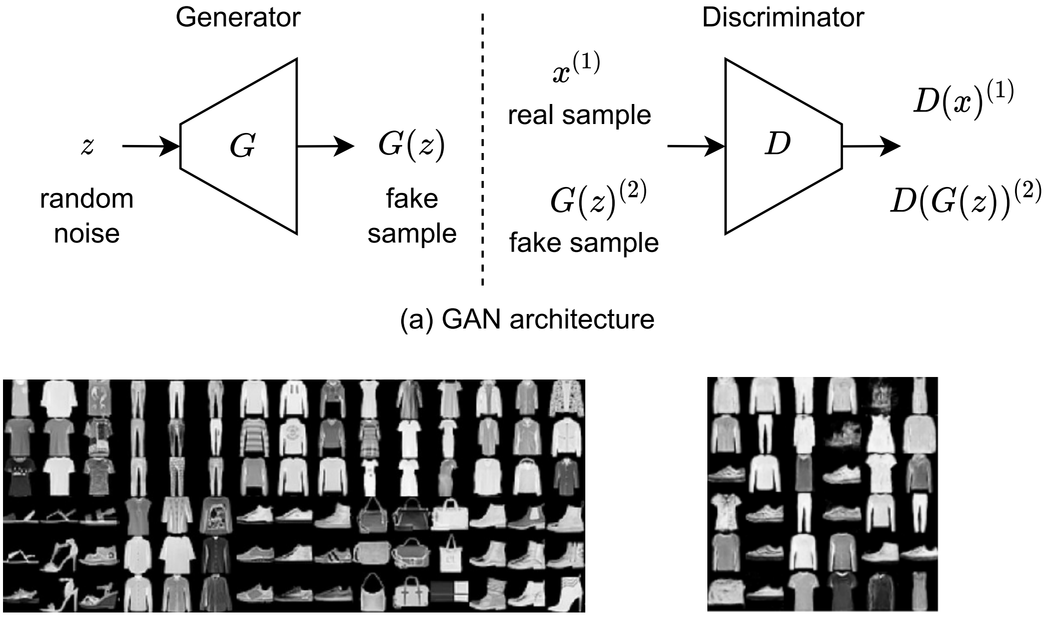
In this line of research, pix2pix [17] was one of the first methods to leverage the architecture of the conditional GANs (cGANs) [26] to carry out the task of i2i translation. In pix2pix source images operate as the condition of the generator and discriminator networks. Utilising pairs of images in two different domains for training, pix2pix is able to transfer input images from a source to a target domain. Following pix2pix, more i2i translation models where proposed to transfer images between two or more distinct image domains [18, 19, 99], but soon more flexible methods for image generation based on continuous domain codes were introduced [20, 29].

Arguably, face attribute editing in images is most often expressed as i2i translation based on an input image and additional attribute specific codes. In the rest of this Section, we provide concise information on GANs, conditional GANs, as well as some of the most important i2i translation methods, which have constituted the basis for our face expression editing method, SliderGAN [29], presented in detail in Chapter 4.

2.3.1 Generative Adversarial Networks

GANs normally consist of two modules, namely the generator G and the discriminator D, trained to optimise two competing tasks. That is, given a vector of random noise z, the generator tries to produce images as close as possible to the ones from a given distribution, while the goal of the discriminator is to correctly classify images as real or fake, i.e. generated by G. By the end of the training process, ideally, D cannot tell the difference between the two sources of images. The competing goals optimised during training is ultimately the reason for the characterisation “Adversarial” to this class of generative models. A visual representation of a typical GAN is presented in Figure 2.6 (a).

In mathematical terms, assuming p_{data}, p_{gen}, p_z are distributions of the real data, of the generated ones and of random noise z, and x is a given real data point, the competitive task optimised by the



(a) GAN architecture

(b) Fashion-MNIST real images

(c) Fashion-MNIST generated images

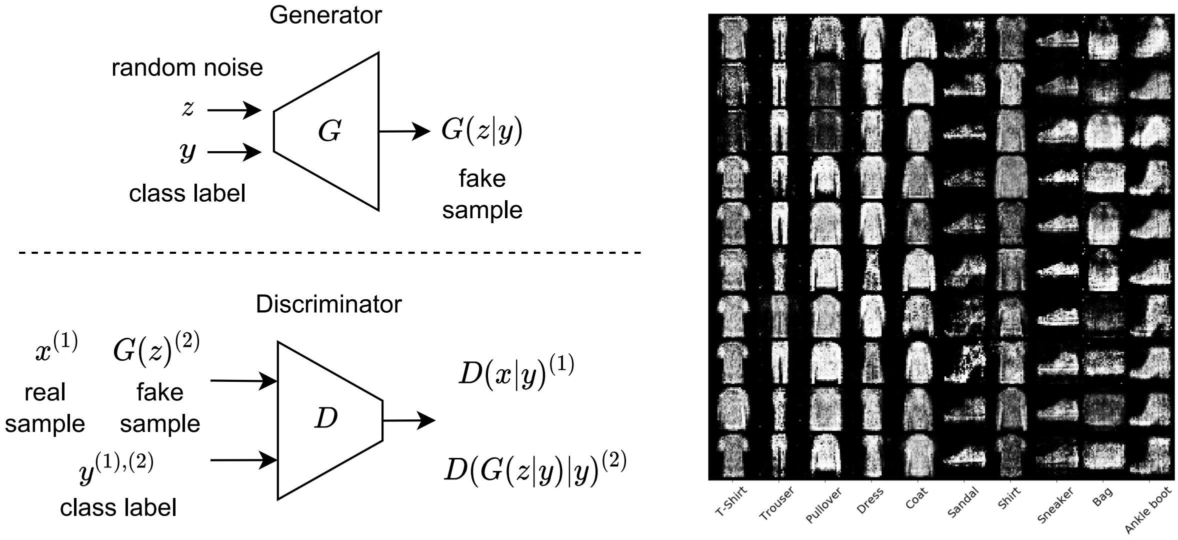
Figure 2.6: (a) Overview of the architecture of GANs [1]. A Generator G produces images given a noise vector z. Additionally, a Discriminator D tries to classify real and synthetic images as either real or fake boosting the quality of the generated images. Real (b) and fake (c) samples of Fashion-MNIST [100], generated by a GAN. The image with generated samples has been taken from [101].

generator and the discriminator can be expressed as:

$$\min_{G} \max_{D} \mathbb{E}_{x \sim p_{data}}[\log D(x)] + \mathbb{E}_{z \sim p_z}[\log(1 - D(G(z)))], \tag{2.13}$$

where D maximises the probability of the correct label to be assigned to a given data sample, while G tries to improve the quality of the generated data. Samples generated by a GAN trained on the Fashion-MNIST dataset [100], as well as real ones are presented in Figure 2.6 (b)-(c).

Since the inception of GANs [1], numerous variations have been proposed in the literature, including methods that introduce different GAN architectures [26, 102, 103, 17, 104, 2], as well as different cost functions [105, 36, 106]. These methods are proposed to enhance the generation capabilities of GANs and alleviate problems in training, such as vanishing gradients (if the discriminator is too good, it doesn't provide enough information for the generator to make progress) and mode collapse (the generator produces only certain types of images that succeed in ”fooling” the discriminator).



(a) cGAN architecture

(b) Fashion-MNIST generated images

Figure 2.7: (a) Overview of the architecture of cGANs [26]. The Generator G and discriminator are conditioned on a class label c additionally to the noise vector z. (b) Fake samples from specific classes of Fashion-MNIST [100], generated by a cGAN. The image with generated samples has been taken from `https://www.kaggle.com/arturlacerda/pytorch-conditional-gan/notebook`, which has been released under the Apache 2.0 open source license.

One characteristic of the original GAN architecture is that the generation process is random, meaning that G randomly produces data from a learnt data distribution. To achieve some control over the type or class of data to be generated, conditional GANs (cGANs) [26] were introduced. For example, we might want to produce images from a specific class of clothes of the Fashion-MNIST dataset, e.g. ”T-shirts”. To solve that cGANs proposed to introduce class labels in the inputs of both the generator and the discriminator, in addition to the random noise z or the real/generated samples respectively. A overview of the cGAN architecture along with generated samples from specific classes of Fashion-MNIST are depicted in Figure 2.7.

2.3.2 pix2pix

The pix2pix model [17] solves the problem of learning one-way translations between images of the same object depicted in two different domains (e.g. night/day, winter/summer, segmentation map/image, sketch/image, facial expression A/facial expression B, etc.). In particular, having a collection of im-

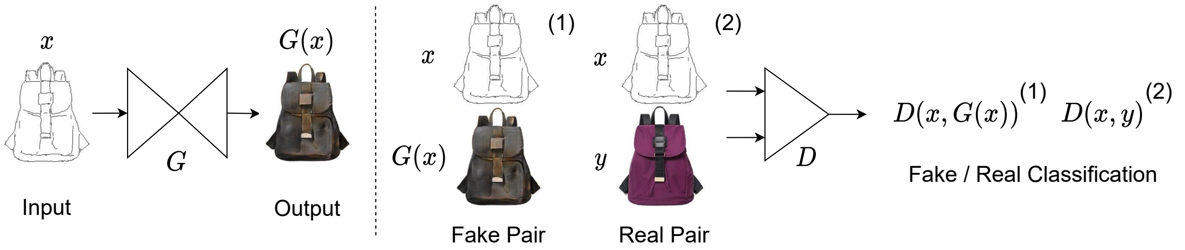


Figure 2.8: A diagram of the pix2pix model and data flow during training. Pairs of images of the same scene/object but in two different domains are used to train the Generator. The Discriminator is trained with pairs of images where the target domain image is either a real or a generated one. Images are taken from the original paper [17].

ages $(x_1, \ldots, x_N)$ of a source domain X and their corresponding instantiations $(y_1, \ldots, y_N)$ in a target domain Y, pix2pix attempts to learn a mapping between domains X and Y as $F : X \rightarrow Y$.

To achieve that pix2pix utilises the cGAN framework to train a generator network G, which is conditioned on input images x to produce the outputs as $G : \{x, z\} \rightarrow y$, where z is random noise which aids to avoid deterministic outputs. The generator G is a particular encoder-decoder architectures with skip connections, named U-net [107], which helps to pass low-level information from input images to the output ones in contrast to standard encoder-decoder networks which might be restricted by the capacity of the bottleneck. As in GANs and cGANs, the generator is trained to produce realistic images by enforcing an adversary by simultaneously training a discriminator network D which operates on images patches, coined PatchGAN [17], to distinguish between real images and fake ones produced by G. A diagram of pix2pix is presented in Figure 2.8.

To optimise pix2pix the objective of cGANs is employed along with an $L1$ distance loss between the pixels of the generated images and the ground truth ones to encourage G to produce outputs which are as close as possible to the available ground truth. The total objective can then be written as:

$$\min_G \max_D \mathcal{L}_{GAN}(G, D) + \lambda \mathcal{L}_{L1}(G), \tag{2.14}$$

where:

$$\mathcal{L}_{GAN}(G, D) = \mathbb{E}_{x,y}[\log D(x, y)] + \mathbb{E}_{x,z}[\log(1 - D(x, G(x, z))] \tag{2.15}$$

is the cGAN objective,

$$\mathcal{L}_{L1}(G) = \mathbb{E}_{x,y,z}[\|y - G(x, z)\|_1] \tag{2.16}$$

the pixel wise loss and λ a regularizer.

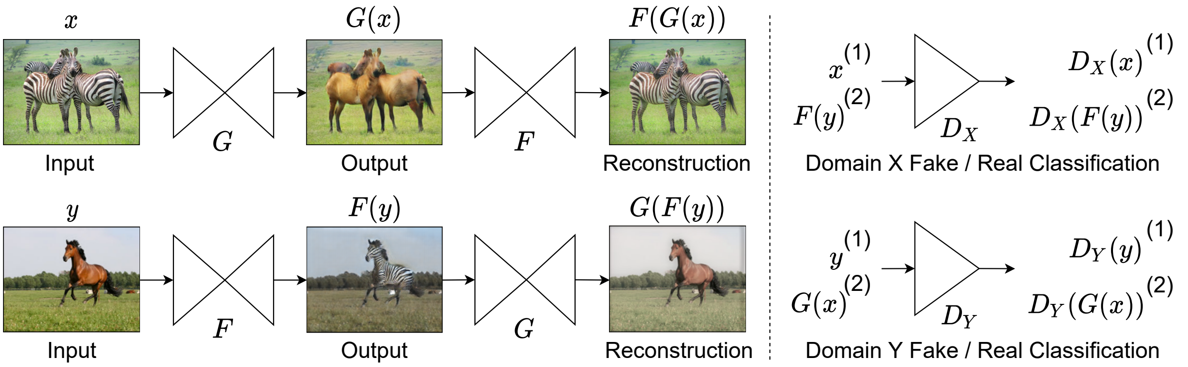


Figure 2.9: A diagram of the CycleGAN model and data flow during training. Images of two separate domains are employed during training and no pairs are required. During training, two separate mappings (Generators) and two Discriminators are trained within a cyclic loss. Images are taken from the original paper [18].

2.3.3 CycleGAN

One drawback of pix2pix is the requirement for image pairs of the same object/subject in both source and target domains, which is posed by the $L1$ pixel loss between generated and ground truth images. To overcome this, authors in [18] proposed CycleGAN, a cGAN training framework based on cyclic consistency of the image translation process. That is, exact image pairs are not necessary anymore, but arbitrary images of the two domains are enough for training.

CycleGAN is trained to learn mappings between two image domains X and Y in both directions, i.e. mappings $G : X \rightarrow Y$ and $F : Y \rightarrow X$. Mappings G and F are implemented by two separate generator networks with encoder-decoder architectures adopted from [108]. Two adversarial discriminators D_X and D_Y are also employed, where D_X aims to distinguish between images from domain X and the ones generated as $F(y), y \in Y$, while D_Y aims to distinguish between images from domain Y and the ones generated as $G(x), x \in X$. The above networks are trained by optimising the adversarial loss of [1], which for networks G and D_Y is

$$\mathcal{L}_{GAN}(G, D_Y) = \mathbb{E}_{y\sim p_{data}(y)}[\log D_Y(x, y)] + \mathbb{E}_{x\sim p_{data}(x)}[\log(1 - D_Y(x, G(x))], \tag{2.17}$$

where $x \sim p_{data}(x)$ and $y \sim p_{data}(y)$ are the data distributions, while for networks F and D_X it is similarly expressed as $\mathcal{L}_{GAN}(F, D_X)$.

Adversarial losses alone cannot guarantee that the learned functions can map inputs x to specific outputs y. That is why a reconstruction loss is required to restrict the space of possible mapping outcomes. In pix2pix, this is achieved by the $L1$ loss between ground truth and generated images. In CycleGAN, a cyclic mapping $x \rightarrow G(x) \rightarrow F(G(x)) \rightarrow x'$ is assumed to be able to bring x back to the original image. Similarly, reconstructions y' of images of domain Y should be obtained by the mapping chain $y \rightarrow F(y) \rightarrow G(F(y)) \rightarrow y'$. These observations are employed to formulate a cycle consistency loss as

$$\mathcal{L}_{cyc}(G, F) = \mathbb{E}_{x \sim p_{data}(x)}[\|x - F(G(x))\|_1 + \mathbb{E}_{y \sim p_{data}(y)}[\|y - G(F(y))\|_1]. \tag{2.18}$$

An overall diagram of CycleGAN's data flow during training is provided in Figure 2.9. Moreover, the total objective is formed by combining the GAN and the cycle consistency losses weighted by a regularising parameter λ as:

$$\min_{G,F} \max_{D_X, D_Y} \mathcal{L}_{GAN}(G, D_Y) + \mathcal{L}_{GAN}(G, D_Y) + \lambda \mathcal{L}_{cyc}(G, F). \tag{2.19}$$

2.3.4 StarGAN

According to cycleGAN, generator networks can be trained to translate images between two specific image domains without the requirement of exact supervision during training. However, if translations between more than two domains need to be learnt, these must be learnt in pairs which means separate models must be utilised and trained for each pair of image domains.

StarGAN [19] solves the previous issue by training a single generator model G to perform translations between multiple domains. This is achieved by conditioning G to the target domain label, i.e. learning a mapping $G : \{x, c\} \rightarrow y$. Additionally, the discriminator D of StarGAN is not only trained to distinguish real and fake images, but also to predict the domain of an input image either it is real or generated, which encourages the generator to produce images that match the desired domain distribution. That is, D is a CNN with two heads producing probability distributions for both image source and image domain such as $D : x \rightarrow \{D_{src}(x), D_{cls}(x)\}$. A model diagram of StarGAN is provided in Figure 2.10.

StarGAN is trained using the adversarial loss $\mathcal{L}_{adv} = \mathcal{L}_{GAN}(G, D_{src})$, similarly to the previous methods, which drives G to produce realistic images of a target domain c. Besides, classification losses are introduced to train the classification head of D with both real and fake images, which are

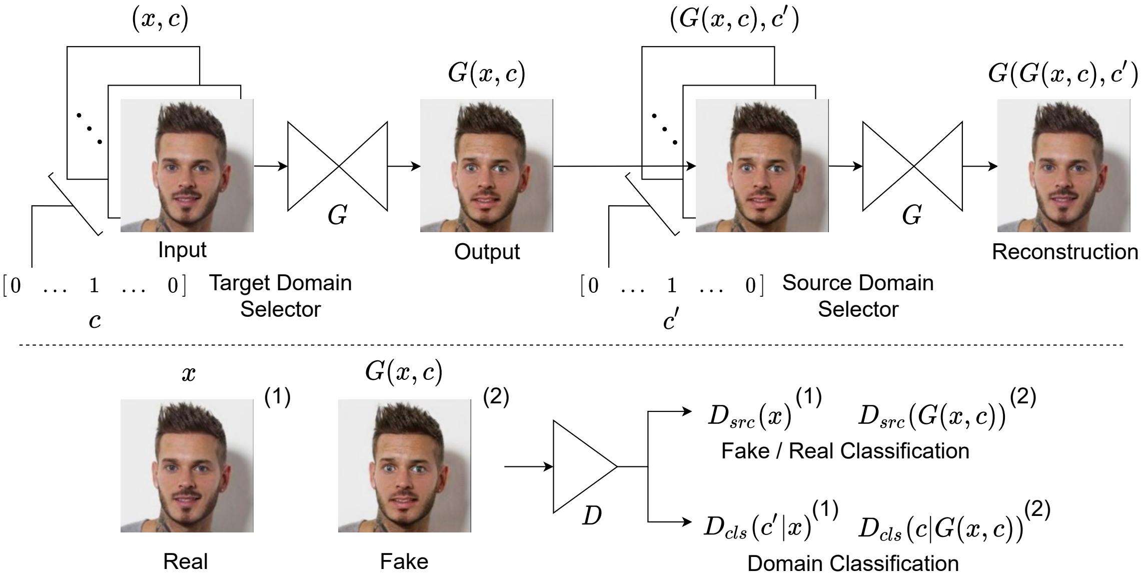


Figure 2.10: A diagram of the StarGAN model and data flow during training. In StarGAN a single Generator G is trained conditioned on a target domain class vector c to perform image translation between multiple domains. No pairs are required as supervision is obtained through cycle consistency. The Discriminator D predicts the domain class vector of its input images along with classifying them as real or fake, which enables domain specific generation for G. Images are taken from the original paper [19].

formulated as:

$$\mathcal{L}^r_{cls} = \mathbb{E}_{x,c'}[-\log D_{cls}(c'|x)] \tag{2.20}$$

$$\mathcal{L}^f_{cls} = \mathbb{E}_{x,c}[-\log D_{cls}(c|G(x, c))] \tag{2.21}$$

Lastly, a reconstruction loss is employed to ensure that the generated images maintain the content of input images changing only the domain specific information. The reconstruction loss is defined by a cyclic transformation similarly to cycleGAN, with the difference of using a single generator model and selecting the target domain by a domain label. That is the $L1$ reconstruction loss is defined as:

$$\mathcal{L}_{rec} = \mathbb{E}_{x,c,c'}[\|x - G(G(x, c), c')\|_1]. \tag{2.22}$$

The optimisation objectives for G and D are finally formulated as:

$$\min_G \mathcal{L}_{adv} + \lambda_{cls}\mathcal{L}^f_{cls} + \lambda_{rec}\mathcal{L}_{rec}, \tag{2.23}$$

$$\max_D - \mathcal{L}_{adv} + \lambda_{cls}\mathcal{L}^r_{cls}, \tag{2.24}$$

where λ_{cls} and λ_{rec} are hyper-parameters controlling the contribution of the corresponding losses in the total loss.

The StarGAN is closely related to our model SliderGAN discussed in Chapter 4 of this thesis. While StarGAN controls domain translation by discrete domain codes, SliderGAN employs continuous codes allowing for smooth translations in facial expression editing applications.

2.4 Recent Methods for Generative Editing and Implicit Modelling of Faces

Recently, significant progress has taken place in the field of generative modelling of images including images of the human face. These methods have allowed the generation and manipulation of face images while maintaining significant levels of detail and realism. Four categories in which recent techniques can be divided into, according to their 2D or 3D nature and their fundamental model structure are: (a) Image-based methods, which include generative models such as GANS that directly act on the 2D space of images, (b) 3D-aware GAN methods which leverage implicit 3D representations to model the 3D geometry and offer pose consistent generation, (c) methods that create and modify Volumetric Neural Face Avatars and (d) Diffusion-based models that handle generation as image demonising.

2.4.1 Image-Based Methods

Early face synthesis and editing methods were relying on reconstruction of the shape and texture using morphable models followed by forward rendering [109, 110, 111]. Because of the 3DMM, these methods offer continuous and accurate deformations, nevertheless fail to model the non-facial parts such as hair and the mouth interior and produce artifacts around the rendering area. To tackle these problems, hybrids between 3DMM rendering and learning-based methods were developed. Starting from 3DMM renderings, some methods have attempted to enhance their realism and add facial details [112, 3, 113, 114, 115, 116, 117, 118]. Deep Video Portraits [114] and Deferred Neural Rendering [117] are among the earliest methods that learn i2i translation networks for face image generation based on rendered dense correspondence maps and correspondence-aware feature maps respectively. In contrast, the methods in [118, 119, 120] rely on landmarks which is a more sparse face representation. Estimating motion in videos, First Order Motion Model [121] is able to transfer face animation between a source and a target video. Combining insight from the above, HeadGAN [3, 122] estimates motion between a source and a target state, warps the source image and generates realistic faces based on rendered dense [3] or sparse [122] face landmarks.

A key attribute of face image editing methods is the disentanglement of different facial characteristics and motions. In such methods disentanglement cannot only be achieved by imposing 3D priors such as the 3DMM, but also via optimizing the latent space of a network to learn decoupled representations that can be controlled by users to edit specific facial attributes. DiscoFaceGAN [112] learns a disentangled latent space in an imitative-contrastive scheme, based on 3D face priors. Other works, operate directly on pre-trained GAN models such as the StyleGAN [2], and attempt to disentangle their latent space [123, 124, 125, 116] offering parametric control over the generation process. Differently from these approaches, controlled image-based face manipulation has also been achieved via editing 2D semantic masks [126, 127, 128, 129] or sketches [130, 131]. Among these SPADE [128] is a general semantic-guided image generation method that utilizes spatially adaptive normalization and produces appealing results with high semantic alignment.

Even though the above methods achieve photorealistic results and flexibility in face editing and generation, they do not understand the 3D nature of objects and thus, they suffer from the lack of multi-view geometric consistency, i.e. the generated face images are not consistent for different head poses, as well as generation artifacts for large expression changes as expression is not completely disentangled from the head pose.

2.4.2 3D-Aware GAN Methods

To offer explicit 3D camera control in image generation, numerous methods have recently been proposed that lift image synthesis to 3D [132, 133, 134, 135, 136, 137]. EG3D [132] combines the high fidelity of StyleGAN [2] with neural-volume rendering to offer multi-view consistency and state-of-the-art 3D face generation. Inspired by EG3D, HFA-GP [133] learns personalized 3D generative priors and reconstructs reliable individual characteristics. Moreover, π-GAN [134] combines 3D shape and texture under global latent codes which are employed as conditions to a siren-based neural implicit representation. Even though these methods demonstrate image generation of impressive quality, they do not offer fine-grained face editing controlled by a user.

Focusing on the editability of 3D-aware GANs the models in[138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148] allow for controlled face image generation and editing. In particular, the works in [142, 141, 147, 146] perform image editing via semantic masks. FENeRF [142] and IDE-3D [141] employ radiance fields to model the semantic representtaions rather than only shape and texture and involve GAN inversion to map input images to a pre-trained model's latent space. NeRFFaceEditing [147] allows for disentangled editing of both shape and texture via decomposing and recomposing triplane features. SofGAN [146] employs multi-view images paired with ground-truth 3D shapes to learn the semantic volumes as semantic occupancy fields, which are used in test time for image synthesis and editing. A drawback of the above methods is that they do not produce consistent and continuous results, thus are prohibited for video editing. On the contrary [138, 140, 143, 144, 148] incorporate different 3D priors to guide synthesis and produce consistent animations. Common weak points of these approaches include that they require more elaborate losses and that the topology changes are not supported as in the case of semantically-guided methods. Lastly, Next-3D [149] overcomes the topology limitation by combing semantic 3D volumes and mesh-guidance, offering 3D-consistent animations.

2.4.3 Volumetric Neural Face Avatars

Implicit Neural Representations (INR) have recently been employed by many works to represent face appearance and geometry. Signed Distance Functions (SDF) [150], Neural Radiance Fields (NeRF) [151], discrete feature voxel grids [] and Tri-planes [] are four commonly used representations that have allowed to implicitly model animatable Neural Face Avatars. Compared to classic graphics based reconstruction and editing approaches, INRs can be used to encode non-skin facial structures such as hair and the mouth interior. Particularly NerFs combined with volumetric rendering [], due to their recent success in 3D scene reconstruction, have recently led to an extensive list of publications on learning deformable and photo-realistic head avatar models from multi-view images or videos [152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162]. Compared to 2D and 3D GAN methods for face image animation, implicit volumetric avatar models offer more strict pose and expression control but generally lack the fidelity of GANs.

Nerfies [156] was among the first NeRF based methods to generate static face avatars from unposed images or videos from a mobile phone via introducing a deformation field between the observations and a canonical space. To allow interpolations and better control of deformed parts, HyperNeRF [157] introduced a hyper-space which extended the deformation filed by an additional slicing surface offering more fine-grained sampling in canonical space. To achieve strict 3D consistency and animation, many methods have incorporated 3DMM priors or audio signals into the INRs [163, 152, 159, 160, 161, 164]. In NerFace [152] deformation field is conditioned on per-observation expression parameters recovered from a 3DMM. Although is can produce interpolated expressions and poses, it struggles to create new ones. RigNeRF [163] estimates a deformation residual over one calculated from reconstructed 3D meshes and provides full control of the pose and expressions. IMvatar [161] uses neural implicit surfaces and an analytical gradient to control head avatars via 3DMM parameters. [165] learns separate neural radiance fields for each 3DMM expression component. PointAvatar [166] represents the avatars via deformable points, while ConFies [167] employs automatic AU tracking to control the face. Lastly, CoNeRF [168] leverages manual semantic segmentations to control the face without any prior model.

The methods mentioned above construct head avatars for specific subjects meaning that they cannot generalize to new ones. As fine-grained details of faces are not always visible in short videos used to train the above models, many methods are using multi-view image datasets obtained under structured camera and lighting conditions [158, 169, 170, 171, 172]. Aiming to generalize to new subjects and construct avatars from as few as one face image, Pixel-aligned Volumetric Avatars [158] employs trains an encoder to estimate per-pixel features from input images and uses them to condition a NeRF. HeadNeRF [169] learns a generic radiance field from a combination of mutli-subject structured datasets and FFHQ, and employs per-pixel features and 2D neural rendering to generate images. MorF [171] is a similar model trained on a structured multi-subject dataset and generates images from the predicted density, albedo and specular information using volume rendering. Mofanerf [172] additionally employs adversarial training to achieve more detailed results. All the above methods can be fitted on target images and perform animation. Lastly, [170] proposed an approach to personalize a generic prior volumetric avatar model to new identities using only a short phone capture.

2.4.4 Diffusion methods

Diffusion models such as the Diffusion Probabilistic Models (DMs) [173, 174, 175] are generative models which first map images to a latent variable by gradually adding noise using a Markov chain, and then gradually denoising it to obtain the generated result via a learned denoising process. Diffusion models are trained with score-matching objectives [176] at the various noise levels of the denoising process, which are less complex than the GAN losses leading to more stable training than GANs. Diffusion models have recently demonstrated high quality results in various image generation tasks [174, 177, 175, 178] including text-to-image generation [179, 180], super resolution [181] image inpainting [182], video generation [183, 184] and image restoration [185]. What is more, the latest works in diffusion models have shown superior performance than GANs in multiple image generation tasks [186, 187, 188, 189, 190].

A specific field which has seen tremendous advancements with the rise of diffusion models is test guided image generation or text-to-image generation [191, 187, 192, 179, 193, 180]. StableDiffusion [190], Imagen [194], and DALL-E 2 [189], DiffusionCLIP [193], dreambooth [180] and Imagic [179] are state-of-the-art methods for general image synthesis which allow the customization of the style and contents of the synthesized results, based on text guidance. However, regardless of the power of text-to-image diffusion models, they are not ideal for face image synthesis and editing as language guidance does not provide as accurate control as other encodings, such as semantic image masks and 3D prior embeddings. For example, face editing requires to edit attributes such as head pose, expression or skin tone while maintaining all other attributes unchanged including the identity of the subject.

The above models are not desigend to handle such cases and thus, attempts to provide methods tailored to the problem have arose [195, 196, 197, 198]. FADM [195] is a diffusion model with attribute guidance for face animation. To address the difficulties in controlling diffusion models, an Attribute-Guided Conditioning Network is proposed to combine coarse animation results with outputs from a 3D face reconstruction network and condition the diffusion process. FADM produces consistent animations and outperforms 2D GAN methods in image generation quality [121, 199]. In [198] the authors propose Collaborative Diffusion to achieve control of existing, pre-trained diffusion models based on both text and 2D semantic masks, without the need to fine-tune the base models. DiffTalk [197], addresses the problem of talking-head synthesis, using Latent Diffusion models which they condition on reference images and driving landmarks, while successfully preserving the face attributes of the reference subject. Lastly, Dual Condition Face Generator (DCFace) [196] proposes a method for controllable face image synthesis, aiming to improve the task of face verification. DCFace produces consistent face images with different styles of the same subject, using a patch-wise style extractor and enforcing an ID loss at various time steps

2.5 Conclusions

In this Chapter, we presented technical details on methods and datasets that we have employed in the works of the next chapters, as well as discussed recent advancements in generative image modelling. Moreover, we discussed works which have motivated and challenged the development of our methodologies. In particular, we started by reviewing developments in 3D reconstruction of faces from images, providing details on employed methods for obtaining useful representations for training our algorithms. We continued by reviewing CA methods which are related to ours, highlighting the importance of developing robust alternatives to handle corrupted data. Then, we reviewed fundamental works in generative modeling with an emphasis on image generation with GANs and image-to-image translation methods. Lastly, we presented an overview of recent developments on generative image processing which have revolutionised the field. In the next Chapters, we present our works which attempt to mitigate the above problems.

CHAPTER 3

Recovering Joint and Individual Components in Facial Data

Contents

3.1 Introduction

Facial images convey rich information, which can be perceived as a superposition of components associated with attributes, such as facial identity, expression, age etc. For instance, a set of images depicting expressive faces consists of components that are shared across all images (i.e., *joint* components) and imparts to the depicted object the properties of human faces. Besides joint components, an expressive face consists of *individual* components that are related to different expressions. Such individual components can be expression-specific deformation of face, i.e., deformations around lips and eyes in case of smiles. Similarly, a set of images depicting faces in different ages can be seen as a superposition of joint components that are invariant to the age and age-specific components that are individual to each age group (e.g., wrinkles). Consequently, being able to extract such joint and individual components from facial images is crucial for applications such as facial expression synthesis and face age progression [200, 201, 202, 203, 204, 205], among other visual data analysis tasks.

Extracting the joint components among data has created a wealth of research in statistics, signal processing, and computer vision. Two mathematically similar but conceptually different models underlie the bulk of the methodologies. In particular, the Canonical Correlation Analysis (CCA) [93] and its variants e.g.,[206, 207] have been proposed for extracting linear correlated components among two or more sets of variables. Similarly, inter-battery factor analysis [94] and its extensions e.g., [208] determines the common factors among two sets of variables. The main limitation of the aforementioned methods is that they only recover the most correlated linear subspace of the data, ignoring the individual components among the different views or datasets.

The above mentioned limitation is alleviated by recent methods such as the Joint and Individual Variation Explained (JIVE) [14], the Common Orthogonal Basis Extraction (COBE) [15], and the Robust Correlated and Individual Component Analysis (RCICA) [16], which are briefly described in Section 2.2. Besides the rich structure in facial visual data, images are subject to various types of errors, distortions, and noise. Common dense distortions such as ambient noise or quantisation noise are of small magnitude and it is natural to assume that they follow a Gaussian distribution of small variance. Methods such as the CCA and its variants, the JIVE, and the COBE are stable in the presence of Gaussian noise.

Apart from these small but dense noises, there are gross errors that are sparsely supported but of large or even unbounded magnitude, such as the salt-and-pepper noise in imaging devices, occlusions in facial images, registration errors, or errors due incorrect localisation and tracking. These errors rarely follow a Gaussian distribution and due to their sparse nature (i.e.,the number of errors is bounded below some constant) are collectively referred to as sparse gross errors or noise. Except for the most recent RCICA, the COBE and JIVE rely on least squares error minimisation and thus they are prone to gross errors and outliers [95]. That is, the estimated components can be arbitrarily away from the true ones. Hence, the problem of joint and individual components recovery is rather challenging when dealing with facial images and in general visual data captured under unconstrained (i.e., "in-the-wild") conditions.

In this work, we investigate the problem of recovering the joint and individual components from facial (and in general visual) data consisting of an arbitrary number of views, captured "in-the-wild". Such data are therefore contaminated by sparse, gross, non-Gaussian noise and possibly contain missing values. To this end, we propose robust alternatives to the JIVE (coined collectively as Robust JIVE- RJIVE), where the components are estimated by employing the $L1$-norm. The $L1$-norm is suitable for robust estimation in the presence of sparse gross errors [95]. The contributions of this work are summarised as follows:

- We propose a novel, general framework, the RJIVE in Section 3.2.1, for the robust recovering of joint and individual components from multi-view data in the presence of sparse gross errors and possibly missing values. The proposed RJIVE decomposes the data into three terms: a low-rank matrix that captures the joint variation across views, low-rank matrices accounting for structured variation individual to each view, and a sparse matrix collecting the sparse gross errors.

- In particular, the RJIVE consists of 4 different models, namely, $L1$-RJIVE, NN-$L1$-RJIVE, SRJIVE, and RJIVE-M. In the $L1$-RJIVE, the rank of both joint and individual components are user-defined, while in the NN-$L1$-RJIVE the rank of each one of the individual components is automatically estimated via nuclear norm minimisation. As opposed to the previous two models, the SRJIVE directly extracts the orthonormal bases of joint and individual components and improves their scalability. Finally, the RJIVE-M extends the SRJIVE in order to handle missing values.

- Based on the recovered joint and individual components from training data, two suitable optimisation problems that extracts the corresponding modes of variation (i.e., joint and individual components) of unseen test samples, are proposed in Section 3.2.3.

- To tackle the proposed optimisation problems, algorithms based on the Alternating-Directions Method of Multipliers (ADMM) [28] are developed in Sections 3.2.1 and 3.2.3.

- We demonstrate the applicability of the proposed methods in three challenging computer vision tasks, namely facial expression synthesis, face age progression in 2D images and 3D data captured "in-the-wild". Experimental results corroborate the effectiveness of the proposed approach in Section 3.3.

- Furthermore, a new challenging data-set of 19.000 images captured "in-the-wild" with annotations in terms of age, is introduced in Section 3.3.3 for age-invariant face verification.

Notation: Throughout this Chapter, scalars are denoted by lower-case letters, vectors (matrices) are denoted by lower-case (upper-case) boldface letters i.e., $\mathbf{x}$, $(\mathbf{X})$. $\mathbf{I}$ denotes the identity matrix. The j-th column of $\mathbf{X}$ is denoted by $\mathbf{x}_j$.

Several norms and metrics will be used.The $L1$ and the $L2$ norms of $\mathbf{x}$ are defined as $\|\mathbf{x}\|_1 = \sum_i |x_i|$ and $\|\mathbf{x}\|_2 = \sqrt{\sum_i x_i^2}$, respectively. $|\cdot|$ denotes the absolute value operator. The matrix $L1$ norm is defined as $\|\mathbf{X}\|_1 = \sum_i \sum_j |x_{ij}|$, and the Frobenius norm is defined as $\|\mathbf{X}\|_F = \sqrt{\sum_i \sum_j x_{ij}^2}$, and the nuclear norm of $\mathbf{X}$ (i.e., the sum of singular values of a matrix) is denoted by $\|\mathbf{X}\|_*$. The vector (matrix) $L0$ -(quasi) norm returns the total number of non-zero elements in a vector (matrix). The rank function is denoted by rank$(\cdot)$.

The minimisation of both the rank function and the $L0$-norm are NP-hard [209, 210] problems. Consequently, the rank function and the $L0$-norm are typically replaced by their convex surrogates [97, 98].

Operators: The solution of the several problems appeared in this Chapter rely on different (proximal) operators which are defined next. Let, for any matrix $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ be the Singular Value Decomposition.

- Shrinkage operator [96]: $\mathcal{S}_\tau[\sigma] = \text{sgn}(\sigma)\max(|\sigma| - \tau, 0)$.

- Singular Value Thresholding (SVT) operator [211]: $\mathcal{D}_\tau = \mathbf{U}\mathcal{S}_\tau\mathbf{V}^T$.

- Rank-r SVD operator:
  $\mathcal{Q}_r\left[\mathbf{X}\right] = \left[\mathbf{U}(:,1:r)\mathbf{\Sigma}(1:r,1:r)\mathbf{V}(:,1:r)^T\right]$.

- Procrustes operator: $\mathcal{P}\left[\mathbf{D}\right] = \mathbf{PR}^T$ (given the rank-r SVD of a matrix $\mathbf{D} = \mathbf{GPR}^T$).

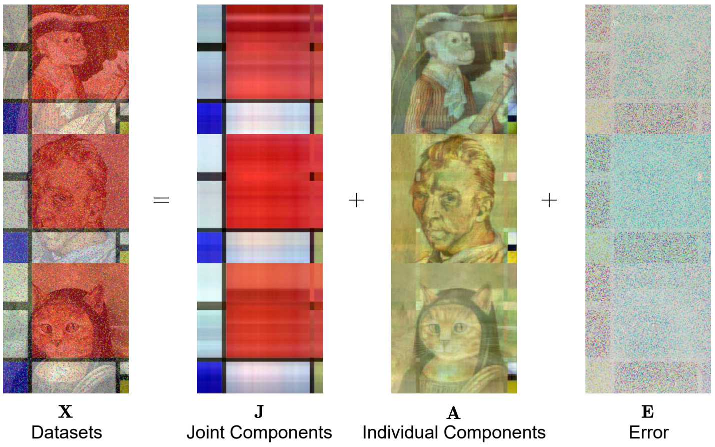


Figure 3.1: A visual representation of the proposed RJIVE decomposition. Given a arbitrary number of data-sets or views captured under totally unconstrained conditions, the proposed method extracts components that capture the joint structure between the data-sets $(\mathbf{J})$, the individual structure to each data-set $(\mathbf{A})$, and a sparse matrix collecting the sparse non-Gaussian errors $(\mathbf{E})$.

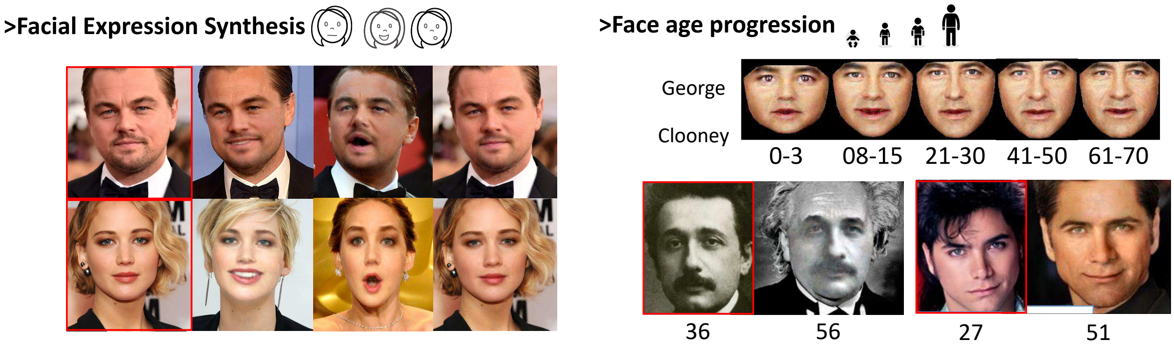


Figure 3.2: A visual representation of the applications considered in this Chapter i.e., facial expression transfer and face age progression. Images highlighted in red boxes are given as input to the RJIVE.

3.2 Methodology

Here we formulate the various versions of RJIVE, as well as methods for their optimisation. Additionally, we construct problems for image reconstruction based on recovered components and present their solutions.

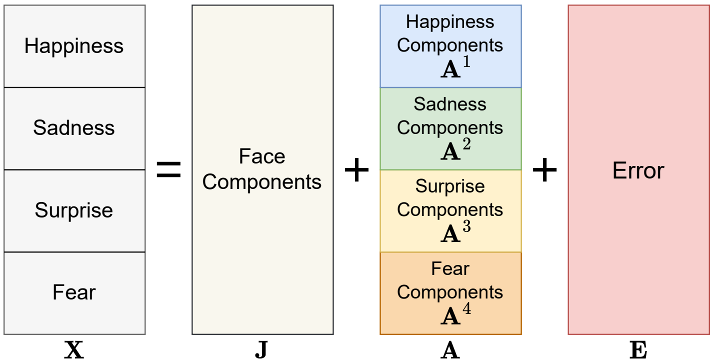

Happiness
Sadness
Surprise
Fear
X
=
Face Components
J
+
Happiness Components $\mathbf{A}^1$
Sadness Components $\mathbf{A}^2$
Surprise Components $\mathbf{A}^3$
Fear Components $\mathbf{A}^4$
A
+
Error
E

Figure 3.3: A visual representation of the RJIVE decomposition in the example of analysing data annotated with respect to four facial expressions. As can be seen, the ordered dataset $\mathbf{X}$ is analysed into a matrix of joint components $\mathbf{J}$, matrices $\mathbf{A}_1$, $\mathbf{A}_2$, $\mathbf{A}_3$ and $\mathbf{A}_4$ each corresponding to a specific annotated expression and matrix $\mathbf{E}$ which collects sparse, non-gaussian noise allowing for cleaner data matrices $\mathbf{J}$ and $\mathbf{A}_i, i = \{1, 2, 3, 4\}$.

3.2.1 Robust JIVE

Consider data consisting of M views $\{\mathbf{X}^{(i)} \in \mathbb{R}^{d^{(i)} \times J}\}_{i=1}^{M}$, with $\mathbf{x}_j^{(i)} \in \mathbb{R}^{d^{(i)}}$, $j = 1, \dots, J$ being a vectorised (visual) data sample, possibly contaminated by gross, sparse errors. The goal of the RJIVE is to robustly recover the joint components which are shared across all views as well as the components which are deemed individual for each view. That is:

$$\mathbf{X} = \mathbf{J} + \left[\mathbf{A}^{(1)^T}, \cdots, \mathbf{A}^{(M)^T}\right]^T + \mathbf{E}, \tag{3.1}$$

where $\mathbf{X} = \left[\mathbf{X}^{(1)^T}, \cdots, \mathbf{X}^{(M)^T}\right]^T \in \mathbb{R}^{q \times J}$, $\mathbf{J} = \left[\mathbf{J}^{(1)^T}, \cdots, \mathbf{J}^{(M)^T}\right]^T \in \mathbb{R}^{q \times J}$, $\{\mathbf{A}^{(i)} \in \mathbb{R}^{d^{(i)} \times J}\}_{i=1}^{M}$, $q = d^{(1)} + \cdots + d^{(M)}$, are low-rank matrices capturing the joint and individual variations, respectively and $\mathbf{E} \in \mathbb{R}^{q \times J}$ denotes the error matrix accounting for the gross, but sparse, non-Gaussian noise. Figure 3.3 presents the structure of the decomposition that RJIVE aims to achieve in the particular example of analysing facial expression data. In order to ensure the identifiability of (3.1), the joint and common components should be mutual incoherent, i.e., $\{\mathbf{J}\mathbf{A}^{(i)^T} = \mathbf{0}\}_{i=1}^{M}$. Assuming that the number of errors is bounded below some constant, the number of errors in the estimated components is similarly bounded and hence a natural estimator accounting for the sparsity of the error matrix $\mathbf{E}$, is to minimise the number of the nonzero entries of $\mathbf{E}$ measured by the $L0$-quasi norm [96]. However as in case of the RCICA, to make the problem computationally tractable the $L0$-norm is replaced by its convex surrogate, namely the $L1$-norm. Thus, the joint and individual components as well as the sparse error are recovered by solving the following constrained non-linear optimisation problem:

$$\begin{aligned} &\min_{\mathbf{J},\{\mathbf{A}^{(i)}\}_{i=1}^{M}} \quad \left\| \mathbf{X} - \mathbf{J} - \left[\mathbf{A}^{(1)^T}, \cdots, \mathbf{A}^{(M)^T} \right]^T \right\|_1 . \\ &\text{s.t.} \quad \operatorname{rank}(\mathbf{J}) = r, \{\operatorname{rank}(\mathbf{A}^{(i)}) = r^{(i)}, \mathbf{J}\mathbf{A}^{(i)^T} = \mathbf{0}\}_{i=1}^{M} \end{aligned} \tag{3.2}$$

Clearly, (3.2) is a robust extension to JIVE [14] and requires an estimation for the rank of both joint and individual components. However, in practice those $(M+1)$ values are unknown and difficult to estimate since an extensive tunning procedure is required. To alleviate this issue, we propose a variant of (3.2) which is able to determine the optimal ranks of individual components directly. By assuming that the actual ranks of individual components are upper bounded i.e., $\{\operatorname{rank}(\mathbf{A}^{(i)}) \leq K^{(i)}\}_{i=1}^{M}$, problem (3.2) is relaxed to the following one:

$$\begin{aligned} &\min_{\mathbf{J},\{\mathbf{A}^{(i)}\}_{i=1}^{M}} \lambda \left\| \mathbf{X} - \mathbf{J} - \left[\mathbf{A}^{(1)^T}, \cdots, \mathbf{A}^{(M)^T} \right]^T \right\|_1 + \sum_{i=1}^{M} \left\| \mathbf{A}^{(i)} \right\|_* , \\ &\text{s.t.} \quad \operatorname{rank}(\mathbf{J}) = r, \{\mathbf{J}\mathbf{A}^{(i)^T} = \mathbf{0}\}_{i=1}^{M} \end{aligned} \tag{3.3}$$

where the rank function is replaced by its convex envelope, namely the nuclear norm and $\lambda > 0$ is a regularizer.

3.2.2 Optimisation Algorithms

In this section, algorithms for solving (3.2) and (3.3) are developed.

To solve (3.2), the Alternating-Direction Method of Multipliers (ADMM) [28] is employed. To this end, problem (3.2) is reformulated to the following separable one:

$$\begin{aligned} &\min_{\mathbf{J},\{\mathbf{A}^{(1)}\}_{i=1}^{M},\mathbf{E}} \quad \|\mathbf{E}\|_1 , \\ &\text{s.t.} \quad \mathbf{X} = \mathbf{J} + \left[\mathbf{A}^{(1)^T}, \cdots, \mathbf{A}^{(M)^T} \right]^T + \mathbf{E}, \\ &\qquad \operatorname{rank}(\mathbf{J}) = r, \{\operatorname{rank}(\mathbf{A}^{(i)}) = r^{(i)}, \mathbf{J}\mathbf{A}^{(i)^T} = \mathbf{0}\}_{i=1}^{M}, \end{aligned} \tag{3.4}$$

where $\mathbf{E}$ is an auxiliary variable. To solve (3.4), the corresponding augmented Lagrangian function is given by:

$$\begin{aligned} &\mathcal{L}(\mathbf{J}, \{\mathbf{A}^{(i)}\}_{i=1}^{M}, \mathbf{E}, \mathbf{L}) = \|\mathbf{E}\|_1 - \frac{1}{2\mu} \|\mathbf{L}\|_F^2 \\ &+ \frac{\mu}{2} \left\| \mathbf{X} - \mathbf{J} - \left[\mathbf{A}^{(1)^T}, \cdots, \mathbf{A}^{(M)^T} \right]^T - \mathbf{E} + \frac{\mathbf{L}}{\mu} \right\|_F^2 , \end{aligned} \tag{3.5}$$

Algorithm 1: ADMM solver for (3.4) ($L1$-RJIVE).

```
Input     : Data $\{\mathbf{X}^{(i)} \in \mathbb{R}^{d^{(i)} \times J}\}_{i=1}^{M}$. Rank of joint component $r$. Ranks of individual
            components $\{r^{(i)}\}_{i=1}^{M}$. Parameter $\rho$.
Output    : Joint component $\mathbf{J}$, individual components $\{\mathbf{A}^{(i)}\}_{i=1}^{M}$
Initialise: Set $\mathbf{J}_0$, $\{\mathbf{A}_0^{(i)}\}_{i=1}^{M}$, $\mathbf{E}_0$, $\mathbf{L}_0$ to zero matrices, $t = 0$, $\mu_0 > 0$.
$\mathbf{X} = \left[\mathbf{X}^{(1)^T}, \cdots, \mathbf{X}^{(M)^T}\right]^T$;
while not converged do
    $\mathbf{M} = \mathbf{X} - \left[\mathbf{A}_t^{(1)^T}, \cdots, \mathbf{A}_t^{(M)^T}\right]^T - \mathbf{E}_t + \mu_t^{-1}\mathbf{L}_t$;
    $\mathbf{J}_{t+1} = \mathcal{Q}_r[\mathbf{M}]$, $[\mathbf{U}, \mathbf{\Sigma}, \mathbf{V}] = \text{svd}(\mathbf{M})$;
    $\mathbf{P} = \mathbf{I} - \mathbf{V}(:, 1:r)\mathbf{V}(:, 1:r)^T$;
    for $i = 1 : M$ do
        $\mathbf{A}_{t+1}^{(i)} = \mathcal{Q}_{r^{(i)}}\left[\left(\mathbf{X}^{(i)} - \mathbf{J}_{t+1}^{(i)} - \mathbf{E}_t^{(i)} + \mu_t^{-1}\mathbf{L}_t^{(i)}\right)\mathbf{P}\right]$
    $\mathbf{E} = \mathcal{S}_{\frac{1}{\mu_t}}\left[\mathbf{X} - \mathbf{J}_{t+1} - \left[\mathbf{A}_{t+1}^{(1)^T}, \cdots, \mathbf{A}_{t+1}^{(M)^T}\right]^T - \mu_t^{-1}\mathbf{L}\right]$;
    $\mathbf{L}_{t+1} = \mathbf{L}_t + \mu_t\left(\mathbf{X} - \mathbf{J}_{t+1} - \left[\mathbf{A}_{t+1}^{(1)^T}, \cdots, \mathbf{A}_{t+1}^{(M)^T}\right]^T - \mathbf{E}_{t+1}\right)$;
    $\mu_{t+1} = \min(\rho \cdot \mu_t, 10^7)$;
    $t = t + 1$;
```

where $\mathbf{L}$ is the Lagrange multipliers matrix related to the equality constraint in (3.4), and μ is a positive parameter. Then, by employing the ADMM, (3.5) is minimised with respect to each variable in an alternating fashion and finally the Lagrange multipliers $\mathbf{L}$ are updated. The ADMM solver of (3.4) is outlined in Algorithm 1 which terminates when $\left\|\mathbf{X} - \mathbf{J}_{t+1} - \left[\mathbf{A}_{t+1}^{(1)^T}, \cdots, \mathbf{A}_{t+1}^{(M)^T}\right]^T - \mathbf{E}_{t+1}\right\|_F^2 / \|\mathbf{X}\|_F^2$ is less than a predefined threshold ϵ or the number of iterations reach a maximum value.

To solve problem (3.3) via ADMM, we firstly reformulate it as:

$$\begin{aligned} &\min_{\mathbf{J}, \{\mathbf{A}^{(i)}, \mathbf{R}^{(i)}\}_{i=1}^{M}, \mathbf{E}} \sum_{i=1}^{M} \left\|\mathbf{R}^{(i)}\right\|_* + \lambda \|\mathbf{E}\|_1, \\ &\text{s.t. } \mathbf{X} = \mathbf{J} + \left[\mathbf{A}^{(1)^T}, \cdots, \mathbf{A}^{(M)^T}\right]^T + \mathbf{E}, \\ &\quad \text{rank}(\mathbf{J}) = r, \{\mathbf{R}^{(i)} = \mathbf{A}^{(i)}, \mathbf{J}\mathbf{A}^{(i)^T} = \mathbf{0}\}_{i=1}^{M} \end{aligned} \tag{3.6}$$

where $\{\mathbf{R}^{(i)} \in \mathbb{R}^{d^{(i)} \times J}\}_{i=1}^{M}$, $\{\mathbf{R}^{(i)} = \mathbf{A}^{(i)}\}_{i=1}^{M}$ are auxiliary variables and the corresponding con-

Algorithm 2: ADMM solver of (3.6) (NN-$L1$-RJIVE).

```
Input     : Data $\{\mathbf{X}^{(i)} \in \mathbb{R}^{d^{(i)} \times J}\}_{i=1}^{M}$. Rank of joint component $r$. Ranks of individual
            components $\{r^{(i)}\}_{i=1}^{M}$. Parameter $\rho$.
Output    : Joint component $\mathbf{J}$, individual components $\{\mathbf{A}^{(i)}\}_{i=1}^{M}$
Initialise: Set $\mathbf{J}_0$, $\{\mathbf{A}_0^{(i)}, \mathbf{R}_0^{(i)}, \mathbf{Y}_0^{(i)}\}_{i=1}^{M}$, $\mathbf{E}_0$, $\mathbf{F}_0$ to zero matrices, $t = 0$, $\mu_0 > 0$.
$\mathbf{X} = \left[\mathbf{X}^{(1)^T}, \cdots, \mathbf{X}^{(M)^T}\right]^T$;
while not converged do
    $\mathbf{J}_{t+1} = \mathcal{Q}_{\mathbf{r}}\left[\mathbf{X} - \left[\mathbf{A}_t^{(1)^T}, \cdots, \mathbf{A}_t^{(M)^T}\right]^T - \mathbf{E}_t + \frac{\mathbf{F}_t}{\mu_t}\right]$;
    for $i = 1 : M$ do
        $\mathbf{A}_{t+1}^{(i)} = \frac{\left(\mathbf{X}^{(i)} - \mathbf{J}_{t+1}^{(i)} - \mathbf{E}_t^{(i)} + \frac{\mathbf{F}^{(i)}}{\mu_t} + \mathbf{R}_t^{(i)} + \frac{\mathbf{Y}_t^{(i)}}{\mu_t}\right)\mathbf{P}}{2}$;
        $\mathbf{R}_{t+1}^{(i)} = \mathcal{D}_{1/\mu_t}\left[\mathbf{A}_{t+1}^{(i)} - \frac{\mathbf{Y}_t^{(i)}}{\mu_t}\right]$;
        $\mathbf{Y}_{t+1}^{(i)} = \mathbf{Y}_t^{(i)} + \mu_t(\mathbf{R}_{t+1}^{(i)} - \mathbf{A}_{t+1}^{(i)})$;
    $\mathbf{E}_{t+1} = \mathcal{S}_{\frac{\lambda}{\mu_t}}\left[\mathbf{X} - \mathbf{J}_{t+1} - \left[\mathbf{A}_{t+1}^{(1)^T}, \cdots, \mathbf{A}_{t+1}^{(M)^T}\right]^T + \frac{\mathbf{F}_t}{\mu_t}\right]$;
    $\mathbf{F}_{t+1} = \mathbf{F}_t + \mu_t(-\mathbf{J}_{t+1} - \left[\mathbf{A}_{t+1}^{(1)^T}, \cdots, \mathbf{A}_{t+1}^{(M)^T}\right]^T - \mathbf{E}_{t+1} + \mathbf{X})$;
    $\mu_{t+1} = \min(\rho\mu_t, 10^7)$;
    $t = t + 1$;
```

straints, respectively. The augmented Lagrangian function of (3.6) is then formulated as:

$$
\begin{aligned}
\mathcal{L}(\{\mathbf{J}, \{\mathbf{A}^{(i)}, \mathbf{R}^{(i)}, \mathbf{Y}^{(i)}\}_{i=1}^{M}, \mathbf{E}, \mathbf{F}\}) = {} & \sum_{i=1}^{N} \left\|\mathbf{R}^{(i)}\right\|_* + \lambda \left\|\mathbf{E}\right\|_1 \\
& + \frac{\mu}{2}\left\|\mathbf{X} - \mathbf{J} - \left[\mathbf{A}^{(1)^T}, \cdots, \mathbf{A}^{(M)^T}\right]^T - \mathbf{E} + \frac{\mathbf{F}}{\mu}\right\|_F^2 - \frac{1}{2\mu}\left\|\mathbf{F}\right\|_F^2 \\
& + \sum_{i=1}^{M}\left(\frac{\mu}{2}\left\|\mathbf{R}^{(i)} - \mathbf{A}^{(i)} + \frac{\mathbf{Y}^{(i)}}{\mu}\right\|_F^2 - \frac{1}{2\mu}\left\|\mathbf{Y}^{(i)}\right\|_F^2\right),
\end{aligned}
\tag{3.7}
$$

where $\mathbf{F}, \{\mathbf{Y}^{(i)}\}_{i=1}^{M}$ are the Lagrange multipliers related to the equality constraints and μ is a positive parameter. Then, by employing the ADMM algorithm, (3.7) is minimised with respect to each variable $\{\mathbf{J}, \{\mathbf{A}^{(i)}, \mathbf{R}^{(i)}, \mathbf{Y}^{(i)}\}_{i=1}^{M}, \mathbf{E}, \mathbf{F}\}$ in an alternating fashion and finally the Lagrange multipliers $\{\mathbf{F}, \{\mathbf{Y}^{(i)}\}_{i=1}^{M}\}$ are updated. The ADMM solver of (3.6) is wrapped up in Algorithm 2. The convergence criterion employed here is similar to Algorithm 1.

3.2.3 RJIVE-Based Reconstruction

Having recovered the individual and common components of the M views or different data-sets during training, we can exploit them them in order to extract the joint and individual modes of variations of a test sample. For instance, the components recovered by applying the RJIVE on a set of facial images of M different expressions can be utilised in order to reconstruct M expressive images $\{\mathbf{y}^{(i)}\}_{i=1}^{M}$ of an input face $\mathbf{t}$. The key motivation here, is that the expression-related patterns of the image $\mathbf{t}$ in the expression (i) lie in a linear subspace spanned by $\mathbf{D}^{(i)} \in \mathbb{R}^{d^{(i)} \times W_{\mathbf{A}}^{(i)}}$, where $\mathbf{D}^{(i)}$ has been obtained by applying the SVD onto extracted $\mathbf{A}^{(i)}$ components. Thus, the expression-related (individual) part of the test image $\mathbf{t}$ in expression (i) can be represented as a linear combination of the orthonormal bases $\mathbf{D}^{(i)}$ i.e., $\mathbf{y}_{\text{individual}}^{(i)} \approx \mathbf{D}^{(i)}\mathbf{c}^{(2)}$ with $\mathbf{c}^{(2)} \in \mathbb{R}^{W_{\mathbf{A}}^{(i)} \times 1}$ being a sparse coefficient vector. Similarly, the joint part $\mathbf{y}_{\text{joint}}^{(i)}$ is expressed as a linear combination of the orthonormal bases $\mathbf{B}^{(i)} \in \mathbb{R}^{d^{(i)} \times W_{\mathbf{J}}^{(i)}}$ extracted from the corresponding joint component $\mathbf{J}^{(i)}$ i.e., $\mathbf{y}_{\text{joint}}^{(i)} \approx \mathbf{B}^{(i)}\mathbf{c}^{(1)}$, $\mathbf{c}^{(1)} \in \mathbb{R}^{W_{\mathbf{J}}^{(i)} \times 1}$. Thus, the expressive image $\mathbf{y}^{(i)}$ of the unseen input face $\mathbf{t}$ is reconstructed by solving the following constrained optimisation problem:

$$
\begin{aligned}
&\min_{\{\mathbf{c}^{(n)}, \mathbf{v}^{(n)}\}_{n=1}^{2}, \mathbf{y} \geq \mathbf{0}} \sum_{n=1}^{2} \left\| \mathbf{v}^{(n)} \right\|_1 + \lambda \left\| \mathbf{e} \right\|_1, \\
&\text{s.t. } \{\mathbf{v}^{(n)} = \mathbf{c}^{(n)}\}_{n=1}^{2} \\
&\qquad \mathbf{t} = \mathbf{B}^{(i)}\mathbf{c}^{(1)} + \mathbf{D}^{(i)}\mathbf{c}^{(2)} + \mathbf{e},\ \mathbf{y} = \mathbf{B}^{(i)}\mathbf{c}^{(1)} + \mathbf{D}^{(i)}\mathbf{c}^{(2)}
\end{aligned}
\tag{3.8}
$$

where λ is a positive parameter that balances the norms, $\mathbf{v}^{(1)}$, $\mathbf{v}^{(2)}$ are auxiliary variables which are employed in order to make the problem separable, $\mathbf{y}$ corresponds to the non-negative clean reconstruction, and $\mathbf{e}$ is an error term accounting for the gross, non-Gaussian sparse noise. Equation (3.8) resembles the dense error correction model proposed in [212], which is suitable for guaranteed recovery of sparse representations from high-dimensional measurements, such as images of high resolution (e.g., 22000 pixels in this method) in the presence of noise. The augmented Lagrangian function of problem (3.8) is given by:

$$\begin{aligned}\mathcal{L}\Big(\{\mathbf{v}^{(n)}\mathbf{c}^{(n)}\}_{n=1}^{2}, \mathbf{y}, \mathbf{e}, \{\mathbf{h}^{(n)}\}_{n=1}^{4}\Big) &= \sum_{n=1}^{2}\left\|\mathbf{v}^{(n)}\right\|_1 + \lambda\left\|\mathbf{e}\right\|_1 \\ &- \frac{1}{2\mu}\sum_{n=1}^{4}\left\|\mathbf{h}^{(n)}\right\|_2^2 + \frac{\mu}{2}\Big(\sum_{n=1}^{2}\left\|\mathbf{v}^{(n)} - \mathbf{c}^{(n)} + \frac{\mathbf{h}^{(n)}}{\mu}\right\|_2^2 \\ &+ \left\|\mathbf{t} - \mathbf{B}^{(i)}\mathbf{c}^{(1)} - \mathbf{D}^{(i)}\mathbf{c}^{(2)} - \mathbf{e} + \frac{\mathbf{h}^{(3)}}{\mu}\right\|_2^2 \\ &+ \left\|\mathbf{y} - \mathbf{B}^{(i)}\mathbf{c}^{(1)} - \mathbf{D}^{(i)}\mathbf{c}^{(2)} + \frac{\mathbf{h}^{(4)}}{\mu}\right\|_2^2\Big),\end{aligned} \tag{3.9}$$

By employing the ADMM, (3.8) is minimised with respect to each variable $\{\{\mathbf{v}^{(n)}\mathbf{c}^{(n)}\}_{n=1}^{2}, \mathbf{y}, \mathbf{e}\}$ in an alternating fashion and finally the Lagrange multipliers $\{\mathbf{h}^{(n)}\}_{n=1}^{4}$ are updated. The ADMM solver of (3.8) is outlined in Algorithm 3. Algorithm 3 terminates when the reconstruction error $\left\|\mathbf{t} - \mathbf{U}_{\mathbf{J}^{(i)}}\mathbf{c}_{t+1}^{(1)} - \mathbf{U}_{\mathbf{A}^{(i)}}\mathbf{c}_{t+1}^{(2)} - \mathbf{e}_{t+1}\right\|_2^2 / \left\|\mathbf{t}\right\|_2^2$ is less than a predefined threshold ϵ or the number of iterations reached.

3.2.4 Scalable RJIVE

The computational complexity of the vanilla JIVE as well as the $L1$-RJIVE and NN-$L1$-RJIVE at each iteration is $O(\max(q^2J, qJ^2)) + \sum_{i=1}^{M} O(\max(d^{(i)^2}J, d^{(i)}J^2)) = O(\max(q^2J, qJ^2))$, due to the SVD. Clearly, this is computationally prohibitive when dimension of the images $\{d^{(i)}\}_{i=1}^{M}$ becomes very large, e.g., 22500 in our case. To alleviate the aforementioned computational complexity issue and at the same time learn the orthonormal bases that are used for reconstruction , we propose to factorise the matrices $\mathbf{J}, \{\mathbf{A}^{(i)}\}_{i=1}^{M}$ as products of orthonormal basis matrices $\mathbf{B} \in \mathbb{R}^{(d^{(1)}+\cdots d^{(M)})\times W_{\mathbf{J}}}, \mathbf{B}^T\mathbf{B} = \mathbf{I}$, $\{\mathbf{D}^{(i)} \in \mathbb{R}^{d^{(i)}\times W_{\mathbf{A}}^{(i)}}\mathbf{D}^{(i)^T}\mathbf{D}^{(i)} = \mathbf{I}\}_{i=1}^{M}$ and low-rank coefficients matrices $\mathbf{G}, \{\mathbf{C}^{(i)}\}_{i=1}^{M}$ such that $\mathbf{J} = \mathbf{B}\mathbf{G}$ and $\{\mathbf{A}^{(i)} = \mathbf{D}^{(i)}\mathbf{C}^{(i)}\}_{i=1}^{M}$. It can be easily shown that the constraints are now written as $\{\mathbf{J}\mathbf{A}^{(i)^T}\}_{i=1}^{M} = \mathbf{G}\mathbf{C}^{(i)^T} = \mathbf{0}$ and $\operatorname{rank}(\mathbf{J}) = \operatorname{rank}(\mathbf{B}\mathbf{G}) = \operatorname{rank}(\mathbf{G}) = r$. In addition, due to the unitary invariance property of the nuclear norm we have $\left\|\mathbf{A}^{(i)}\right\|_* = \left\|\mathbf{D}^{(i)}\mathbf{C}^{(i)}\right\|_* = \left\|\mathbf{C}^{(i)}\right\|_*$. Thus, by incorporating the factorisations of joint and individual components the optimisation problem now

Algorithm 3: ADMM solver of (3.8) (RJIVE-based Reconstruction)

```
Input      : Input sample $\mathbf{t}$. Orthonormal bases $\mathbf{B}^{(i)} \in \mathbb{R}^{d^{(i)} \times W_{\mathbf{J}}^{(i)}}, \mathbf{D}^{(i)} \in \mathbb{R}^{d^{(i)} \times W_{\mathbf{A}}^{(i)}}$.
             Parameters $\lambda, \rho$.
Output     : Clean reconstructed image $\mathbf{y}$.
Initialise: Set $\{\mathbf{v}_0^{(n)}, \mathbf{c}_0^{(n)}\}_{n=1}^{2}$, $\{\mathbf{h}_0^{(n)}\}_{n=1}^{4}$, $\mathbf{y}_0$, and $\mathbf{e}_0$ to zero vectors, $t = 0$, $\mu_0 > 0$.
while not converged do
    for n=1:2 do
        $\mathbf{v}_{t+1}^{(n)} = \mathcal{S}_{\frac{1}{\mu_t}}\left[\mathbf{c}_t^{(n)} - \frac{\mathbf{h}_t^{(n)}}{\mu_t}\right]$;
    $\tilde{\mathbf{t}}_1 = \mathbf{t} - \mathbf{D}^{(i)}\mathbf{c}_t^{(2)} - \mathbf{e}_t + \mathbf{h}_t^{(3)}\mu_t^{-1}$;
    $\tilde{\mathbf{t}}_2 = \mathbf{y} - \mathbf{D}^{(i)}\mathbf{c}_t^{(2)} + \mathbf{h}_t^{(4)}\mu_t^{-1}$;
    $\mathbf{c}_{t+1}^{(1)} = \frac{\mathbf{B}^{(i)T}\left(\tilde{\mathbf{t}}_1 + \tilde{\mathbf{t}}_2\right) + \mathbf{v}_{t+1}^{(1)} + \mathbf{h}_t^{(1)}\mu_t^{-1}}{3}$;
    $\tilde{\mathbf{t}}_1 = \mathbf{t} - \mathbf{B}^{(i)}\mathbf{c}_{t+1}^{(1)} - \mathbf{e}_t + \mathbf{h}_t^{(3)}\mu_t^{-1}$;
    $\tilde{\mathbf{t}}_2 = \mathbf{y} - \mathbf{B}^{(i)}\mathbf{c}_{t+1}^{(1)} + \mathbf{h}_t^{(4)}\mu_t^{-1}$;
    $\mathbf{c}_{t+1}^{(2)} = \frac{\mathbf{D}^{(i)T}\left(\tilde{\mathbf{t}}_1 + \tilde{\mathbf{t}}_2\right) + \mathbf{v}_{t+1}^{(2)} + \mathbf{h}_t^{(2)}\mu_t^{-1}}{3}$;
    $\mathbf{y}_{t+1} = \max\left(\mathbf{B}^{(i)}\mathbf{c}_{t+1}^{(1)} + \mathbf{D}^{(i)}\mathbf{c}_{t+1}^{(2)} - \mathbf{h}_t^{(4)}/\mu_t, 0\right)$;
    $\mathbf{e}_{t+1} = \mathcal{S}_{\frac{\lambda}{\mu_t}}\left[\mathbf{t} - \mathbf{B}^{(i)}\mathbf{c}_{t+1}^{(1)} - \mathbf{D}^{(i)}\mathbf{c}_{t+1}^{(2)} + \mathbf{h}_t^{(3)}\mu_t^{-1}\right]$;
    $\mathbf{h}_{t+1}^{(1)} = \mathbf{h}_t^{(1)} + \mu_t(\mathbf{v}_{t+1}^{(1)} - \mathbf{c}_{t+1}^{(1)})$;
    $\mathbf{h}_{t+1}^{(2)} = \mathbf{h}_t^{(2)} + \mu_t(\mathbf{v}_{t+1}^{(2)} - \mathbf{c}_{t+1}^{(2)})$;
    $\mathbf{h}_{t+1}^{(3)} = \mathbf{h}_t^{(3)} + \mu_t(\mathbf{t} - \mathbf{B}^{(i)}\mathbf{c}_{t+1}^{(1)} - \mathbf{D}^{(i)}\mathbf{c}_{t+1}^{(2)} - \mathbf{e}_{t+1})$;
    $\mathbf{h}_{t+1}^{(4)} = \mathbf{h}_t^{(4)} + \mu_t(\mathbf{y} - \mathbf{B}^{(i)}\mathbf{c}_{t+1}^{(1)} - \mathbf{D}^{(i)}\mathbf{c}_{t+1}^{(2)})$;
    $\mu_{t+1} = \min(\mu_t\rho, 10^7)$;
```

is as follows:

$$
\begin{aligned}
&\min_{\mathbf{B},\mathbf{G},\{\mathbf{D}^{(i)},\mathbf{C}^{(i)},\mathbf{\Delta}^{(i)}\}_{i=1}^{M},\mathbf{E}} \sum_{i=1}^{M}\left\|\mathbf{\Delta}^{(i)}\right\|_{*} + \lambda\left\|\mathbf{E}\right\|_1, \\
&\text{s.t. } \mathbf{X} = \mathbf{B}\mathbf{G} + \left[\left(\mathbf{D}^{(1)}\mathbf{C}^{(1)}\right)^T \cdots, \left(\mathbf{D}^{(M)}\mathbf{C}^{(M)}\right)^T\right]^T + \mathbf{E}, \\
&\qquad \operatorname{rank}(\mathbf{G}) = r, \mathbf{B}^T\mathbf{B} = \mathbf{I}, \\
&\qquad \{\mathbf{\Delta}^{(i)} = \mathbf{C}^{(i)}, \mathbf{G}\mathbf{C}^{(i)T} = \mathbf{0}, \mathbf{D}^{(i)T}\mathbf{D}^{(i)} = \mathbf{I}\}_{i=1}^{M},
\end{aligned}
\tag{3.10}
$$

where $\{\mathbf{\Delta}^{(i)} \in \mathbb{R}^{d^{(i)} \times J}\}_{i=1}^{M}$, $\{\mathbf{\Delta}^{(i)} = \mathbf{A}^{(i)}\}_{i=1}^{M}$ are auxiliary variables and the corresponding constraints, respectively. The augmented Lagrangian function that corresponds to problem (3.10) is given by:

$$\begin{aligned}\mathcal{L}(\{\mathbf{B}, \mathbf{G}\{\mathbf{C}^{(i)}, \mathbf{D}^{(i)}, \mathbf{\Delta}^{(i)}, \mathbf{Z}^{(i)}\}_{i=1}^{M}, \mathbf{E}, \mathbf{\Gamma}\}) &= \sum_{i=1}^{M} \left\| \mathbf{\Delta}^{(i)} \right\|_* + \lambda \left\| \mathbf{E} \right\|_1 \\ + \frac{\mu}{2} \left\| \mathbf{X} - \mathbf{BG} - \left[\left(\mathbf{D}^{(1)} \mathbf{C}^{(1)} \right)^T \cdots, \left(\mathbf{D}^{(M)} \mathbf{C}^{(M)} \right)^T \right]^T - \mathbf{E} + \frac{\mathbf{\Gamma}}{\mu} \right\|_F^2 &- \frac{1}{2\mu} \left\| \mathbf{\Gamma} \right\|_F^2 \\ + \sum_{i=1}^{M} \left(\frac{\mu}{2} \left\| \mathbf{\Delta}^{(i)} - \mathbf{C}^{(i)} + \frac{\mathbf{Z}^{(i)}}{\mu} \right\|_F^2 - \frac{1}{2\mu} \left\| \mathbf{Z}^{(i)} \right\|_F^2 \right)&,\end{aligned} \tag{3.11}$$

where $\mathbf{\Gamma}$ and $\{\mathbf{Z}^{(i)}\}_{i=1}^{M}$ are the Lagrangian multipliers related to the equality constraints of (3.10). Similarly to the previous problems, (3.10) is minimised with respect to each variable in an alternating fashion and finally the Lagrange multipliers are updated. The ADMM solver of the proposed SRJIVE method is outlined in Algorithm 4.

The computational complexity of Algorithm 4 is dominated by the cost of the SVD involved in the computation of SVT and Procrustes operators in Steps 4 and 5, respectively. Thus, the computational complexity of each iteration is $O(\max(W_{\mathbf{J}}^2 J, W_{\mathbf{J}} J^2))$ and $O(\max(q^2 W_{\mathbf{J}}, q W_{\mathbf{J}}^2))$, respectively. Given that $W_{\mathbf{J}} \ll q = d^1 + \cdots d^{(M)}$ (in this work $q = 225000$ and $W_{\mathbf{J}} \leq 600$), which implies $W_{\mathbf{J}} J + q W_{\mathbf{J}} \ll qJ$, the proposed scalable version of JIVE, i.e., the SRJIVE has a significantly reduced computational cost compared to that of JIVE and RJIVE.

Regarding the convergence of the presented Algorithms 1, 2, 4 there is currently no theoretical proof known for the ADMM in problems with more than two blocks of variables. However ADMM has been applied successfully in non-linear optimisation problems in practice [16, 213, 214, 215, 216]. In addition, the thorough experimental evaluation of the proposed methods, presented in Section 3.3, indicates that the obtained solutions are good for the data that RJIVE tested.

3.2.5 RJIVE with missing values and application to face aging using 3D Morphable Models

3D Morphable Models (3MMs) can be employed to extract aligned texture representations from standard 2D face images, as described in Section 2.1.4. In brief, after fitting a 3DMM on a test image, a UV texture map can be calculated by projecting the reconstructed 3D shape on the image plane and sampling the image at the locations of the shape's vertexes. The UV space constitutes a densely aligned

Algorithm 4: ADMM solver of (3.10) (Scalable NN-$L1$-RJIVE, SRJIVE).

```
Input     : Data $\{\mathbf{X}^{(i)} \in \mathbb{R}^{d^{(i)} \times J}\}_{i=1}^{M}$. Rank of joint component $r$. Number of bases to be
            extracted from the Joint and Individual components $W_{\mathbf{J}}$ and $W_{\mathbf{A}}^{(i)}$, respectively.
            Parameter $\rho$.
Output    : Orthonormal Joint and Individual bases matrices $\mathbf{B}$, $\{\mathbf{D}^{(i)}\}_{i=1}^{M}$. Coefficient matrices
            $\mathbf{G}$, $\{\mathbf{C}^{(i)}\}_{i=1}^{M}$.
Initialise: Set $\mathbf{G}_0$, $\mathbf{B}_0$, $\{\mathbf{\Delta}_0^{(i)}, \mathbf{D}_0^{(i)}, \mathbf{C}_0^{(i)}, \mathbf{Z}_0^{(i)}\}_{i=1}^{M}$, $\mathbf{E}_0$, $\mathbf{\Gamma}_0$ to zero matrices, $t = 0$, $\mu_0 > 0$.
$\mathbf{X} = \left[\mathbf{X}^{(1)^T}, \cdots, \mathbf{X}^{(M)^T}\right]^T$;
while not converged do
    $\mathbf{M} = \mathbf{B}_t^T \left(\mathbf{X} - \left[\left(\mathbf{D}_t^{(1)}\mathbf{C}_t^{(1)}\right)^T \cdots, \left(\mathbf{D}_t^{(M)}\mathbf{C}_t^{(M)}\right)^T\right]^T - \mathbf{E}_t + \mu_t^{-1}\mathbf{\Gamma}_t\right)$;
    $[\mathbf{U}, \mathbf{\Sigma}, \mathbf{V}] = \mathrm{svd}(\mathbf{M})$;
    $\mathbf{G}_{t+1} = \mathcal{Q}_r[\mathbf{M}]$;
    $\mathbf{B}_{t+1} = \mathcal{P}\left[\left(\mathbf{X} - \left[\left(\mathbf{D}_t^{(1)}\mathbf{C}_t^{(1)}\right)^T \cdots, \left(\mathbf{D}_t^{(M)}\mathbf{C}_t^{(M)}\right)^T\right]^T - \mathbf{E}_t + \mu_t^{-1}\mathbf{\Gamma}_t\right)\mathbf{G}_{t+1}^T\right]$;
    $\mathbf{M} = \mathbf{X} - \mathbf{B}_{t+1}\mathbf{G}_{t+1} - \mathbf{E}_t + \mu_t^{-1}\mathbf{\Gamma}_t$;
    for n=1:M do
        $\mathbf{D}_{t+1}^{(i)} = \mathcal{P}\left[\mathbf{M}^{(i)}\mathbf{C}_t^{(i)^T}\right]$;
        $\mathbf{C}_{t+1}^{(i)} = 0.5\left(\mathbf{D}_{t+1}^{(i)^T}\mathbf{M}^{(i)} + \mathbf{\Delta}_t^{(i)} + \mu_t^{-1}\mathbf{Z}_t^{(i)}\right)(\mathbf{I} - \mathbf{V}\mathbf{V}^T)$;
        $\mathbf{\Delta}_{t+1}^{(i)} = \mathcal{D}_{\frac{1}{\mu_t}}\left[\mathbf{C}_{t+1}^{(i)} - \mu^{-1}\mathbf{Z}_t^{(i)}\right]$;
        $\mathbf{Z}_{t+1}^{(i)} = \mathbf{Z}_{t+1}^{(i)} + \mu_t\left(\mathbf{\Delta}_{t+1}^{(i)} - \mathbf{C}_{t+1}^{(i)}\right)$;
    $\mathbf{E}_{t+1} = \mathcal{S}_{\frac{\lambda}{\mu_t}}\left[\mathbf{X} - \mathbf{B}_{t+1}\mathbf{G}_{t+1} - \left[\left(\mathbf{D}_{t+1}^{(1)}\mathbf{C}_{t+1}^{(1)}\right)^T \cdots, \left(\mathbf{D}_{t+1}^{(M)}\mathbf{C}_{t+1}^{(M)}\right)^T\right]^T + \mu_t^{-1}\mathbf{\Gamma}_t\right]$;
    $\mathbf{\Gamma}_{t+1} = \mathbf{\Gamma}_t + \mu_t\left(\mathbf{X} - \mathbf{B}_{t+1}\mathbf{G}_{t+1} - \left[\left(\mathbf{D}_{t+1}^{(1)}\mathbf{C}_{t+1}^{(1)}\right)^T \cdots, \left(\mathbf{D}_{t+1}^{(M)}\mathbf{C}_{t+1}^{(M)}\right)^T\right]^T - \mathbf{E}_{t+1}\right)$;
    $\mu_{t+1} = \min(\rho \cdot \mu_t, 10^7)$;
    $t = t + 1$;
```

domain for 2D images, which is ideal for use with CA methodologies, such as the one discussed in this Chapter. However, extracting the 3D texture from a 2D image in this way leads to incomplete 3D texture representations, mainly, due to the presence of self-occlusions, especially when the person depicted in the image is not in a frontal pose. Therefore, data collected with the aforementioned technique include missing values. In order to specify the location (i.e., image coordinates) of the missing values in a UV texture image, a self-occlusion mask for each image is calculated by casting a ray from the camera to each vertex of the reconstructed shape. Each element of the extracted mask denotes whether a value of the UV texture map is missing or not (please see the *Input* rows of Figure 3.14 for examples of the extracted UV space).

Even thought, the RJIVE can robustly recover joint and individual components in the presence of sparse non-Gaussian errors of large magnitude, it is not able to handle data with missing values. To overcome this limitation of the RJIVE we propose the RJIVE-Missing (RJIVE-M). Consider M data-sets of different ages $\{\mathbf{X}^{(i)} \in \mathbb{R}^{d^{(i)} \times J}\}_{i=1}^{M}$, with $\mathbf{x}_j^{(i)} \in \mathbb{R}^{d^{(i)}}$, being a vectorised form of the j-th gross corrupted and incomplete UV texture, $j = 1, \ldots, J$, that displays a face within the i-th age group, $i = 1, \ldots M$. The goal of the RJIVE-M is not only to recover the joint and individual components but also to perform completion on the UV textures with missing values. To this end, problem (3.10) is reformulated to the following one:

$$\begin{aligned}
&\min_{\mathbf{B},\mathbf{G},\{\mathbf{D}^{(i)},\mathbf{C}^{(i)},\boldsymbol{\Delta}^{(i)}\}_{i=1}^{M},\mathbf{E}} \sum_{i=1}^{M} \left\| \boldsymbol{\Delta}^{(i)} \right\|_{*} + \lambda \left\| \mathbf{W} \circ \mathbf{E} \right\|_{1}, \\
&\text{s.t. } \mathbf{X} = \mathbf{BG} + \left[\left(\mathbf{D}^{(1)}\mathbf{C}^{(1)} \right)^{T} \cdots, \left(\mathbf{D}^{(M)}\mathbf{C}^{(M)} \right)^{T} \right]^{T} + \mathbf{E}, \\
&\quad \operatorname{rank}(\mathbf{G}) = r, \mathbf{B}^{T}\mathbf{B} = \mathbf{I}, \\
&\quad \{\boldsymbol{\Delta}^{(i)} = \mathbf{C}^{(i)}, \mathbf{G}\mathbf{C}^{(i)^{T}} = \mathbf{0}, \mathbf{D}^{(i)^{T}}\mathbf{D}^{(i)} = \mathbf{I}\}_{i=1}^{M},
\end{aligned} \tag{3.12}$$

where $\circ$ denotes the Hadamard (element-wise) product and $\mathbf{W} = \left[\mathbf{W}^{(1)^{T}}, \cdots, \mathbf{W}^{(M)^{T}} \right]^{T} \in \mathbb{R}^{q \times J}$, $\mathbf{W}^{(i)} = [\mathbf{w}_1^{(i)}, \mathbf{w}_2^{(i)}, \cdots, \mathbf{w}_J^{(i)}] \in \{0, 1\}^{q \times J}$, with $\mathbf{w}_j^{(i)}$ being a vectorised form of the self-occlusion mask that corresponds to the j-th UV texture of the i-th data-set. The Algorithm for solving the proposed RJIVE-M problem is similar to the SRJIVE one and has the same complexity and convergence criterion. The only difference is in the updating step of the error matrix $\mathbf{E}$. More specifically, the following additional step is performed after executing the step 12 of the Algorithm 4: $\mathbf{E} = \mathbf{W} \circ \mathbf{E} + \overline{\mathbf{W}} \circ \left[\mathbf{X} - \mathbf{B}_{t+1}\mathbf{G}_{t+1} - \left[\left(\mathbf{D}_{t+1}^{(1)}\mathbf{C}_{t+1}^{(1)} \right)^{T} \cdots, \left(\mathbf{D}_{t+1}^{(M)}\mathbf{C}_{t+1}^{(M)} \right)^{T} \right]^{T} + \mu_t^{-1}\boldsymbol{\Gamma}_t \right]$.

Similarly, the presented RJIVE-based reconstruction method can be also extended to handle missing values in a test image. To this end, given a test sample with missing values (e.g., face UV texture) and the vectorised form of the corresponding occlusion mask $\mathbf{w}$, problem (3.8) is extended to the following one:

$$\begin{aligned}&\min_{\{\mathbf{c}^{(n)},\mathbf{v}^{(n)}\}_{n=1}^{2},\mathbf{y}\geq\mathbf{0}} \sum_{n=1}^{2}\left\|\mathbf{v}^{(n)}\right\|_1 + \lambda\left\|\mathbf{w}\circ\mathbf{e}\right\|_1,\\ &\text{s.t. } \{\mathbf{v}^{(n)}=\mathbf{c}^{(n)}\}_{n=1}^{2}\\ &\quad \mathbf{t}=\mathbf{B}^{(i)}\mathbf{c}^{(1)}+\mathbf{D}^{(i)}\mathbf{c}^{(2)}+\mathbf{e},\ \mathbf{y}=\mathbf{B}^{(i)}\mathbf{c}^{(1)}+\mathbf{D}^{(i)}\mathbf{c}^{(2)}\end{aligned} \tag{3.13}$$

An ADMM-based solver similar to the Algorithm 3 is employed in order to solve problem (3.13). More specifically, the update step of the error vector performed in step 11 of the Algorithm 3 is followed by the following one: $\mathbf{e}_{t+1}=\mathbf{w}\circ\mathbf{e}+\overline{\mathbf{w}}\circ\left[\mathbf{t}-\mathbf{B}^{(i)}\mathbf{c}_{t+1}^{(1)}-\mathbf{D}^{(i)}\mathbf{c}_{t+1}^{(2)}+\mathbf{h}_t^{(3)}\mu_t^{-1}\right]$.

3.3 Experiments

The performance of the proposed RJIVE method is assessed on synthetic data corrupted by sparse, non-Gaussian noise (Section 3.3.1), as well as on data captured under constrained and "in-the-wild" conditions with applications to (a) *facial expression synthesis*, (b) 2D and (c) 3D *face age progression*. Parameters selected for these experiments are summarised in Table 3.1.

Table 3.1: Parameters used in the conducted experiments.

Section	r	$W_{\mathbf{J}}^{(i)}$	$W_{\mathbf{A}}^{(i)}$	λ	ϵ
3.3.2 (controlled)	20	70	70	$\frac{1}{\sqrt{\max(q,J)}}=0.03$	10^{-5}
3.3.2 (in-the-wild)	150	300	300		
3.3.3	300	600	600		

3.3.1 Synthetic

In this section, the ability of RJIVE to robustly recover the common and individual components of synthetic data corrupted by sparse non-Gaussian noise, is tested. To this end, sets of matrices $\{\mathbf{X}^{(i)}=\mathbf{J}_*^{(i)}+\mathbf{A}_*^{(i)}+\mathbf{E}_*^{(i)}\in\mathbb{R}^{d^{(i)}\times J}\}_{i=1}^{2}$ of varying dimensions were generated. In more detail, a rank-r joint component $\mathbf{J}_*\in\mathbb{R}^{(q=d^{(1)}+d^{(2)})\times J}$ was created from a random matrix $\mathbf{X}=[\mathbf{X}^{(1)^T},\mathbf{X}^{(2)^T}]^T\in\mathbb{R}^{q\times J}$. Next, the orthogonal to $\mathbf{J}$ rank-$r^{(1)}$, $r^{(2)}$ common components $\mathbf{A}_*^{(1)}$ and $\mathbf{A}_*^{(2)}$ were computed by $[\mathbf{A}_*^{(1)^T},\mathbf{A}_*^{(2)^T}]^T=(\mathbf{X}-\mathbf{J}_*)(\mathbf{I}-\mathbf{V}\mathbf{V}^T)$, where $\mathbf{V}$ was formed from the first r columns of the row space of $\mathbf{X}$. $\mathbf{E}_*^{(i)}$ is a sparse error matrix with 20% non-zero entries being sampled independently from $\mathcal{N}(0,1)$.

Table 3.2: Quantitative recovering results produced by JIVE [14], COBE [15], RCICA [16], $L1$-RJIVE (3.4), and NN-$L1$-RJIVE (3.6) under Gaussian and gross non-Gaussian noise. Each compared method was applied on the same data generated by utilising each set of parameters. The average relative reconstruction error (RRE) and computation time (in CPU seconds) were computed by repeating the experiment 10 times.

$\left(d^{(1)}, d^{(2)}, J, r, r^{(1)}, r^{(2)}\right)$	**Method**	$RRE(\mathbf{J})$		$RRE(\mathbf{A})$		**Time** (in CPU seconds)	
		non-Gaussian	**Gaussian**	**non-Gaussian**	**Gaussian**	**non-Gaussian**	**Gaussian**
$(500, 500, 500, 5, 10, 10)$	**COBE**	3.6403	1.0927	1.0975	1.0002	0.06	0.07
	JIVE	0.5424	$1.3558e-04$	0.9349	$2.0782e-04$	4.62	1.22
	RCICA	–	–	$7.1379e-07$	$5.6337e-03$	1.14	1.36
	L1-RJIVE	$5.5628e-08$	$1.3558e-04$	$3.5073e-08$	$2.0782e-04$	3.11	4.78
	NN-L1-RJIVE	$5.1515e-08$	$1.4720e-04$	$4.3416e-08$	$3.3904e-04$	4.06	5.06
	SRJIVE	$2.7770e-08$	$1.6564e-04$	$3.8706e-08$	$2.0012e-04$	0.91	1.97
$(1000, 1000, 1000, 10, 20, 20)$	**COBE**	4.9982	1.08555	1.1890	0.9982	0.122	0.11
	JIVE	0.8398	$1.8880e-04$	1.4810	$2.9261e-04$	14.69	4.45
	RCICA	–	–	$6.7260e-07$	$9.6371e-04$	6.36	5.84
	L1-RJIVE	$8.5033e-08$	$1.8879e-04$	$5.5423e-08$	$2.9260e-04$	8.23	18.01
	NN-L1-RJIVE	$9.3804e-08$	$2.0738e-04$	$7.6262e-08$	$1.1801e-04$	17.34	23.11
	SRJIVE	$6.8905e-08$	$2.3406e-04$	$6.0017e-08$	$1.2041e-04$	3.99	9.05
$(2000, 2000, 2000, 20, 40, 40)$	**COBE**	6.9981	1.088417	1.3469	0.9976	0.83	0.69
	JIVE	1.3961	$2.6525e-04$	2.1977	$4.1133e-04$	203.25	49.06
	RCICA	–	–	$5.9359e-05$	$7.6497e-03$	48.51	49.86
	L1-RJIVE	$1.2305e-07$	$2.6525e-04$	$1.0512e-07$	$4.1133e-04$	142.44	160.21
	NN-L1-RJIVE	$8.8570e-08$	$2.9010e-04$	$9.1058e-08$	$5.6000e-04$	110.36	120.01
	SRJIVE	$9.7434e-08$	$2.7074e-04$	$1.0117e-07$	$5.1173e-04$	18.96	43.07

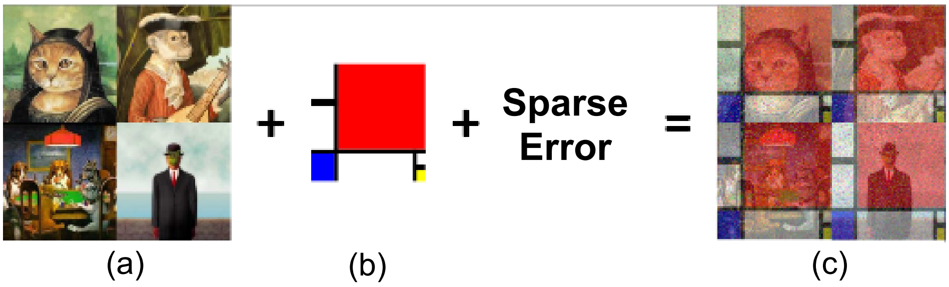


Figure 3.4: Procedure followed to generate data contaminated by sparse, non-Gaussian noise (c). Four images (a) were superimposed by a common painting (b) and added sparse noise sampled form $\mathcal{N}(0, 1)$ for the 20% of the pixels of each image.

The Relative Reconstruction Error (RRE) of the recovered components is employed as evaluation metric, which is defined as:

$$RRE(\mathbf{Q}) = \sum_{i=1}^{q} \left(\frac{\left\| \mathbf{Q}_*^{(i)} - \mathbf{Q}^{(i)} \right\|_F^2}{\left\| \mathbf{Q}_*^{(i)} \right\|_F^2} \right)^2, \tag{3.14}$$

where $\mathbf{Q}^{(i)}$ represents the recovered joint or individual components for the i-th sample and $\mathbf{Q}_*^{(i)}$ the ground truth ones. The RRE of the joint and individual components achieved by both $L1$-RJIVE and Nuclear-Norm regularised (NN-$L1$-RJIVE) for a varying number of dimensions, joint and individual

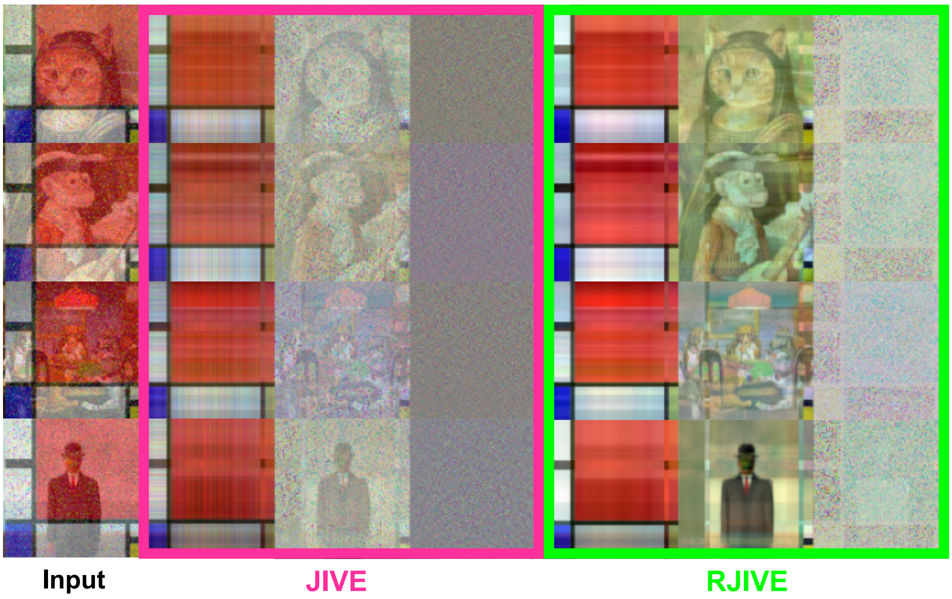


Figure 3.5: Joint, individual components and error matrices produced by the compared JIVE and RJIVE methods. JIVE is able to recover the main parts of the common structure (1st column of the highlighted blocks), however including more noise compared to RJIVE. Additionally, the recovered individual components of JIVE (2nd column of the highlighted blocks) are heavily contaminated by noise, which is not the case for the results of RJIVE. Lastly, RJIVE recovers much more efficiently sparse, non-gaussian errors in the error components (3rd column of the highlighted blocks.)

ranks, are reported in Table 3.2. The corresponding RRE obtained by JIVE [14], COBE [15], and RCICA [16] are also presented. As it can be seen, the proposed methods accurately recovered both the joint and individual components. It is worth mentioning that the NN-$L1$-RJIVE successfully recovered all components by utilising only the true rank of the joint component. In contrast, all the other methods require knowledge regarding the true rank for both joint and individual components. Furthermore, the SRJIVE achieved same results to the NN-$L1$-RJIVE by reducing the computation times more that five times. Based on the performance of SRJIVE on the synthetic data, we decided to exploit it in the experiments described bellow and referred to as RJIVE hereafter.

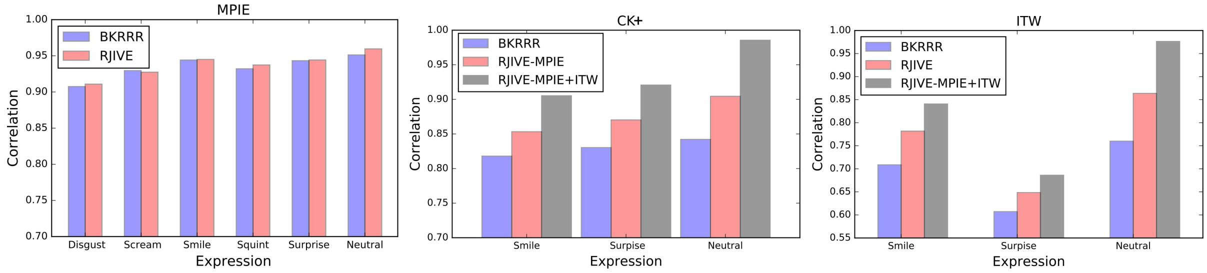
Furthermore, we tested the RJIVE on synthetic data contaminated by Gaussian error. The RJIVE, can implicitly handle data contaminated by Gaussian noise by vanishing the error term. That is by setting the regularizer λ in problems (3.4), (3.6), (3.10) $\lambda \to \infty$ i.e. $\mathbf{E} = 0$. In such case, the Frobenius norms corresponds to the equality constraints $\mathbf{X} = \mathbf{J} + [\mathbf{A}^{(1)^T}, \cdots, \mathbf{A}^{(M)^T}]^T + \mathbf{E}$, $\mathbf{X} = \mathbf{B}\mathbf{G} + [(\mathbf{D}^{(1)}\mathbf{C}^{(1)})^T, \cdots, \mathbf{A}^{(M)^T}]^T + \mathbf{E}$ appearing in the corresponding augmented Lagrangian functions are deemed as the appropriate regularizer for handling Gaussian noise. The RRE of all compared methods are reported in Table 3.2. As it can be seen, the proposed methods accurately recovered both the joint and individual components.

The efficiency of the JIVE and RJIVE methods was qualitatively evaluated on real data contaminated by sparse, non-Gaussian noise. In order to generate the corrupted data we firstly superimposed the paintings of Figure 3.4(a) with the painting appeared in Figure 3.4(b) and subsequently a sparse error matrix was added. In each image the error matrix has 20% non-zero entries being sampled independently from $\mathcal{N}(0, 1)$. Then, the concatenation of the generated paintings (Figure 3.4(c)) was given as input to the JIVE and RJIVE. The joint and individual components as well as the corresponding error matrices obtained from the compared methods are depicted in Figure 3.5. As it can be observed, RJIVE accurately recovered both the joint and individual components. In contrast, the joint components extracted from JIVE are not accurate, while the corresponding individual ones are contaminated by the spare error. This is due to the fact that the JIVE is not robust to sparse, non-Gaussian noise.

3.3.2 Facial Expression Synthesis

In this section, we investigate the ability of the RJIVE to synthesise a set of different expressions of a given facial image. Consider M data-sets where each one contains images of different subjects that depict a specific expression. In order to effectively recover the joint and common components, the faces of each data-set should be put in correspondence. Thus, their $N = 68$ facial landmark points are localised using the detector [217, 218] and subsequently employed to compute a mean reference shape. Then, the faces of each data-set are warped into corresponding reference shape by using the piecewise affine warp function $\mathcal{W}(\cdot)$ [73]. After applying the RJIVE on the warped data-sets, the recovered components can be used for synthesising M different expressions of an unseen subject. To do that, the new (unseen) facial image is warped to the reference frame that corresponds to the expression that we want to synthesise and subsequently is given as input to the solver of (3.8).

The performance of RJIVE in FES task is assessed by conducting inner- and cross-databases experiments on MPIE [219], CK+ [220], and “in-the-wild” facial images collected from the internet (ITW). The synthesised expressions obtained by RJIVE are compared to those obtained by the BKRRR [221] method. In particular, the BKRRR is a regression-based method that learns a mapping from the ‘Neut-



(a) (b) (c)

Figure 3.6: Mean average correlation achieved by JIVE and BKRRR methods on (a) MPIE, (b) CK+, and (c) ITW databases. For (a) a subset of 89 subjects of MPIE (6 experiments) was used to train the compared methods and the remaining 58 were used for testing, synthesising all expressions. (b) and (c) present cross-dataset results with methods trained on 69 subjects of MPIE and tested on CK+ and ITW images (blue and red bars). In both experiments RJIVE outperforms BKRRR in terms of Mean Average Correlation. Augmenting the training set of RJIVE with ITW images, Average Correlation is further increased for both experiments (b) and (c) (grey bar).

ral' expression to the target ones. Then, given the 'Neutral' face of an unseen subject, new expressions are synthesised by employing the corresponding learnt regression functions. The performance of the compared methods is measured by computing the correlation between the vectorised forms of true images ($\mathbf{t}_{true}$) and the reconstructed ones ($\mathbf{t}_{rec}$):

$$\text{Cr}(\mathbf{t}_{true}, \mathbf{t}_{rec}) = \frac{\mathbf{t}_{true}^T \mathbf{t}_{rec}}{\sqrt{\|\mathbf{t}_{true}\|_2^2 \, \|\mathbf{t}_{rec}\|_2^2}}. \tag{3.15}$$

Controlled Conditions

In the first experiment, 534 frontal images of MPIE database that depict 89 subjects under six expressions (i.e., 'Neutral', 'Scream', 'Squint', 'Surprise', 'Smile', 'Disgust') were employed to train both RJIVE and BKRRR. Then, all expressions of 58 unseen subjects from the same database were synthesised by using their images that correspond to 'Neutral' expressions. In Figure 3.6(a) the average correlations obtained by the compared methods for the different expressions are visualised. As it can be seen the proposed RJIVE method achieves the same accuracy to BKRRR without learning any kind of mappings between the different expressions of the same subject. Specifically, the RJIVE extracts only the individual components of each expression and the common one.

Furthermore, the performance of both methods is compared by performing a cross-database experiment on CK+ database. More specifically, we employed the 'Neutral', 'Smile', and 'Surprised' images

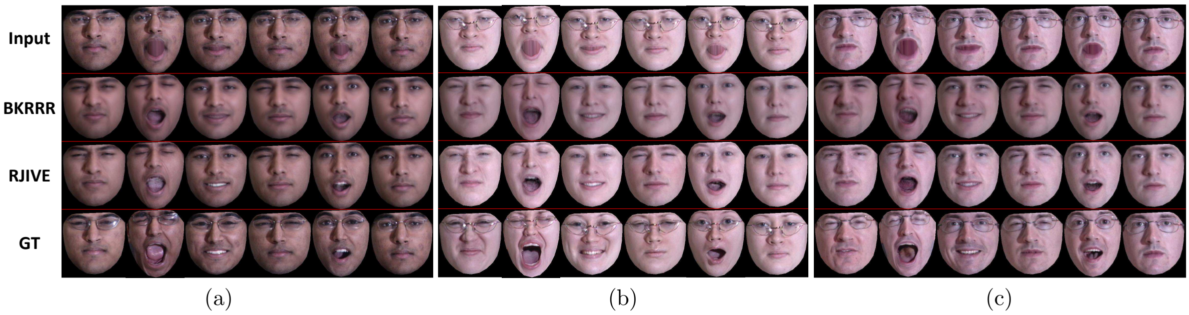


Figure 3.7: synthesised expressions of MPIE's subject (a) '014' (b) '015' and (c) '250' produced by the BKRRR and RJIVE methods. The methods were trained with images from the 6 available expressions from 89 subjects of MPIE. RJIVE is able to produce more fine-grained details compared to BKRRR, such as eyes, the mouth interior and details of the skin.

of MPIE for training purposes while images of 69 subjects (three images per subject) of CK+ were used as test ones. In Figure 3.6(b) we can see that RJIVE outperforms by a large margin the BKRRR. This is due to the fact that the BKRRR performs the regression based on how close the unseen 'Neutral' face is to the training ones. Thus, in cases that the unseen subjects (e.g., subjects of CK+) present enough differences compared to the training ones (e.g., subjects of MPIE), the synthesised expressions are characterised as non-accurate. Figure 3.6(c), which includes results on expression synthesis on "in-the-wild" images will be discusses in the following subsection (In-The-Wild Conditions). Laslty, the synthesised expressions of subjects '014', '015' and 250 from MPIE produced by the BKRRR and RJIVE are visualised in Figure 3.7. Clearly, the proposed method produces expressive images of higher quality compared to the BKRRR.

The accuracy of the components recovered by JIVE and RJIVE in FES is also qualitatively assessed. Figure 3.8 displays the obtained components and the corresponding error matrices after applying JIVE and RJIVE on images used in the previous experiments which were additionally contaminated by sparse errors. Clearly, the proposed RJIVE method successfully recovered all the components. It is worth mentioning that the RJIVE removes the sparse noise and outliers e.g., occlusions due to eyeglasses (please see the red boxes of Figure 3.8). Clearly, the JIVE is not able to cope with the additive noise and occlusions.

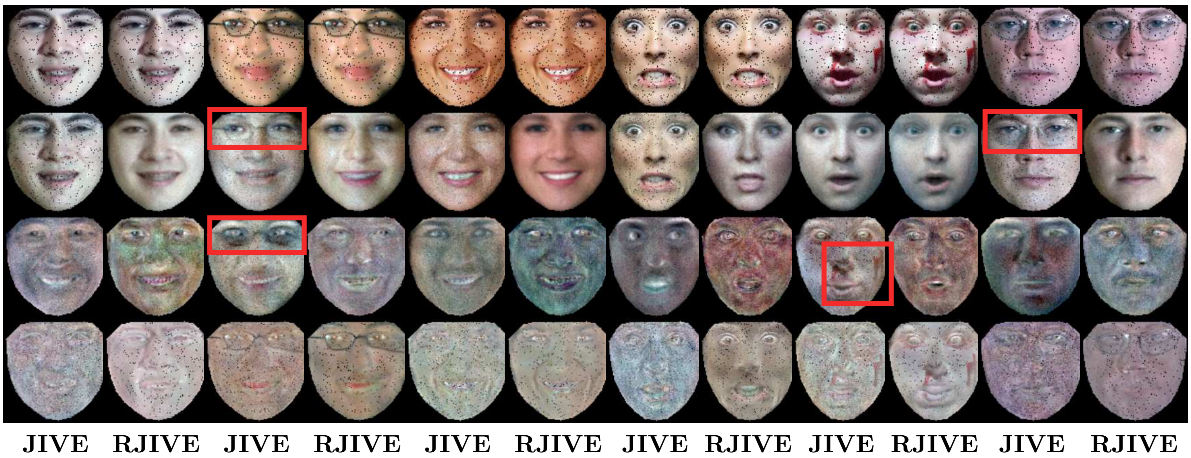


Figure 3.8: Joint, individual components and error matrices produced by the compared JIVE and RJIVE methods on "in-the-wild" images. The results show that RJIVE successfully decomposes inputs (1st row) into clean shared (2nd row) and individual (3rd row) components, as well as an error component (4th row), while JIVE is not able to disentangle sparse, non-gaussian noise from the clean information. The red rectangles demonstrate examples of sparse details not handled by JIVE.

In-The-Wild Conditions

As an additional experiment, we collected from the internet 180 images depicting 60 subjects with 'Surprise', 'Smile', and 'Neutral' expressions (three images for each subject). Then, all the expressions were generated by employing the 'Neutral' images and the BKRRR and RJIVE methods trained on MPIE. Figure 3.6(c) depicts the obtained correlations for each subject. Clearly, the RJIVE outperforms the BKRRR. Compared to the previous experiments, there is a drop in performance for both methods. This is attributed to the fact that the methods were trained by employing only images captured under controlled condition. Thus, synthesising expressions of "in-the-wild" images is a very difficult task.

In order to alleviate this problem we can augment the training set with "in-the-wild" images. Although the RJIVE can be trained from "in-the-wild" images of different subjects, this is not the case of BKRRR, which requires the correspondence of expressions across the training subjects. Collecting "in-the-wild" images of same subjects under different expressions is a very tedious task. In order to improve the performance of RJIVE, we augmented the training set with another 1200 images from WWB database [222] (400 images for each expression). As it can be observed in Figure 3.6(c), the "in-the-wild" train set improved the accuracy of RJIVE in both CK+ and ITW data-sets. Figure 3.9 depicts examples synthesised "in-the-wild" expressions produced by the RJIVE. The images from the 'Input' column were given as input to the RJIVE and subsequently the synthesised expressions were

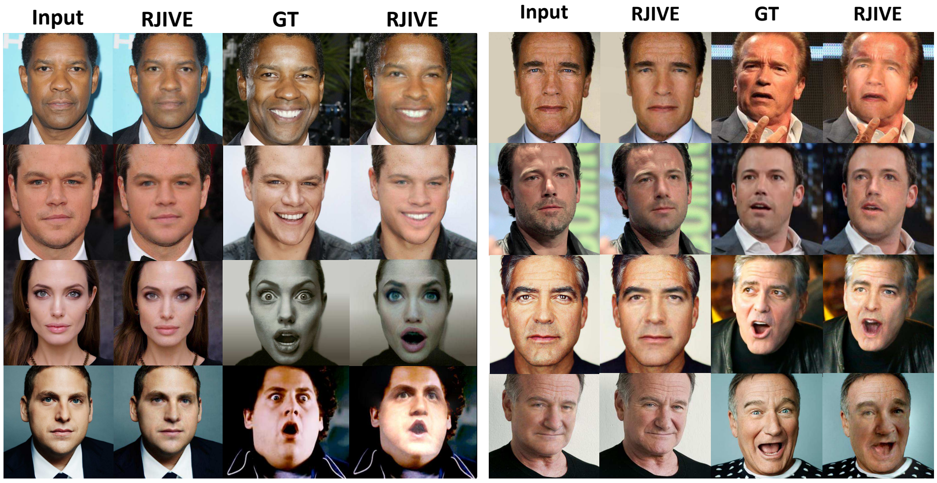


Figure 3.9: Synthesised in-the-wild expressions produced by the RJIVE method. RJIVE was trained with images of 69 subjects from MPIE, as well as 1200 "in-the-wild" images. RJIVE consistently reproduces facial details of the input images.

warped and fused with the actual images [223]. Clearly, the produced expressions are characterised by high quality of both expression and identity information. It is worth mentioning that RJIVE synthesise almost perfectly the input images without using any kind of information about the depicted subject.

3.3.3 Face Age Progression In-The-Wild

2D age progression of an unseen subject

Face age progression consists in synthesising plausible faces of subjects at different ages. It is considered as a very challenging task due to the fact that the face is a highly deformable object and its appearance drastically changes under different illumination conditions, expressions, and poses. Various databases that contain faces at different ages have been collected in the last couple of years [224, 225]. Although these databases contain huge number of images, they have some limitations including limited images for each subject that cover a narrow range of ages and noisy age labels, since most of them have been collected by employing automatic procedures (crawlers). In order to overcome the aforementioned problems, we collected a new data-set called Age In-The-Wild (ATW). More specifically, 19.000 images that depict 540 subjects from 0 to 100 years old were collected from the internet. Subsequently, each image was manually annotated in terms of age and identity of the depicted subject. On average, there are 36 images that span 55 years for each subject.

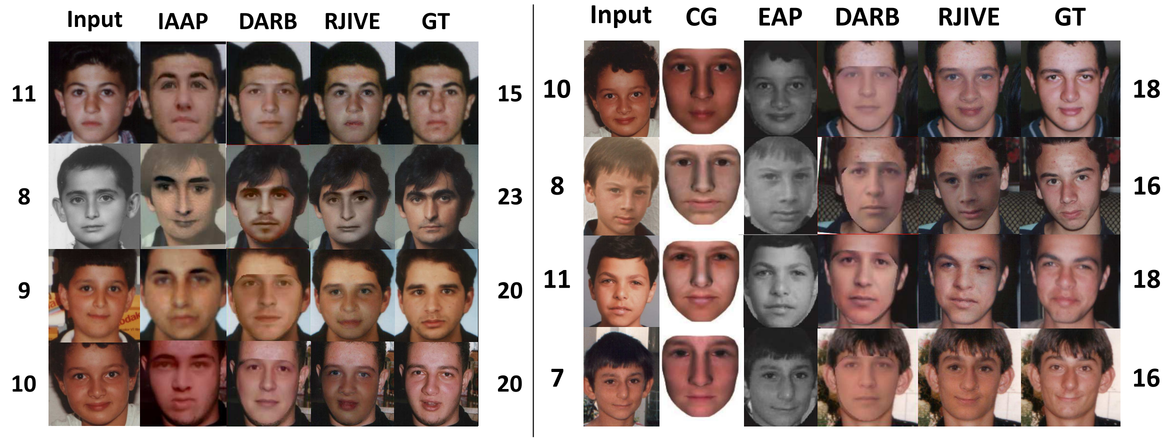


Figure 3.10: Progressed faces produced by the compared methods on the FG-NET database. ATW dataset was used for training. RJIVE maintains the facial shape and identity characteristics and produces images closer to the ground truth compared to the rest methods.

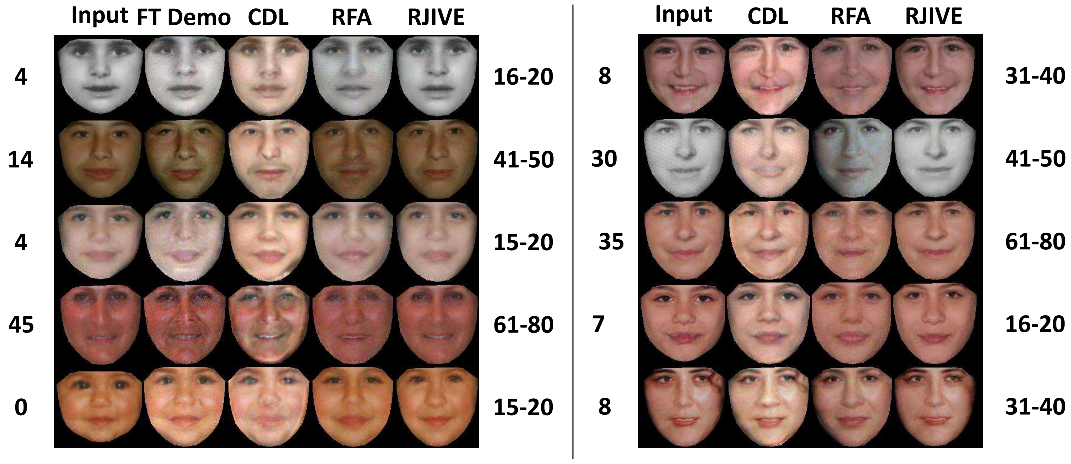


Figure 3.11: Progressed faces produced by the compared methods on the FG-NET database. ATW dataset was used for training. RJIVE better maintains face and identity characteristics such as skin tone and eye shape compared to the rest methods.

In order to train the RJIVE, the ATW was divided into $M = 10$ age groups: $0-3$, $4-7$, $8-15$, $16-20$, $21-30$, $31-40$, $41-50$, $51-60$, $61-70$, and $71-100$. Then, following the same procedure as in FES task, the RJIVE was employed to extract the joint and common components from the warped images. The performance of RJIVE in face age progression "in-the-wild" is qualitatively assessed conducting experiments on images from the FG-NET database [226]. To this end, we compare the performance of RJIVE with the Illumination Aware Age Progression (IAAP) method [200], Coupled Dictionary Learning (CDL) method [201], Deep Ageing with Restricted Boltzmann Machines (DARB) method [202], Craniofacial Growth (CG) [203] model, Exemplar-based Age Progression (EAP) [204]

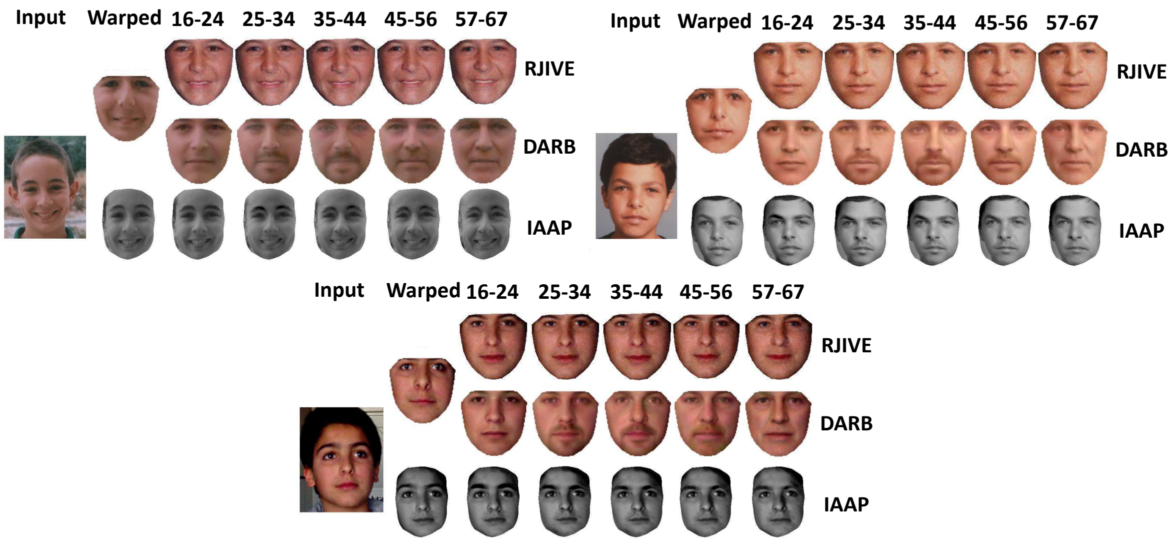


Figure 3.12: Comparisons between the IAAP, DARB, and RJIVE methods. ATW dataset was used for training. RJIVE maintains the facial shape and identity, while DARB produces heavy identity shift and IAAP synthesises black and white images.

method, Face Transformer (FT Demo) [227], and Recurrent Face Aging (RFA) method [205]. In Figures 3.10, 3.11 progressed images produced by the compared methods are depicted. Note, that all the progressed faces have been warped back and fused with the actual ones. Figure 3.12 depicts faces synthesised by the DARB, IAAP, and RJIVE methods. By observing the results, it can be clearly seen that the identity information is not preserved in case of DARB. In particular, the progressed faces of all subjects for a specific age group are very similar between them. Instead, the identity information remains in the faces produced by the proposed RJIVE method, while the age progression result looks more natural. Finally, progressed example faces in all the age-groups produced the RJIVE are visualised in Figure 3.13.

3D age progression of an unseen subject

Here, the ability of the proposed RJIVE-M method to perform 3D face age progression is demonstrated. Similarly to the 2D face age progression experiments presented previously, the ATW database was divided into $M = 6$ age groups (21-30, 31-40, 41-50, 51-60, 61-70, 71+) and used to train the RJIVE-M. In order to acquire the 3D training data for this task the 3DMM-ITW [33] was employed. The optimal shape and camera parameters were extracted by fitting the model to each one of the images of all age groups as described in Section 2.1.2. In order to recover 3D shapes of high quality, we used

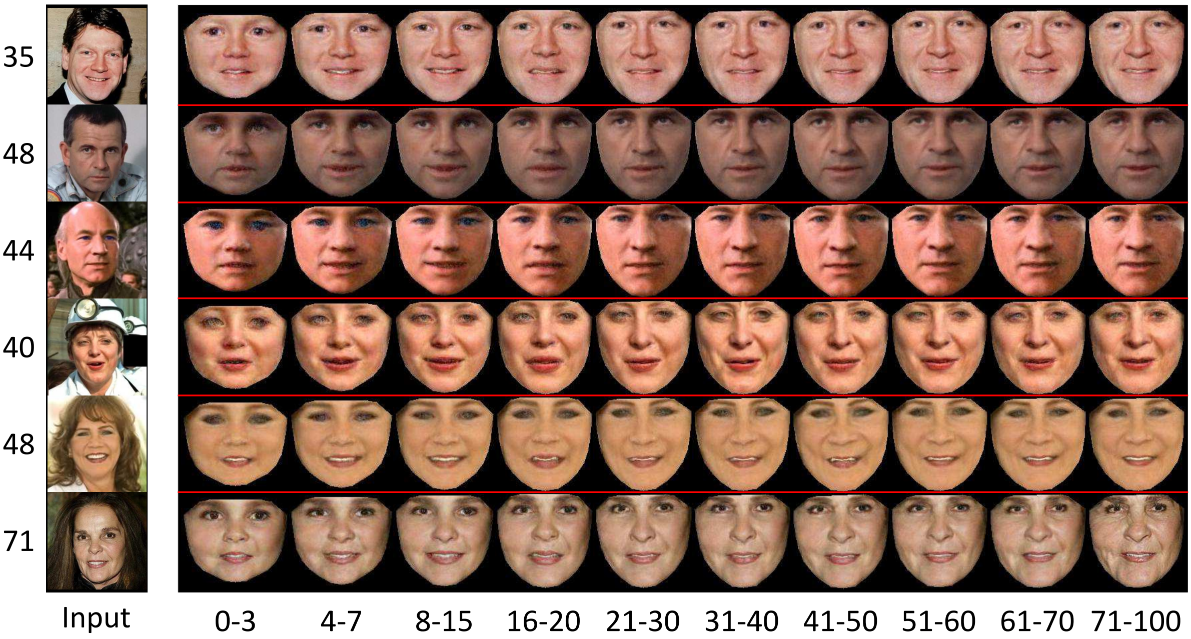


Figure 3.13: Progressed faces produced by the proposed RJIVE method. Trianed on the ATW dataset, RJIVE produces plausible progressed faces across all age groups from arbitrary input face images.

the age and gender specific versions of the LSFM shape model introduced in [44] to describe identity and the blendshapes of [58] to describe facial expressions. After recovering the 3D shape of each face, we computed the self-occlusion mask by using ray-tracing (see *Input* rows of Figure 3.14). Then, the completed joint and individual components of the grossly corrupted and incomplete UV textures were obtained by employing the RJIVE-M. The joint components obtained by applying a variant of JIVE with missing values, i.e. JIVE-M, and the RJIVE-M on UV textures are displayed in Figure 3.14. By observing the results, we can clearly see that the RJIVE successfully removed the occlusions produced from eyeglasses and fingers in all images. This is attributed to the fact that the matrix $L1$-norm was adopted in RJIVE, which effectively handles sparse noise of possibly large magnitude.

Similarly, to the 2D face aging experiment we can apply the RJIVE-M to the recovered UV maps to learn components that can be used to age the UV texture of a test unseen subject. Since, the 3D shapes are produced by the LSFM model they neither have missing values nor are contaminated by noise. Hence, for training aging components for the 3D shape we used standard JIVE.

In the test phase, the 3D facial shape of the test face is obtained by using the 3DMM-ITW algorithm [33]. Then, the UV texture and the corresponding self-occlusion mask are computed by employing the

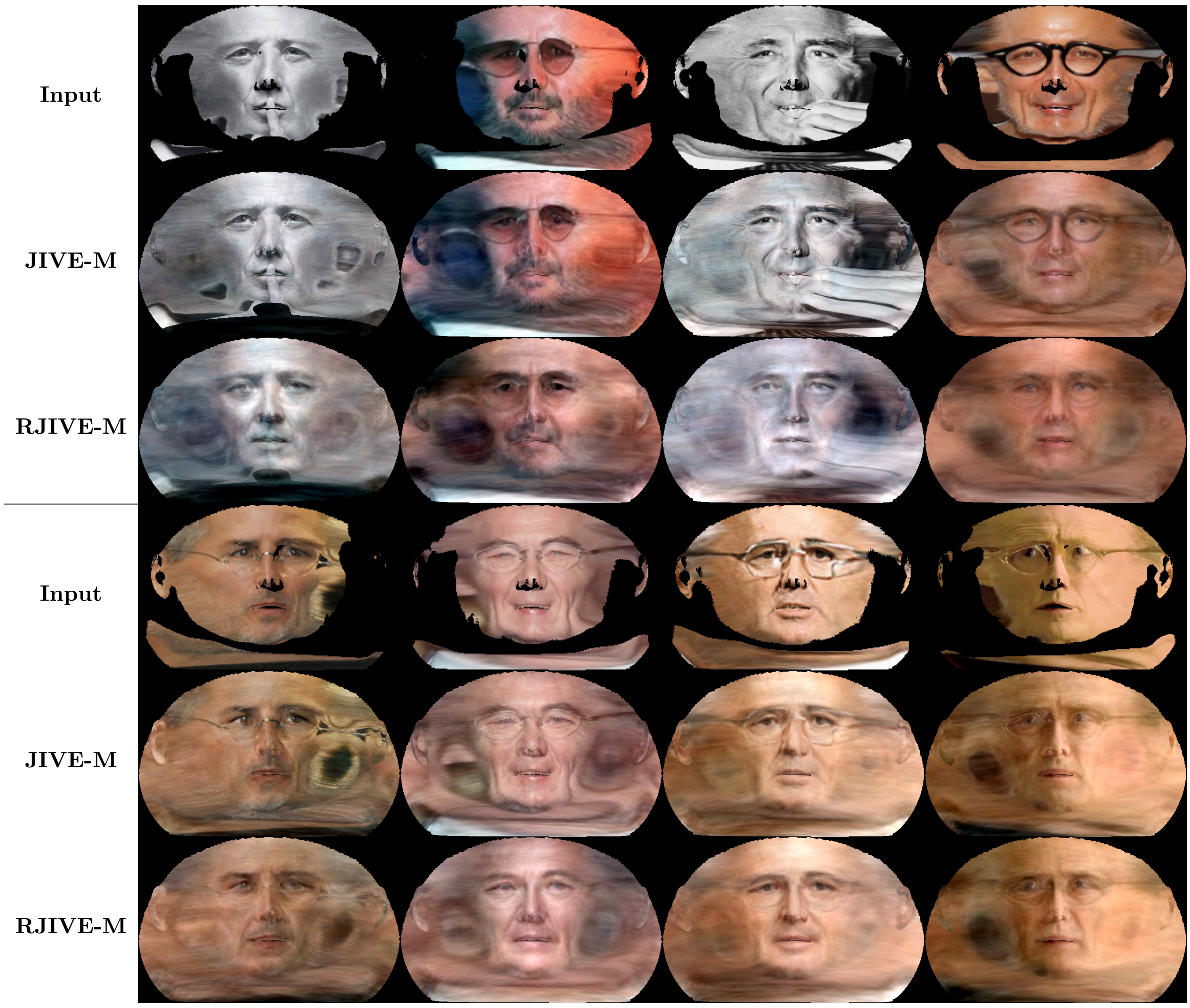


Figure 3.14: Input images and corresponding joint components produced by the compared JIVE-M and RJIVE-M methods. As it can be observed the proposed method is able to remove occlusions such as fingers and glasses from unwrapped UV maps, producing cleaner textures.

recovered 3D shape. The progression of the texture of the test subject in an age group is obtained by solving the problem (3.13) (for the shape we use the problem in (3.1)). Progressed unseen subjects in all age groups, projected back in the image plane, are visualised in Figure 3.15. After calculating a progressed 3D texture image and 3D shape the result face model is projected back in the image plane using the camera parameters initially acquired by fitting the 3DMM-ITW in the test image.

Figure 3.16 presents additional results that demonstrate the ability of the RJIVE-M to perform not only age progression but also completion. For each subject the original and two side poses are depicted. The extracted by the 3DMM-ITW 3D face model of the input image is displayed on the first row. By

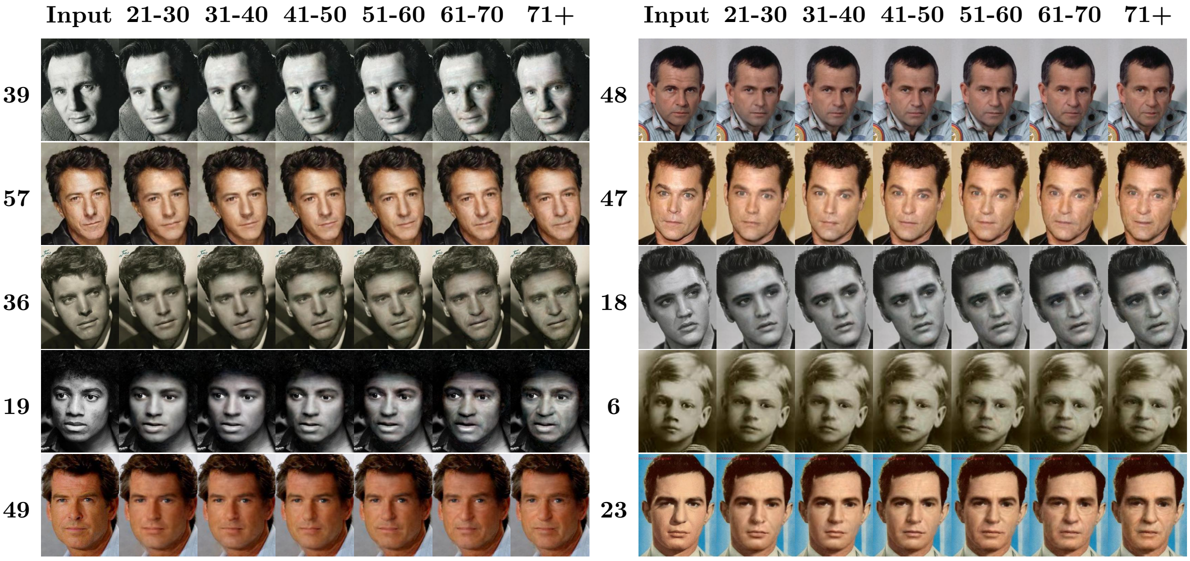


Figure 3.15: Progressed faces produced by the proposed RJIVE-M method, projected back in the image plane using the 3D face shapes and camera parameters acquired by fitting the 3DMM-ITW. The same fitting technique was employed to extract incomplete UV maps from ATW which were used for training. The progressed faces demonstrate plausible age-specific details such as wrinkles, while maintaining the identity of the subjects.

observing the results it becomes obvious that due to the self-occlusions, the instance of the 3D model with pose different to the input one contains huge areas of missing values (black color). This is not the case for the progressed and completed results produced by the RJIVE-M (second row). As it can be seen, the completion of the regions with missing data blends naturally with the rest of the texture which proves the significant representational power of the bases extracted from RJIVE-M.

Age-invariant face verification in-the-wild

The performance of the RJIVE is also quantitatively assessed by conducting age-invariant face verification experiments. Following the successfully used verification protocol of the LFW database [228], we propose four new age-invariant face verification protocols based on the proposed ATW database. Each one of the protocols was created by splitting the ATW database into 10 folds, with each fold consisting of 300 intra-class pairs and 300 inter-class pairs. The essential difference between these protocols is that in each protocol the age difference of each pair's faces is equal to a predefined value i.e., {5 ages, 10 ages, 20 ages, 30 ages}.

In order to assess the performance of RJIVE, the following procedure was performed. For each fold

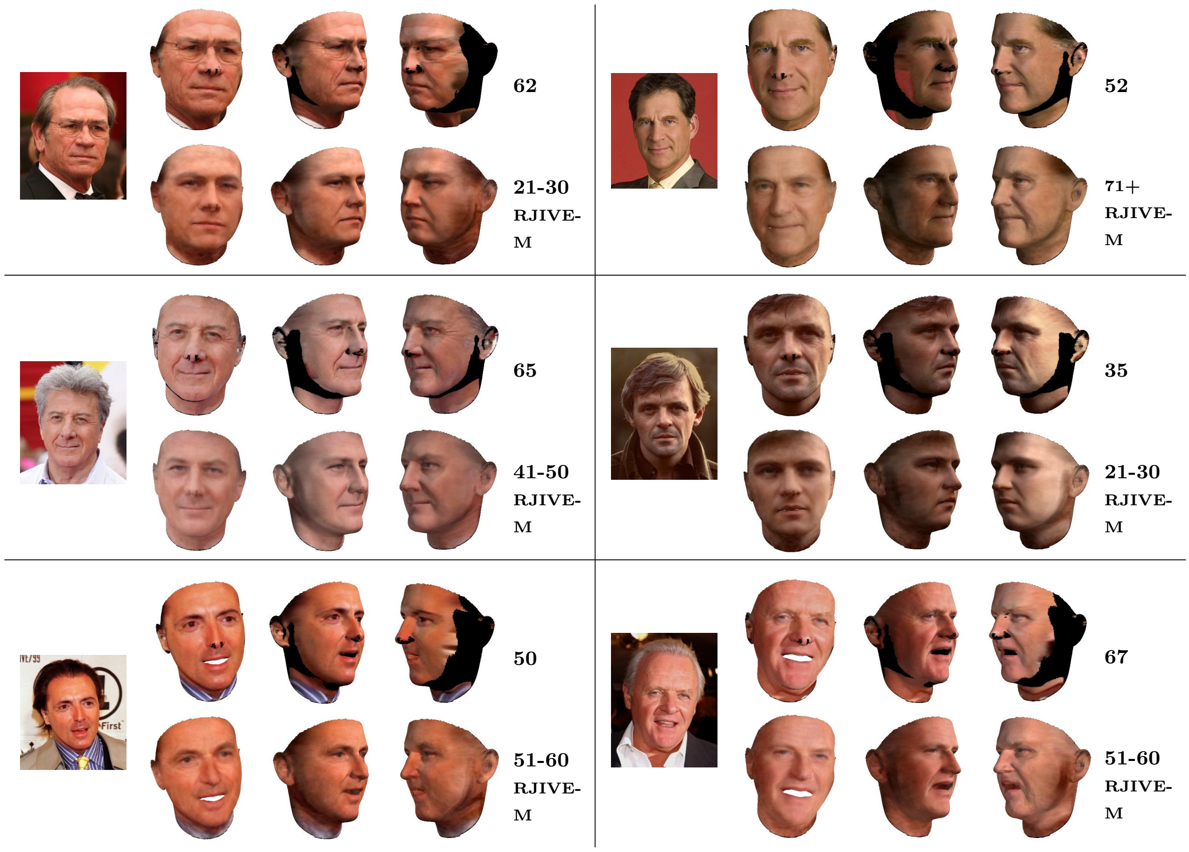
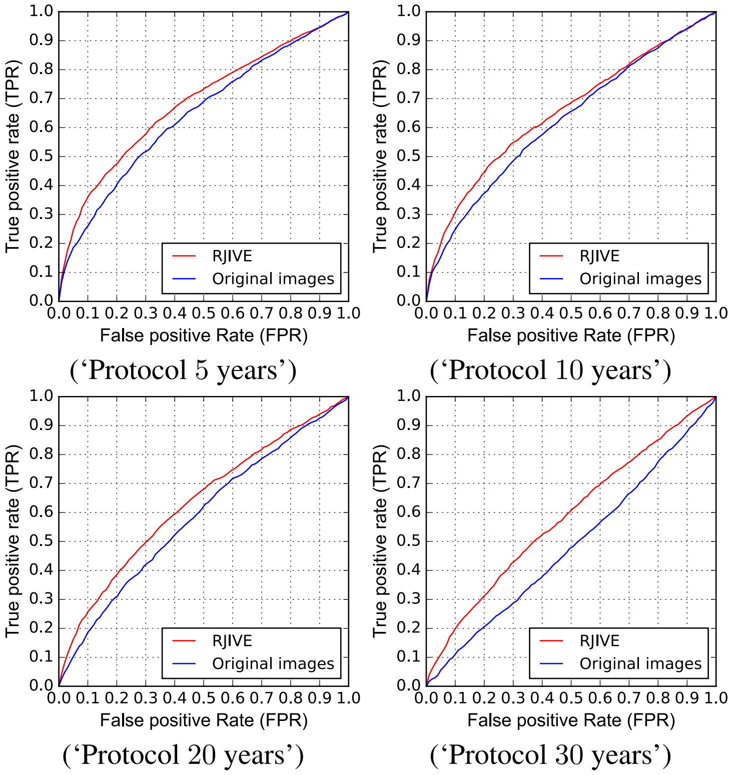


Figure 3.16: Progressed and completed 3D texture images, produced by the proposed RJIVE-M method. The 3D face models are visualised in the original and two side poses, so that the missing and the completed data become visible. RJIVE is able to produce plausible progressed faces with age-specific details, while simultaneously competing the missing information of UV textures.

of a specific protocol the training images were split into $M = 10$ age-groups and subsequently the RJIVE was employed on their warped version in order to extract the joint and individual components. All images of each training pair were then progressed into $M = 10$ age groups resulting into 10 new pairs. The progressed images of six subjects are depicted in Figure 3.13. As we wanted to represent each pair by using a single feature, gradients orientations were extracted from the corresponding images and subsequently the mean value of their cosine difference was employed as the pair's feature. M different Support Vector Machines (SVM) were trained by utilising the extracted features. Finally, the scores produced by all the SVMs were fused by using SVM.

In Figure 3.17, Receiver Operating Characteristic (ROC) curves computed based on the 10 folds of each one of the proposed protocols are depicted. The corresponding mean classification accuracy and



('Protocol 5 years')

('Protocol 10 years')

('Protocol 20 years')

('Protocol 30 years')

Figure 3.17: ROC curves of RJIVE on the proposed four protocols. 'Original images' corresponds to the results obtained by employing the actual images. Augmenting the existing images with RJIVE synthesised ones for the tasks of age-invariant face verification boosts performance compared to using just the originals. In particular, larger performance improvements are seen for protocols referring to the larger age differences.

Area Under Curve (AUC) are reported in Table 3.3. In order to assess the effect of progression, the results obtained by utilising only the original images are also provided. Some interesting observations are drawn from the results. Firstly, the improvement in accuracy validates that the identity information of the face remains after the RJIVE-based progression. Furthermore, the improvement in accuracy is higher when the age difference of images of each pair is big enough. For instance, the improvement in accuracy in 'Protocol 30 years' is higher than the corresponding in 'Protocol 5 years'. Finally, the produced results justify that the problem of age-invariant face verification becomes more difficult when the age difference is very large (e.g., 30 years).

The performance of RJIVE in age-invariant face verification is also compared against the IAAP [200] by conducting experiment on the FG-NET database. The experimental protocol employed is as follows. By selecting images where the depicted subjects are older than the age of 18 years, we created

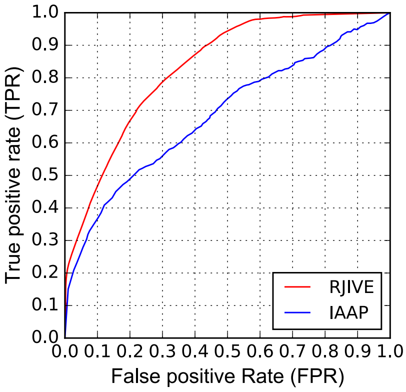


Figure 3.18: ROC curve of the RJIVE and IAAP on FG-NET database. RJIVE outperforms the compared method, meaning that it produces more realistic face images, as validated by our face verification task.

Table 3.3: Mean AUC and Accuracy on the proposed four protocols. Employing RJIVE synthesised images in the tasks of age-invariant face verification boosts performance compared to using only the available real images. In particular, larger performance improvements are seen for protocols referring to the larger age differences.

	RJIVE		Original Images	
Protocol	AUC	Accuracy	AUC	Accuracy
5 years	0.686	0.637	0.646	0.609
10 years	0.654	0.621	0.624	0.591
20 years	0.633	0.598	0.585	0.552
30 years	0.584	0.552	0.484	0.495

a subset of the FG-NET database consisting of 518 images. Then, based on the selected images we created 1250 intra-class pairs i.e., the images of each pair depict the same subject under different ages, and another 1250 inter-class pairs. The experiment protocol was finally created by dividing the pairs on 5 folds with each fold containing 250 intra-class pairs and 250 inter-class ones. All images were then progressed by employing the RJIVE and IAAP methods. A similar to previous experiment procedure was followed in order to perform the age-invariant verification. The produced ROC curves are displayed in Figure 3.18. As it can be observed the proposed RJIVE method outperforms the IAAP by a large margin indicating that the RJIVE produces progressed images of high quality without removing the identity information.

3.4 Conclusions

A general framework for robust recovering of joint and individual variance among several data-sets possibly contaminated by gross non-Gaussian errors and incomplete has been presented in this Chapter. Four different models namely, $L1$-RJIVE, NN-$L1$-RJIVE, SRJIVE, and RJIVE-M have been proposed. Furthermore, based on the recovered components from training data, two novel optimisation problems that extracts the joint and individual components of an unseen test sample, are introduced. The effectiveness of the RJIVE was first tested by conducting experiments on synthetic data. Then, extensive experiments were conducted on facial expression synthesis and 2D an 3D face age progression by utilising five data-sets captured under both controlled and “in-the-wild” conditions. The experimental results validate the effectiveness of the proposed RJIVE method over the state-of-the-art.

Chapter 4

SliderGAN: Synthesising Expressive Face Images by Sliding 3D Blendshape Parameters

Contents

4.1 Introduction

Interactive editing of the expression of a face in an image has countless applications including but not limited to movies post-production, computational photography, face recognition (i.e. expression neutralisation) etc. In computer graphics facial motion editing is a popular field, nevertheless mainly revolves around constructing person-specific models having a lot of training samples [229]. Recently, the advent of machine learning, and especially Deep Convolutional Neural Networks (DCNNs) provide very exciting tools making the community to re-think the problem. In particular, recent advances in Generative Adversarial Networks (GANs) provide very exciting solutions for image-to-image (i2i) translation.

i2i translation, i.e. the problem of learning how to transform aligned image pairs, has attracted a lot of attention during the last few years [17, 18, 19]. The so-called pix2pix model and alternatives demonstrated excellent results in image completion etc. [17]. In order to perform i2i translation in absence of image pairs the so-called CycleGAN was proposed, which introduced a cycle-consistency loss [18]. CycleGAN could perform i2i translation between two domains only (i.e. in the presence of two discrete labels), utilising separate generators and discriminators for each mapping direction. The more recent StarGAN [19] extended the idea of cycle consistency further to accommodate multiple domains (i.e. multiple discrete labels) based on single generator and discriminator networks.

StarGAN can be used to transfer an expression to a given facial image by providing the discrete label of the target expression. Hence, it has quite small capabilities in expression editing and arbitrary expression transfer. Over the last few years, quite some deep learning related methodologies have been proposed for transforming facial images [19, 230, 20]. The most closely related work to us is the recent work [20] that proposed the GANimation model. GANimation follows the same line of research as StarGAN to translate facial images according to the activation of certain facial Action Units (AUs) and their intensities. According to [79], AUs is a system to taxonomise motion of the human facial muscles. Even though AU coding is a quite comprehensive model for describing facial motion, detecting AUs is currently an open problem both in controlled, as well as in unconstrained recording conditions [86, 87]. Recent AU detection techniques achieve around 50% F1 in EmotioNet challenge and from our experiments OpenFace [85] achieves lower than 20-25%. In particular, in unconstrained conditions the detection accuracy for certain AUs is not high-enough yet [86, 87], which affects the generation accuracy of GANimation. More specifically, GANimation's accuracy is related to both the AU detection, as well as the estimation of their intensity, since the generator is jointly trained and influenced by a network that performs detection and intensity estimation.

One of the reasons of the low accuracy of automatic annotation of AUs, is the lack of annotated data and the high cost of annotation which has to be performed by highly trained experts. Finally, even though AUs 10-28 model mouth and lip motion, only 10 of them can be automatically recognised i.e. AUs 10, 12, 14, 15, 17, 20, 23, 25, 26, 28. To make matters worse, the 10 AUs can only be recognised with low accuracy, thus they cannot describe all possible lip motion patterns produced during speech. Hence, GANimation cannot be used in straightforward manner for transferring speech.

In this work, we are motivated by the recent successes in 3D face reconstruction methodologies from “in-the-wild” images [30, 31, 32, 13], which make use of a statistical model of 3D facial motion by means of a set of linear blendshapes, and propose a methodology for facial image translation using GANs driven by the continuous parameters of the linear blendshapes. The linear blendshapes can describe both the motion that is produced by expression [59] and/or motion that is produced by speech [34]. On the contrary, neither discrete emotions nor facial Action Units can be used to describe the motion produced by speech or the combination of motion from speech and expression. We demonstrate that it is possible to transform a facial image along the continuous axis of individual expression and speech blendshapes.

Moreover, contrary to StarGAN, which uses discrete labels regarding expression, and GANimation, which utilises annotations with regards to action units, our methodology does not need any human annotations, as we operate using pseudo-annotations provided by fitting a 3D Morphable Model (3DMM) to images [13] (for expression deformations) or by aligning audio signals [34] (for speech deformations). Building on the automatic annotation process exploited by SliderGAN, a by-product of our training process is a very robust regression DCNN that estimates the blendshape parameters directly from images. This DCNN is extremely useful for expression and/or speech transfer as it can automatically estimate the blendshape parameters of target images.

i2i translation models have achieved photo-realistic results by utilising different GAN optimisation methods in literature. pix2pix employed the original GAN optimisation technique proposed in [1]. However, the loss function of GAN may lead to the vanishing gradients problem during the learning process. Hence, more effective GAN frameworks emerged that were employed by i2i translation methods. CycleGAN uses LSGAN, which builds upon GAN adopting a least squares loss function for the discriminator. StarGAN and GANimation use WGAN-GP [36], which enforces gradient clipping as a measure to regularise the discriminator. WGAN-GP, builds upon WGAN [105] which minimises an approximation of the Wasserstein distance to stabilise training of GANs.

A recent approach of efficient GAN optimisation which has been proven to enhance the texture quality in i2i translation and particularly in super-resolution problems [231], is the Relativistic GAN (RGAN) [35]. RGAN was suggested in order to train the discriminator to simultaneously decrease the probability that real images are real, while increasing the probability that the generated images are real. In our work, we incorporate RGAN in the training process of SliderGAN and demonstrate that it can improve the generator which produces more detailed results in the task of i2i translation for expression and speech synthesis, when compared to training with WGAN-GP. In particular, we employ the Relativistic average GAN (RaGAN) which decides whether an image is relatively more realistic than the others on average, rather than whether it is real or fake. More details, as well as the benefits from this mechanism are presented in Section 4.2.1.

To summarise, the proposed method includes quite a few novelties. First of all, we showcase that SliderGAN is able to synthesise smooth deformations of expression and speech in images by utilising 3D blendshape models of expression and speech respectively, as demonstrated in Figure 4.1. Moreover, it is the first time to the best of our knowledge that a direct comparison of blendshape and AU coding is presented for the task of expression and speech synthesis. In addition, our approach is annotation-free but offers much better accuracy than AUs-based methods. Furthermore, it is the first time that Relativistic GAN was employed for the task of expression and speech synthesis. We demonstrate in our results that SliderGAN trained with the RaGAN framework (SliderGAN-RaD) benefits towards producing more detailed textures, than when trained with the standard WGAN-GP framework (SliderGAN-WGP). Finally, we enhance the training of our model with synthesised data, leveraging

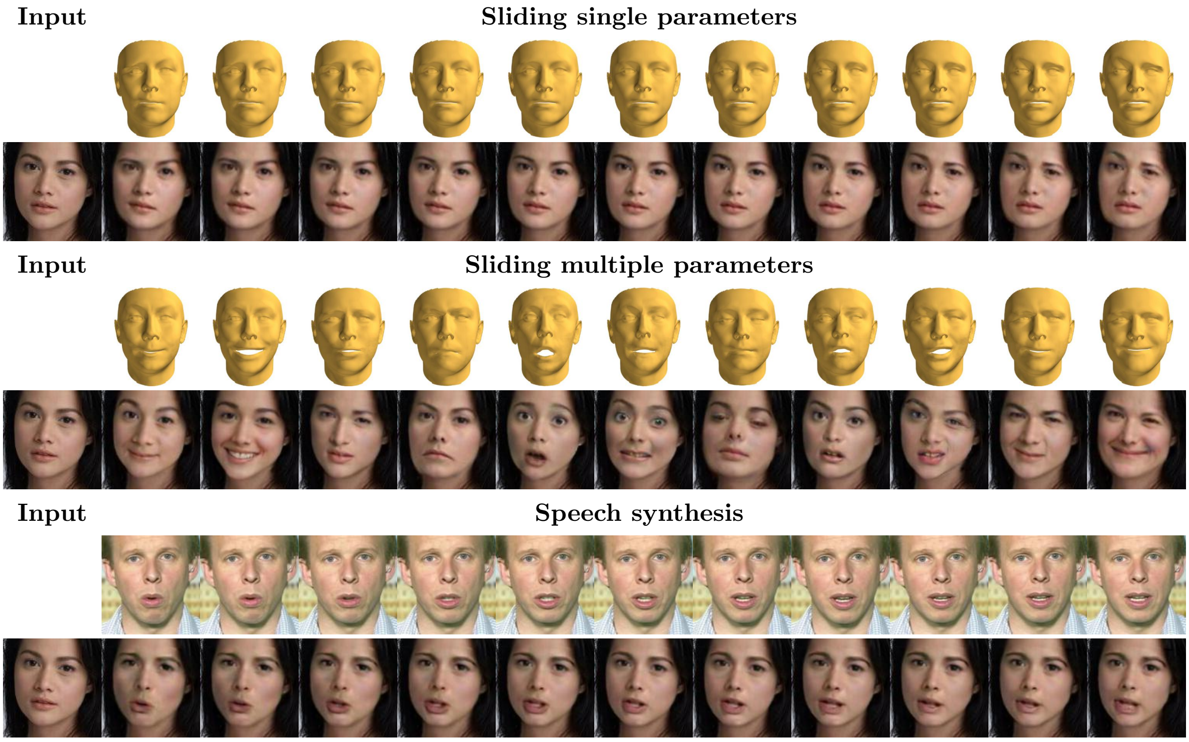


Figure 4.1: Expressive faces generated by sliding a single or multiple blendshape parameters in the normalised range $[-1, 1]$. Rows 1 and 3 depict 3D expressive faces generated by a linear blendshape model of natural face motion and a set of expression parameters. The corresponding edited images generated by SliderGAN using the same set of parameters are depicted in rows 2 and 4. As it is observed, the generated images accurately replicate the 3D faces' motion. The robustness of blendshape coding of facial motion allows SliderGAN to perform speech synthesis, as demonstrated in rows 5 (target speech) and 6 (synthesised speech), for which a 3D blendshape model of human speech was utilised.

the reconstruction capabilities of statistical shape models.

4.1.1 Facial Attribute Editing and Reenactment in Images

Over the past few years, quite some models have been proposed for the task of transforming images and especially facial attributes in images of faces, e.g. expression, pose, hair color, age, gender etc. A rough categorisation of them can be made depending on whether they are targeted to single image manipulation or to face reenactment in a sequence of frames.

Single image manipulation We have already discussed Pix2pix [17], CycleGAN [18], StarGAN [19] and GANimation [20] which are are all methodologies developed to tackle single image editing via i2i translation. Similarly, developed for single image manipulation, DIAT [232] uses an adversarial loss to learn a one directional mapping between images of two domains. ICGAN [99] is a conditional GAN for image attribute editing, which can handle multiple attributes with one generator. ICGAN learns an inverse mapping from input images to latent vectors and manipulates attributes by changing the condition for fixed latent vectors. Moreover, PuppetGAN [233] introduced a new approach to training image manipulation systems. In particular, PuppetGAN transforms attributes in images based on examples of how the desired attribute affects the output of a crude simulation (e.g. a 3D model of facial expression). Also, PuppetGAN uses synthetic data to train attribute disentanglement eliminating the need for annotations for the real data, as the disentanglement is extended to the real domain, too.

Along the direction of developing models for facial expression editing without supervision from expression annotations, StyleRig [116] has enabled rig-like control of face generation performed by a pre-trained StyleGAN [2] generator, associating the parameter space of a 3D blendshape model with the latent space of StyleGAN. Generating smooth facial animation by discovering interpretable directions in the latent space of GANs has also been explicitly studied by the works in [234, 235], in which linear and non-linear paths are discovered respectively. Moreover, in [236] the authors propose to discover semantically meaningful attributes (e.g. gender, expression) of generated images by clustering the features of pre-trained generators and learning mappings in latent space between cluster-specific latent codes and the latent space of the pre-trained generators.

Instead of learning a generator, X2Face [230] changes expression and pose from driving images, pose or audio codes, utilising an embedding network and a driving network. It is trained with videos requiring no annotations apart from identity, but can be tested on single source and target frames. Besides, we acknowledge [237] which is a concurrent work, very closely related to ours. In this work, the authors similarly to us employ blendshape parameters for expression editing but follow a different approach in image editing, handling 3D texture (UV maps) and shape separately and composing them in a final output image by rendering. This method produces realistic results in expression manipulation, but involves 3DMM fitting and rendering during testing which can be computationally demanding. Nevertheless, it demonstrates the usefulness of blendshapes in the task of automatic face manipulation.

Sequence manipulation Face reenactment is the process of animating a target face using the face, audio, text, or other codes from a source video to drive the animation. Differently to most of i2i translation methods, face reenactment methods most often require thousands if not millions of frames of the same person for training, testing or both. Targeted to sequence manipulation, Face2Face [110] animates facial expression of the target video, based on rendering a face with the requested expression, warping the texture from the available frames of the target video and then blending. Face2Face, also, does not require training. Deep Video Portraits [114], produces similar results to Face2Face but animates the whole head and is trained for specific source and target videos, meaning that training has to be repeated when the source or target changes. Other methods drive the animation using audio or text as driving codes [229, 113]. Finally, Deferred Neural Rendering (DNR) [117] is based on learning neural textures, feature maps associated with the scene capturing process, employed by a neural renderer to produce the outputs. DNR is trained for specific source and target videos, too.

4.2 Proposed Methodology

In this section, we develop the proposed methodology for continuous facial expression editing based on sliding the parameters of a 3D blendshape model.

4.2.1 Slider-based Generative Adversarial Network for continuous facial expression and speech editing

Problem Definition: Let us here first formulate the problem under analysis and then describe our proposed approach to address it. We define an input image $\mathbf{I}_{org} \in \mathbb{R}^{H\times W\times 3}$ which depicts a human face of arbitrary expression. We further assume that any facial deformation or grimace evident in image $\mathbf{I}_{org}$, can be encoded by a parameter vector $\mathbf{p}_{org} = [p_{org,1}, p_{org,2}, ..., p_{org,N}]^\top$, of N continuous scalar values $p_{org,i}$, normalised in the range $[-1, 1]$. In addition, the same vector $\mathbf{p}_{org}$ constitutes the parameters of a linear 3D blendshape model $\mathbf{S}_{exp}$ that, as in Figure 4.3, instantiate the 3D representation of the facial deformation of image $\mathbf{I}_{org}$ which is given by the expression:

$$\mathcal{S}_{exp}(\mathbf{p}_{org}) = \bar{\mathbf{s}} + \mathbf{U}_{exp}\mathbf{p}_{org}, \tag{4.1}$$

where $\bar{\mathbf{s}}$ is a mean 3D face component and $\mathbf{U}_{exp}$ the expression eigenbasis of the 3D blendshape model. Detailed information on expression modelling using blenshapes, as well as how parameters $\mathbf{p}_{org}$ are extracted from images are included in Section 2.1.

Our goal is to develop a generative model which given an input image $\mathbf{I}_{org}$ and a target expression

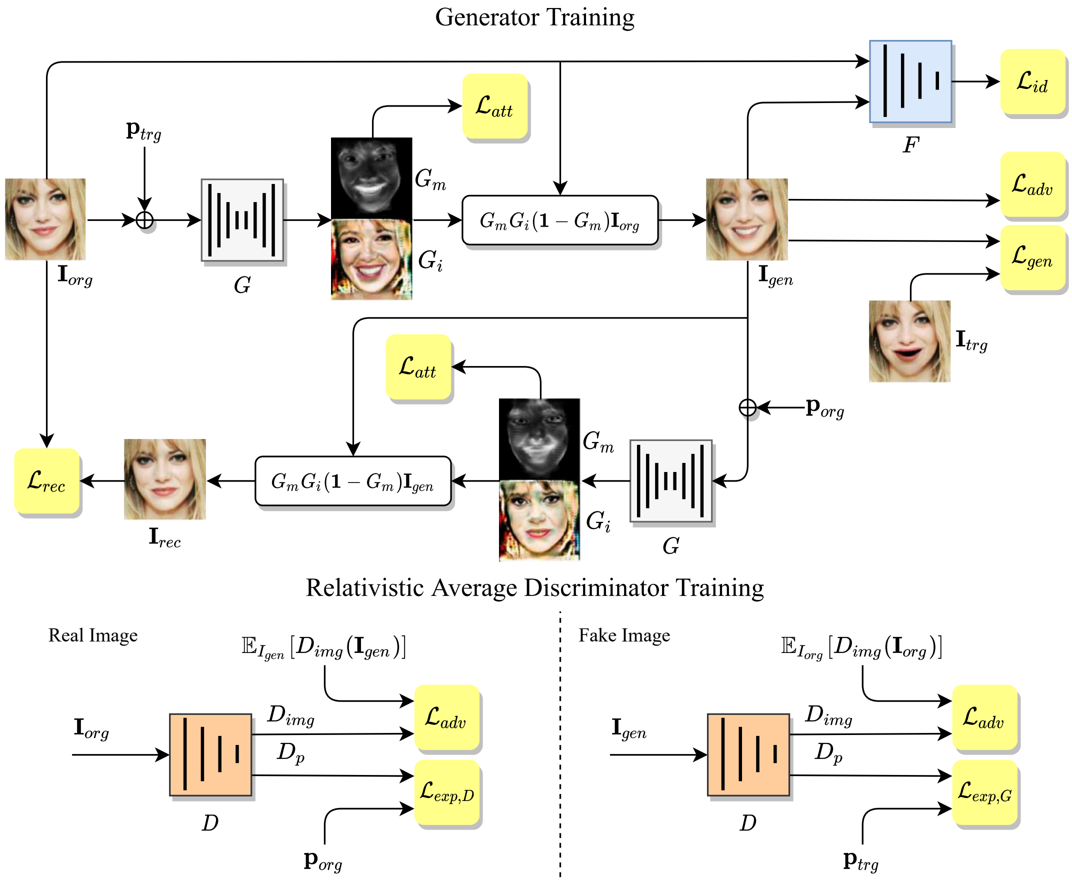


Figure 4.2: Synopsis of the modules, losses and the training process of SliderGAN. An attention-based generator G is trained to generate realistic expressive faces from continuous parameters by employing a set of adversarial, generation, reconstruction, identity and attention losses. The performance of our model is significantly boosted by employing synthetic image pairs through the $\mathcal{L}_{gen}$ loss. Moreover, a relativistic discriminator D is trained to classify images as relatively more real or fake, as well as to regress expression parameters of the input images in order to increase the generation quality of G.

parameter vector $\mathbf{p}_{trg}$, will be able to generate a new version $\mathbf{I}_{gen}$ of the input image with simulated expression given by the 3D expression instance $\mathcal{S}_{exp}(\mathbf{p}_{trg})$.

Attention-Based Generator: To address the above challenging problem, we propose to employ a Generative Adversarial Network architecture in order to train a generator network G that performs translation of an input image $\mathbf{I}_{org}$, conditioned on a vector of 3D blendshape parameters $\mathbf{p}_{trg}$; thus, learning the generator mapping $G(\mathbf{I}_{org}|\mathbf{p}_{trg}) \rightarrow \mathbf{I}_{gen}$. In addition, to better preserve the content and the colour of the original images we employ an attention mechanism at the output of the generator as in [238, 20]. That is we employ a generator with two parallel output layers, one producing a smooth

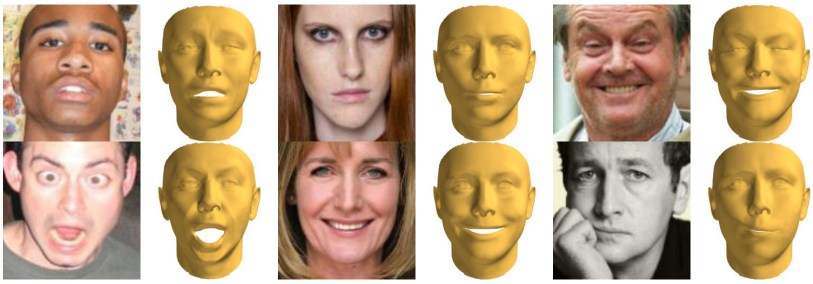


Figure 4.3: Examples of the 3D representation of the expression of an image by the model $\mathcal{S}_{exp}$. The 3D faces of this figure have been generated by 3DMM fitting on the corresponding images.

deformation mask $G_m \in \mathbb{R}^{H\times W}$ and the other a deformation image $G_i \in \mathbb{R}^{H\times W\times 3}$. The values of G_m are restricted in the region $[0, 1]$ by enforcing a sigmoid activation. Then, G_m and G_i are combined with the original image $\mathbf{I}_{org}$ to produce the target expression $\mathbf{I}_{gen}$ as:

$$\mathbf{I}_{gen} = G_m G_i + (1 - G_m)\mathbf{I}_{org}. \tag{4.2}$$

Relativistic Discriminator: We employ a discriminator network D that forces the generator G to produce realistic images of the desired deformation. Different from the standard discriminator in GANimation which estimates the probability of an image being real, we employ the Relativistic Discriminator [35] which estimates the probability of an image being relatively more realistic than a generated one. That is if $D_{img} = \sigma(C(\mathbf{I}_{org}))$ is the activation of the standard discriminator, then $D_{RaD,img} = \sigma(C(\mathbf{I}_{org}) - C(\mathbf{I}_{gen}))$ is the activation of the Relativistic Discriminator. Particularly, we employ the Relativistic average Discriminator (RaD) which accounts for all the real and generated data in a mini-batch. Then, the activation of the RaD is:

$$D_{RaD,img} = \begin{cases} \sigma(C(\mathbf{I}) - \mathbb{E}_{I_{gen}}[C(\mathbf{I}_{gen})]), \text{if } \mathbf{I} \text{ is a real image} \\ \sigma(C(\mathbf{I}) - \mathbb{E}_{I_{org}}[C(\mathbf{I}_{org})]), \text{if } \mathbf{I} \text{ is a generated image} \end{cases} \tag{4.3}$$

where $\mathbb{E}_{I_{org}}$ and $\mathbb{E}_{I_{gen}}$ define the average evaluations of all real and generated images in a mini-batch respectively. In the above definitions σ is the activation function of the discriminator and C is the output of the convolutional structure of the discriminator without activation function. That is, σ is applied on the difference of two terms (which include multiple evaluations of the discriminator) rather than directly on the output of the discriminator.

We further extend D by adding a regression layer parallel to D_{img} that estimates a parameter vector $\mathbf{p}_{est}$, to encourage the generator to produce accurate facial expressions, $D(\mathbf{I}) \rightarrow D_p(\mathbf{I}) = \mathbf{p}_{est}$.

Finally, we aim to boost the ability of G to maintain face identity between the original and the generated images by incorporating a face recognition module F.

Semi-supervised training: We train our model in a semi-supervised manner with both data with no image pairs of the same person under different expressions $\{\mathbf{I}^i_{org}, \mathbf{p}^i_{org}, \mathbf{p}^i_{trg}\}^K_{i=1}$ and data with image pairs that we automatically generate as described in detail in Section 4.3.1, $\{\mathbf{I}^i_{org}, \mathbf{p}^i_{org}, \mathbf{I}^i_{trg}, \mathbf{p}^i_{trg}\}^L_{i=1}$. The supervised part of training essentially supports SliderGAN being robust on errors of expression parameters extracted from 3DMM fitting. Further discussion on the nature and effect of such errors is included in Section 4.3.6. The modules of our model, as well as the training process of SliderGAN are presented in Figure 4.2.

Adversarial Loss: To improve the photorealism of the synthesised images we utilise the Wasserstein GAN adversarial objective with gradient penalty (WGAN-GP) [36]. Therefore, the selected WGAN-GP adversarial objective with RaD is defined as:

$$\begin{aligned}\mathcal{L}_{adv} = &\mathbb{E}_{I_{org}}[D_{RaD,img}(\mathbf{I}_{org})] \\ &- \mathbb{E}_{I_{org},p_{trg}}[D_{RaD,img}(G(\mathbf{I}_{org}, \mathbf{p}_{trg}))] \\ &- \lambda_{gp}\mathbb{E}_{I_{gen}}[(\|\nabla_{I_{org}} D_{RaD,img}(\mathbf{I}_{gen})\|_2 - 1)^2],\end{aligned} \tag{4.4}$$

where the first two terms correspond to the WGAN critic loss, aiming to minimize the distance between the distribution of real and generated images, and the third one to the gradient penalty term encouraging the norm of the gradients to go towards 1. In [36] the authors argue that 1 is a reasonable norm for the gradients as is equal to the norm of the gradients of the optimal WGAN solution, as well as that using the penalty term is preferable to gradient clipping.

Based on 4.3 and different from the standard discriminator, both real and generated images are included in the generator part (the second term) of the objective of 4.4. This allows the generator to benefit from the gradients of both real and fake images, which as we show in the experimental section leads to generated images with sharper edges and more details. This contributes to better representing the distribution of the real data. Based on the original GAN rational [1] and the Relativistic GAN [35], our generator G and discriminator D are involved in a min-max game, where G tries to maximise the objective of (4.4) by generating realistic images to fool the discriminator, while D tries to minimise it by correctly classifying the real images as more realistic than the fake ones and the generated images as less realistic than the real ones.

Expression Loss: To make G consistent in accurately transferring target deformations $\mathcal{S}_{exp}(\mathbf{p}_{trg})$ to the generated images, we consider the discriminator D to have the role of an inspector. To this end, we back-propagate a mean squared loss between the estimated vector $\mathbf{p}_{est}$ of the regression layer of D and the actual vector of expression parameters of an image.

We apply the expression loss both on original images and generated ones. Similarly to the classification loss of StarGAN [19], we construct separate losses for the two cases. For real images $\mathbf{I}_{org}$ we define the loss:

$$\mathcal{L}_{exp,D} = \frac{1}{N}\|D_p(\mathbf{I}_{org}) - \mathbf{p}_{org})\|^2, \tag{4.5}$$

between the estimated and real expression parameters of $\mathbf{I}_{org}$, while for the generated images we define the loss:

$$\mathcal{L}_{exp,G} = \frac{1}{N}\|D_p(G(\mathbf{I}_{org}, \mathbf{p}_{trg})) - \mathbf{p}_{trg})\|^2, \tag{4.6}$$

between the estimated and target expression parameters of $\mathbf{I}_{gen} = G(\mathbf{I}_{org}, \mathbf{p}_{trg})$. Consequently, D minimises $\mathcal{L}_{exp,D}$ to accurately regress the expression parameters of real images, while G minimises $\mathcal{L}_{exp,G}$ to generate images with accurate expression according to D.

Image Reconstruction Loss: The adversarial and the expression losses of (4.4), (4.5) and (4.6), would be enough to generate random realistic expressive images which however, would not preserve the contents of the input image $\mathbf{I}_{org}$. To overcome this limitation we admit a cycle consistency loss [18] for our generator G:

$$\mathcal{L}_{rec} = \frac{1}{W \times H}\|\mathbf{I}_{org} - \mathbf{I}_{rec}\|_1, \tag{4.7}$$

over the vectorised forms of the real image $\mathbf{I}_{org}$ and the reconstructed one $\mathbf{I}_{rec} = G(G(\mathbf{I}_{org}, \mathbf{p}_{trg}), \mathbf{p}_{org})$. Note that we obtain image $\mathbf{I}_{rec}$ by using the generator twice, first to generate image $\mathbf{I}_{gen} = G(\mathbf{I}_{org}, \mathbf{p}_{trg})$ and then to get the reconstructed $\mathbf{I}_{rec} = G(\mathbf{I}_{gen}, \mathbf{p}_{org})$, conditioning $\mathbf{I}_{gen}$ on the parameters $\mathbf{p}_{org}$ of the original image.

Image Generation Loss To further boost our generator towards accurately editing expression based on a vector of parameters, we introduce image pairs of the form $\{\mathbf{I}^i_{org}, \mathbf{p}^i_{org}, \mathbf{I}^i_{trg}, \mathbf{p}^i_{trg}\}^L_{i=1}$ that we automatically generate from neutral images as described in detail in Section 4.3.1. We exploit the synthetic pairs of images of the same individuals under different expression by introducing an image generation loss:

$$\mathcal{L}_{gen} = \frac{1}{W \times H} \|\mathbf{I}_{trg} - \mathbf{I}_{gen}\|_1, \tag{4.8}$$

where $\mathbf{I}_{trg}$ and $\mathbf{I}_{gen}$ are images with either neutral or synthetic expression of the same individual. Here, we calculate the $L1$ loss between the synthetic ground truth image $\mathbf{I}_{trg}$ and the generated by G, $\mathbf{I}_{gen}$, aiming to boost our generator to accurately transfer the 3D expression $\mathcal{S}_{exp}(\mathbf{p}_{trg})$ to the edited image.

Identity Loss: Image reconstruction loss of (4.7), aids to maintain the surroundings between the original and generated images. However, the faces' identity is not always maintained by this loss, as also show by our ablation study in Section 4.3.9. To alleviate this issue, we introduce a face recognition loss adopted from ArcFace [239], which models face recognition confidence by an angular distance loss. Particularly, we introduce the loss:

$$\mathcal{L}_{id} = 1 - \cos(\mathbf{e}_{gen}, \mathbf{e}_{org}) = 1 - \frac{\|\mathbf{e}_{gen}\| \|\mathbf{e}_{org}\|}{\mathbf{e}_{gen}^{\top} \mathbf{e}_{org}}, \tag{4.9}$$

where $\mathbf{e}_{gen} = F(\mathbf{I}_{gen})$ and $\mathbf{e}_{org} = F(\mathbf{I}_{org})$ are embeddings of $\mathbf{I}_{gen}$ and $\mathbf{I}_{org}$ respectively, extracted by the face recognition module F. According to ArcFace, face verification confidence is higher as the cosine distance $\cos(\mathbf{e}_{gen}, \mathbf{e}_{org})$ grows. During training, G is optimised to maintain face identity between $\mathbf{I}_{gen}$ and $\mathbf{I}_{org}$ which minimises (4.9).

Attention Mask Loss: To encourage the generator to produce sparse attention masks G_m that focus on the deformation regions and do not saturate to 1, we employ a sparsity loss $\mathcal{L}_{att}$. That is, we calculate and minimise the $L1$-norm of the produced masks for both the generated and the reconstructed images, defining the loss as:

$$\mathcal{L}_{att} = \frac{1}{W \times H} \Big(\|G_m(\mathbf{I}_{org}, \mathbf{p}_{trg})\|_1 + \|G_m(\mathbf{I}_{gen}, \mathbf{p}_{org})\|_1 \Big), \tag{4.10}$$

Total Training Loss: We combine losses (4.4) - (4.10) to form loss functions $\mathcal{L}_G$ and $\mathcal{L}_D$ for separately training the generator G and the discriminator D of our model. We formulate the loss functions as:

$$\mathcal{L}_G = \begin{cases} \mathcal{L}_{adv} + \lambda_{exp}\mathcal{L}_{exp,G} + \lambda_{rec}\mathcal{L}_{rec} + \lambda_{id}\mathcal{L}_{id} + \lambda_{att}\mathcal{L}_{att}, \\ \qquad \text{for unpaired data } \{\mathbf{I}_{org}^i, \mathbf{p}_{org}^i, \mathbf{p}_{trg}^i\}_{i=1}^K \\ \mathcal{L}_{adv} + \lambda_{exp}\mathcal{L}_{exp,G} + \lambda_{rec}\mathcal{L}_{rec} + \lambda_{gen}\mathcal{L}_{gen} + \lambda_{id}\mathcal{L}_{id} + \lambda_{att}\mathcal{L}_{att}, \\ \qquad \text{for paired data } \{\mathbf{I}_{org}^i, \mathbf{p}_{org}^i, \mathbf{I}_{trg}^i, \mathbf{p}_{trg}^i\}_{i=1}^L \end{cases} \tag{4.11}$$

$$\mathcal{L}_D = -\mathcal{L}_{adv} + \lambda_{exp}\mathcal{L}_{exp,D}, \tag{4.12}$$

where λ_{exp}, λ_{rec}, λ_{gen}, λ_{id} and λ_{att} are parameters that regularise the importance of each term in the total loss function. We discuss the choice of those parameters in Section 4.3.2.

As can be noticed in (4.11), we employ different loss functions $\mathcal{L}_G$, depending on if the training data are the real data with no image pairs or the synthetic data which include pairs. The only difference is that in the case of paired data we use the additional supervised loss term $\mathcal{L}_{gen}$.

4.2.2 Implementation details

Having presented the architecture of our model, here we report further implementation details. For the generator module G of SliderGAN, we adopted the architecture of CycleGAN [18] as it is proved to generate remarkable results in image-to-iamge translation problems, as for example in StarGAN [19]. We extended the generator by adding a parallel output layer to accomodate the attention mask mechanism. Moreover, for D we adopted the architecture of PatchGAN [17] which produces probability distributions of the multiple image patches to be real or generated, $D(\mathbf{I}) \rightarrow D_{img}$. As described in Section 4.2.1, we extended this discriminator architecture by adding a parallel regression layer to estimate continuous expression parameters.

4.3 Experiments

In this section, we present a series of experiments that we conducted in order to evaluate the performance of SliderGAN. First, we describe the datasets we utilised to train and test our model (Section 4.3.1) and provide details on the training setting for each experiment (Section 4.3.2)). Then, we test the ability of SliderGAN to manipulate the expression in images by adjusting a single or multiple parameters of a 3D blendshape model (Section 4.3.3). Moreover, we present our results in direct expression transfer between an input and a target image (Section 4.3.4), as well as in discrete expression synthesis (Section 4.3.5). Next, we test Ganimation on expression editing when trained with blendshape vectors instead of AUs (Section 4.3.6). We examine the ability of SliderGAN to handle face deformations due to speech (Section 4.3.7) and test the regression accuracy of our model's discriminator (Section 4.3.8). We close the experimental section of our work by presenting an ablation study

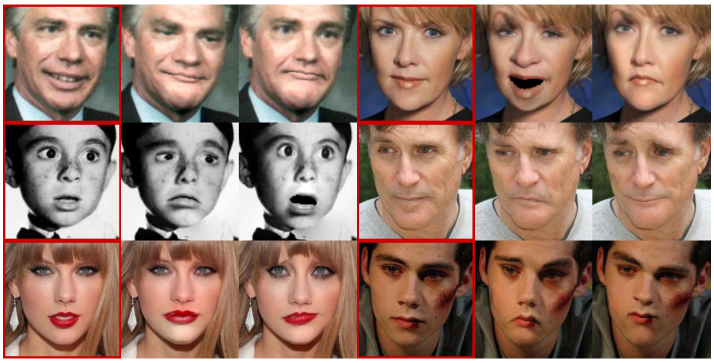


Figure 4.4: Synthetic expressive faces, generated by fitting a 3DMM on the original images and rendering back with a randomly sampled expression. The images with a red frame are the original images.

on the contribution of the different loss functions (Section 4.3.9) and a discussion on limitations and failure cases of our technique (Section 4.3.10).

4.3.1 Datasets

Emotionet: For the training and validation phases of our algorithm we utilised a subset of 250,000 images of the EmotioNet database [240], which contains over 1 million images of expression and emotion, accompanied by annotations about facial Action Units. However, SliderGAN is trained with image - blendshape parameters pairs which are not available. Therefore, in order to extract the expression parameters we fit the 3DMM of [13] on each image of the dataset in use. To ensure the high quality of 3D reconstruction, we employed the LSFM [70] identity model concatenated with the expression model of 4DFAB [59]. The 4DFAB expression model was built from a collection of over 10,000 expressive face 3D scans of spontaneous and posed expressions, collected from 180 individuals in 4 sessions over the period of 5 years. SliderGAN exploits the scale and representation power of 4DFAB to learn how to realistically edit facial expressions in images. The method described above constitutes a technique to automatically annotate the dataset and eliminates the need of costly manual annotation.

3D Warped Images: One crucial problem of training with pseudo-annotations extracted by 3DMM fitting on images, is that the parameter values are not always consistent as small variations in expression can be mistakenly explained by the identity, texture or camera model of the 3DMM. To overcome this limitation, we augmented the training dataset with expressive images that we render and therefore know the exact blendshape parameter values. In more detail, we fit with the same 3DMM 10,000 images of EmotioNet in order to recover the identity and camera models for each image. A 3D texture can also be sampled by projecting the recovered mesh on the original image. Then, we combined the identity meshes with randomly generated expressions from the 4DFAB expression model and rendered back on the original images. Rendering 20 different expressions from each image, we augmented the dataset by 200,000 accurately annotated images. Some of the generated images are displayed in Figure 4.4

4DFAB Images: A common problem of developing generative models of facial expression is the difficulty in accurately measuring the quality of the generated images. This is mainly due to the lack of databases with images of people of the same identity with arbitrary expressions. To overcome this issue and quantitatively measure the quality of images generated by SliderGAN, as well as compare with the baseline, we created a database with rendered images from 3D meshes and textures of 4DFAB. In more detail, we rendered 100 to 500 images with arbitrary expression from each of the 180 identities and for each of the 4 sessions of 4DFAB, thus rendering 300,000 images in total. To obtain expression parameters for each rendered image, we projected the blendshape model $\mathcal{S}_{exp}$ on each corresponding 3D mesh $\mathbf{S}$ such that the obtained parameters are $\mathbf{p} = \mathbf{U}_{exp}^{\top}(\mathbf{S} - \bar{\mathbf{s}})$.

Lip Reading Words in 3D (LRW-3D): Lip Reading in the Wild (LRW) dataset [241] consists of videos of hundreds of speakers including up to 1000 utterances of 500 different words. LRW-3D [34] provides speech blendshapes parameters for the frames of LRW, which were recovered by mapping each frame of LRW that corresponds to one of the 500 words to instances of a 3D blendshape model of speech. This was achieved by aligning the audio segments of the LRW videos and those of a 4D speech database. Moreover, to extract expression parameters for each word segment of the videos we applied the 3DMM video fitting algorithm of [13], which accounts for the temporal dependency between frames. In Section 4.3.7, we utilise the annotations of LRW-3D as well as the expression parameters to perform expression and speech transfer.

4.3.2 Training Details

In all experiments, we trained our models with images of size 128 × 128 pixels, aligned to a reference shape of 2D landmarks. As condition vectors we utilised the 30 most significant expression components of 4DFAB and the 10 most significant speech components of LRW-3D [34]. The later where only used for the combined expression and speech synthesis experiments. We set the batch size to 16 and trained our models for 60 epochs with Adam [242] ($\beta_1 = 0.5, \beta_2 = 0.999$). Moreover, we chose loss weights $\lambda_{adv} = 30$, $\lambda_{exp} = 1000$, $\lambda_{rec} = 10$, $\lambda_{gen} = 10$, $\lambda_{id} = 4$ and $\lambda_{att} = 0.3$. Larger values for λ_{id} significantly restrict G, driving it to generate images very close to the original ones with no change in expression. Also, lower values for λ_{att}, lead to mask saturation.

In all our experiments training was performed in two phases over a total of 60 epochs. Particularly, we first trained our models for 20 epochs, utilising only the generated image pairs of the ”3D warped images” database presented in Section 4.3.1. This training phase makes our models robust to parameter errors as further discussed in Section 4.3.6. Then, we proceeded to unsupervised training for another 40 epochs with a dataset of unpaired real images, which we selected depending on the task. In this training phase, our models learn to generate the realistic details related to expression and speech. For speech synthesis, we train the model from the beginning with an extended parameter vector of 40 elements, setting the speech parameters to zero for the first phase of training where we train only for expression.

In more detail, the datasets we employed for the second phase of training in our experiments are as follows. We employed:

- **EmotioNet** for our experiments on:
  - 3D model-based expression editing (Section 4.3.3),
  - expression transfer and interpolation on images of Emotionet (Section 4.3.4),
  - discrete expression synthesis (Section 4.3.5),
  - comparing with Ganimation conditioned on blendshape parameters (Section 4.3.6),
  - 3d expression reconstruction (Section 4.3.8),
  - the ablation study (Section 4.3.9),
  - limitations of our model (Section 4.3.10),
- **4DFAB Images** for the experiment on expression transfer and interpolation on images of 4DFAB (Section 4.3.4),
- **LRW-3D** for the combined expression and speech synthesis experiment (Section 4.3.7).

4.3.3 3D Model-based Expression Editing

Sliding single expression parameters: In this experiment, we demonstrate the capability of SliderGAN to edit the facial expression of images when single expression parameters are slid within the normalised range [-1, 1]. In Figure 4.5 we provide results for 10 levels of activation of single parameters of the model (-1, -0.8, -0.6, -0.4, -0.2, 0, 0.2, 0.4, 0.6, 0.8, 1), while the rest parameters remain zero. As can be observed in Figure 4.5, SliderGAN successfully learns to reproduce the behaviour of each blendshape separately, producing realistic facial expressions while adequately maintaining the identity of the input image. Also, the transition between the generated expressions is smooth for successive values of the same parameter and the intensity of the expressions dependent on the magnitude of the parameter value. Note that when the zero vector is applied, SliderGAN produces the neutral expression, whatever the expression of the original image.

Sliding multiple expression parameters: The main feature of SliderGAN is its ability to edit facial expressions in images by sliding multiple parameters of the model, similarly to sliding parameters in a blendshape model to generate new expressions of a 3D face mesh. To test this characteristic of our model, we synthesise random expressions by conditioning the generator input on parameter vectors with elements randomly drawn from the standard normal distribution. Note that the model was trained with expression parameters normalised by the square root of the eigenvalues $e_i, i = 1, ..., N$ of the PCA blendshape model. This means that all combinations of expression parameters within the range [-1, 1] correspond to feasible facial expressions.

As illustrated by Figure 4.6, SliderGAN is able to synthesise face images with a great variability of expressions, while adequately maintaining identity. The generated expressions accurately resemble the 3D meshes' expressions when the same vector of parameters is used for the blendshape model. This fact makes our model ideal for facial expression editing in images. A target expression can first be chosen by utilising the ease of perception of 3D visualisation of a 3D blendshape model and then, the target parameters can be employed by the generator to edit a face image accordingly.

4.3.4 Expression Transfer and Interpolation

A by-product of SliderGAN is that the discriminator D learns to map images to expression parameters $D_{\mathbf{p}}$ that represent their 3D expression through $\mathcal{S}_{exp}(D_{\mathbf{p}})$. We capitalise on this fact to perform

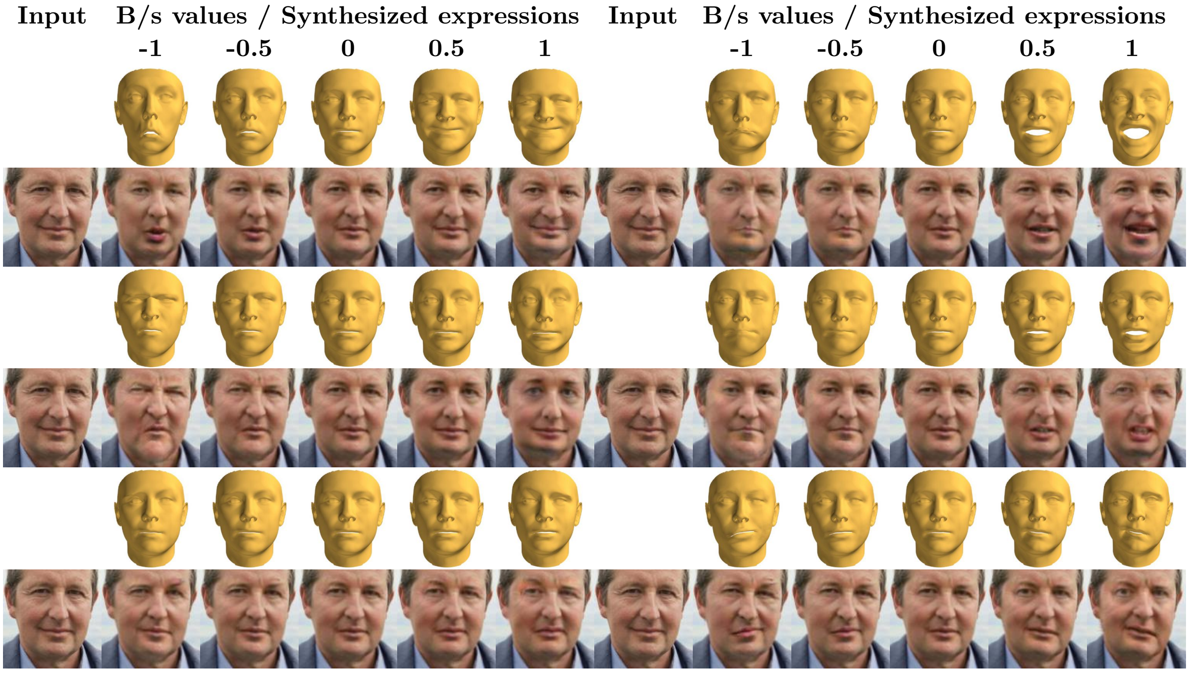

Input
B/s values / Synthesized expressions
-1
-0.5
0
0.5
1
Input
B/s values / Synthesized expressions
-1
-0.5
0
0.5
1

Figure 4.5: Expressive faces generated by sliding single blendshape (b/s) parameters in the range $[-1, 1]$. As it is observed, the edited images accurately replicate the 3D faces' motion in the whole range of parameter values.

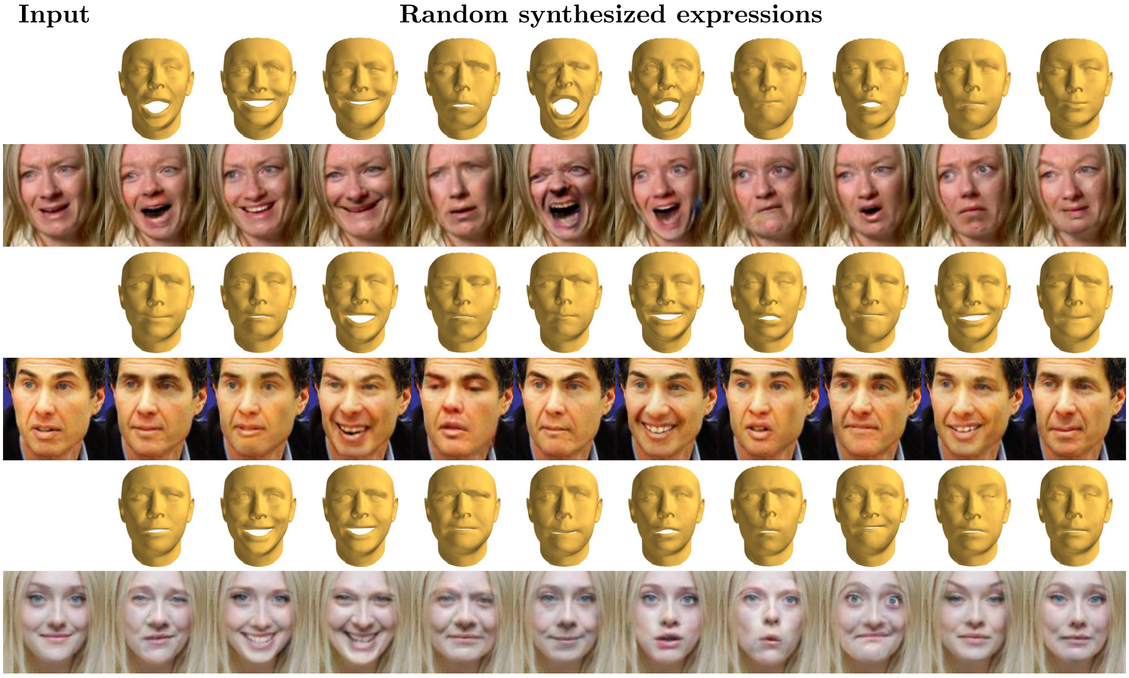


Figure 4.6: Expressive faces generated by sliding multiple blendshape (b/s) parameters in the range $[-1, 1]$. As it is observed, the wide range of the edited images accurately replicate the 3D faces' motion.

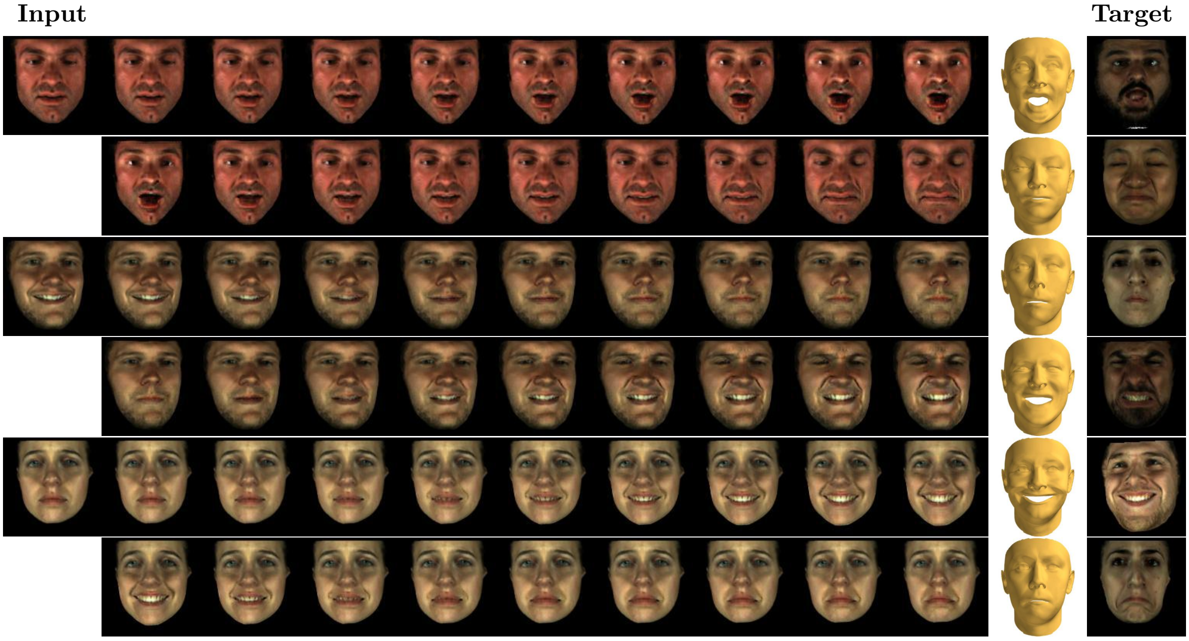


Figure 4.7: Expression interpolation between images of 4DFAB. First, we employ D to recover the expression parameters from an input and the target images. Then, we capitalise on these parameter vectors to animate the expression of the input image towards multiple targets.

direct expression transfer and interpolation between images without any annotations about expression. Assuming a source image $\mathbf{I}_{src}$ with expression parameters $\mathbf{p}_{src} = D_{\mathbf{p}}(\mathbf{I}_{src})$ and a target image $\mathbf{I}_{trg}$ with expression parameters $\mathbf{p}_{trg} = D_{\mathbf{p}}(\mathbf{I}_{trg})$, we are able to transfer expression $\mathbf{p}_{trg}$ to image $\mathbf{I}_{src}$ by utilising the generator of SliderGAN, such that $\mathbf{I}_{src \rightarrow trg} = G(\mathbf{I}_{src}|\mathbf{p}_{trg})$. Note that no 3DMM fitting or manual annotation is required to extract the expression parameters and transfer the expression, as this is performed by the trained discriminator.

Additionally, by interpolating the expression parameters of the source and target images, we are able to generate expressive faces that demonstrate a smooth transition from expression $\mathbf{p}_{src}$ to expression $\mathbf{p}_{trg}$. Interpolation of the expression parameters can be performed by sliding an interpolation factor a within the region [0,1] such that the requested parameters are $\mathbf{p}_{interp} = a\mathbf{p}_{src} + (1-a)\mathbf{p}_{trg}$.

Qualitative Evaluation: Results of performing expression transfer and interpolation on images of the 4DFAB rendered database and Emotionet are displayed in Figure 4.7 and Figure 4.8 respectively, where it can be seen that the expressions of the generated images obviously reproduce the target expressions. The smooth transitions between expressions $\mathbf{p}_{src}$ and $\mathbf{p}_{trg}$ indicate that SliderGAN successfully

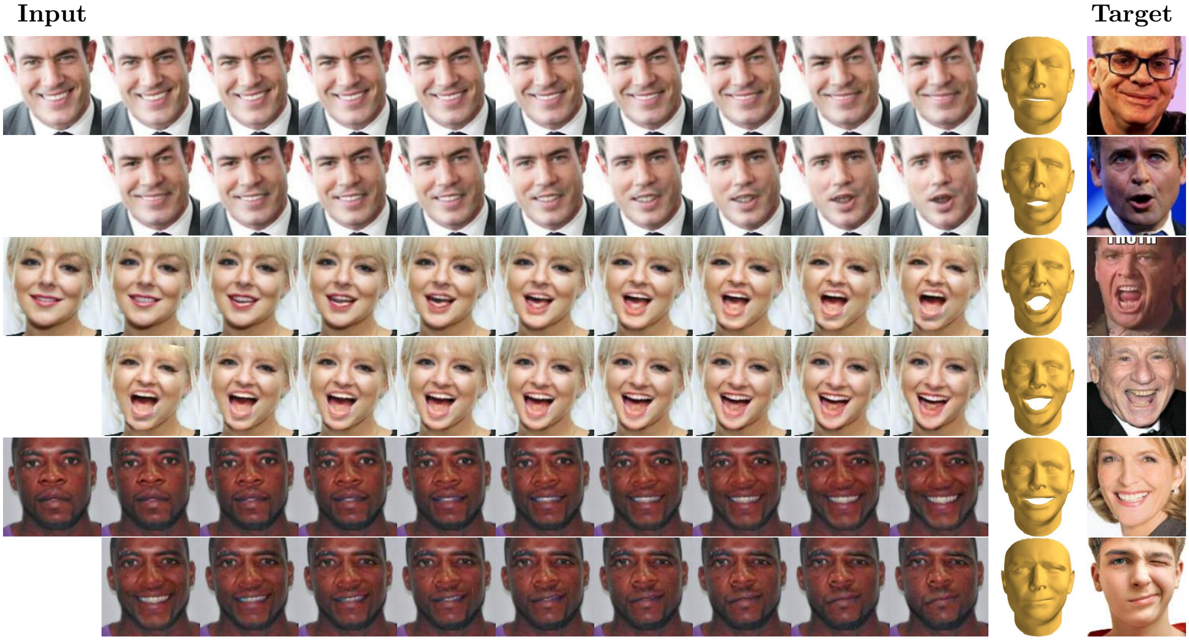


Figure 4.8: Expression interpolation between images of Emotionet. First, we employ D to recover the expression parameters from an input and the target images. Then, we capitalise on these parameter vectors to animate the expression of the input image towards multiple targets.

learns to map images to expressions across the whole expression parameter space. Also, it is evident that D accurately regresses the blendshape parameters from images $\mathbf{I}_{trg}$ by observing the recovered 3D faces. The accuracy of the regressed parameters is also examined in Section 4.3.8.

To further validate the quality of our results, we trained GANimation on the same dataset with AU annotations extracted with OpenFace [85] as suggesed by the authors. We performed expression transfer between images and present results for SliderGAN-RaD, SliderGAN-WGP and GANimation. In Figure 4.9, it is obvious that SliderGAN-RaD benefits from the Relativistic GAN training and produces higher quality textures than SliderGAN-WGP, while both SliderGAN implementations better simulate the expressions of the target images than GANimation. In particular, details such as the eyes, teeth, the inside of the mouth and wrinkles are better defined by SliderGAN-RaD while with SliderGAN-WGP such details are more blurry (for example, the mouth of the the 10th generated image of the 2nd input subject) and the generated images include more artifacts (for example, the mouth of the the 8th generated image of the 2nd input subject). In comparison to both SliderGAN models, GANimation produces images with more blurry details (for example, the inside of the mouth in the 10th generated images of the 1st input subject) and less accurate expressions (for example, the 4th generated image of

Table 4.1: Image Euclidean Distance (IED), calculated between ground truth images of 4DFAB and corresponding generated images by Ganimation [20], SliderGAN-WGP and SliderGAN-RaD. Results from SliderGAN-RaD produce the lowest IED between the three methods.

Method	IED
GANimation [20]	$1.04e - 02$
SliderGAN-WGP	$7.932 - 03$
SliderGAN-RaD	$\mathbf{6.84e - 03}$

the 4th input subject).

Quantitative Evaluation: In this section, we provide quantitative evaluation on the performance of SliderGAN on arbitrary expression transfer. We employ the 4DFAB rendered images dataset which allows us to calculate the Image Euclidean Distance (IED) [243] between ground truth rendered images of 4DFAB and images generated by SliderGAN. Image Euclidean Distance is a robust alternative metric to the standard pixel loss for image distances, which is defined between two RGB images x and y each with $M \times N$ pipxels as:

$$\frac{1}{2\pi}\sum_{i=1}^{MN}\sum_{j=1}^{MN} \exp\{|P_i - P_j|^2/2\}(\|x_i - y_i\|^2)(\|x_j - y_j\|^2) \tag{4.13}$$

where P_i and P_j are the pixel locations on the 2D image plane and x_i, y_i, x_j, y_j the RGB values of images x and y at the vectorised locations i and j.

We trained SliderGAN with the rendered images from 150 identities of 4DFAB, leaving 30 identities for testing. To allow direct comparison between generated and real images, we randomly created 10,000 pairs of images of the same session and identity (this ensures that the images were rendered with the same camera conditions) from the testing set and performed expression transfer within each pair. To compare our model against the baseline model GANimation, we trained and performed the same experiment using GANimation on the same dataset with AUs activations that we obtained with OpenFace. Also, to showcase the benefits of the relativistic discriminator in image quality of the generated images, we repeated the experiment with SliderGAN-WGP. The results are presented in Table 4.1 where it can be seen that SliderGAN-RaD produces images with the lowest IED.

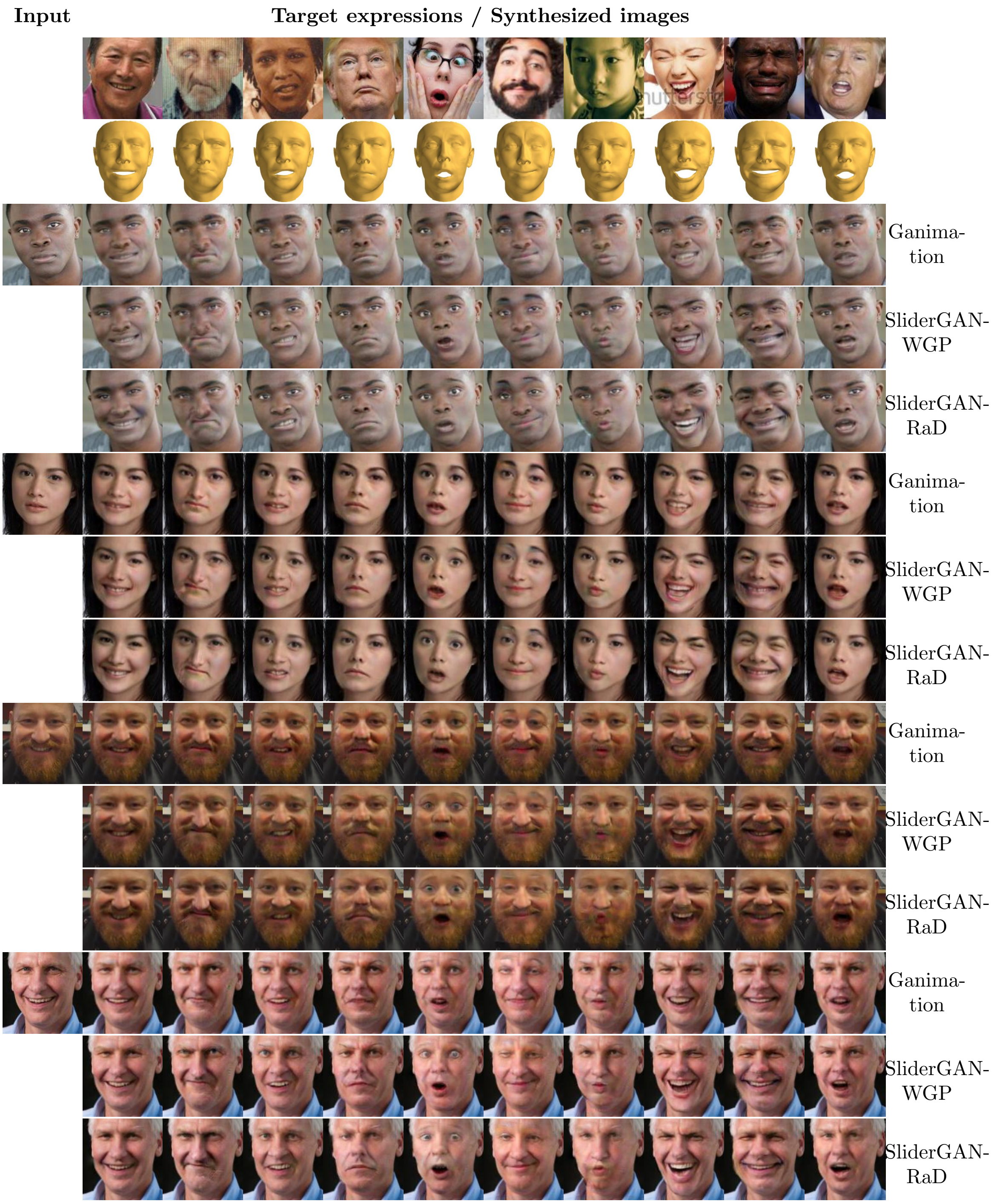


Figure 4.9: Expression transfer between images of Emotionet. First, we employ D to recover expression parameters from the target images. Then, we utilise these parameter vectors to transfer the target expressions to the input images. From the results, SliderGAN-RaD produces higher quality textures than any of the other two methods (mostly evident in the mouth and eyes regions). Moreover, GANimation reproduces the target expressions with lower accuracy. (Please, zoom in the images to notice the differences in texture quality.)

4.3.5 Synthesis of Discrete Expressions

Specific combinations of the 3D expression model parameters represent the discrete expressions anger, contempt, fear, disgust, happiness, sadness, surprise and neutral. To directly translate input images into these expressions, we need appropriate blendshape parameter vectors which reproduce the corresponding 3D model instances. Of course, as our condition vectors consist of real numbers, there do not exist unique 3D instances for each expression, but infinitely many with varying intensity.

To extract such parameter vectors we adopted the following approach. First, we manually picked 10 images for each category of the questioned expressions from EmotioNet. Then, we employed D to estimate parameter vectors for each image, similarly to the expression transfer of Section 4.3.4. We computed the mean vectors for each of the 7 expressions and manually adjusted the values trough visual inspection of the 3D model instances, to create 3D faces that depict the expressions in average intensity (removing any exaggeration or mistakes from the discriminator).

We employ these parameter vectors to synthesise expressive face images of the aforementioned discrete expressions and test our results both qualitatively and quantitatively.

Qualitative Evaluation: To evaluate the performance of SliderGAN in this task, we visually compare our results against the results of five baseline models: DIAT [232], CycleGAN [18], ICGAN [99], StarGAN [19] and GANimation [20]. In Figure 4.10 it is evident that SliderGAN generates results that resemble the queried expressions while maintaining the original face's identity and resolution. The results are close to those of GANimation, however the Relativistic GAN training of SliderGAN allows for slightly higher quality of images.

The neutral expression can also be synthesised by SliderGAN when all the elements of the target parameter vector are set to 0. In fact, the neutral expression of the 3D blendshape model is also synthesised by the same vector. Results of image neutralisation on “in-the-wild” images of arbitrary expression are presented in Figure 4.11, where it can be observed that the neutral expression is generated without significant loss in faces' identity.

Quantitative Evaluation: We further evaluate the quality of the generated expressions by performing expression recognition with the off-the-self recognition system [244]. In more detail, we randomly selected 10,000 images from the test set of Emotionet, translated them to each of the discrete expressions

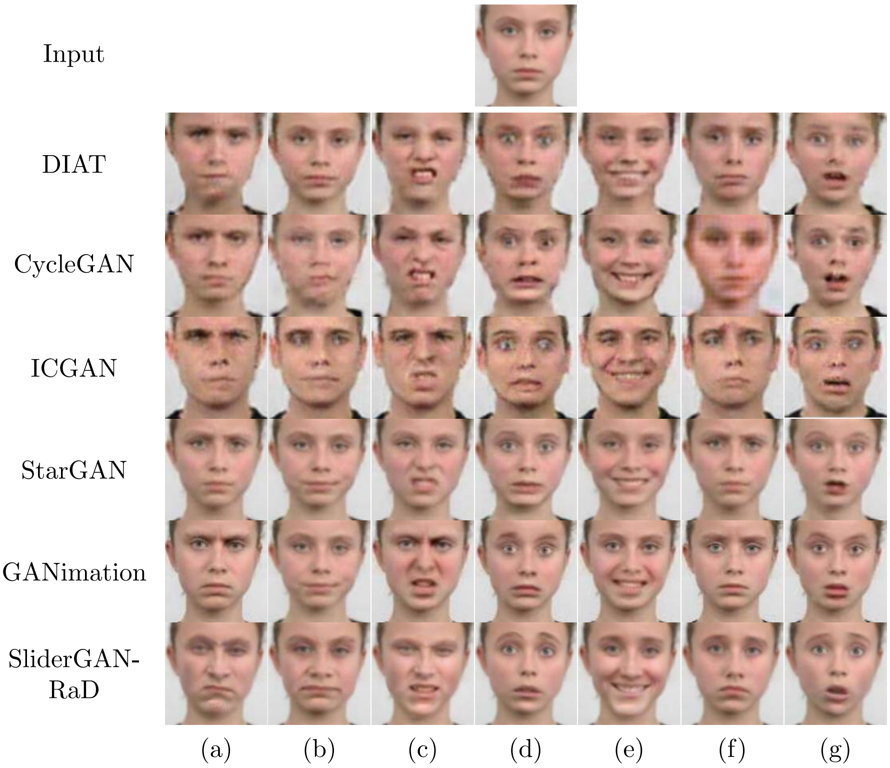


Figure 4.10: Generation of the 7 discrete expressions a) anger, b) contempt, c) disgust, d) fear, e) happiness, f) sadness, g) surprise. By comparing SliderGAN against DIAT [232], CycleGAN [18], ICGAN [99], StarGAN [19] and GANimation [20] we observe that our model generates results of high texture quality that resemble the queried expressions. The results of the rest of the methods where taken from [20].

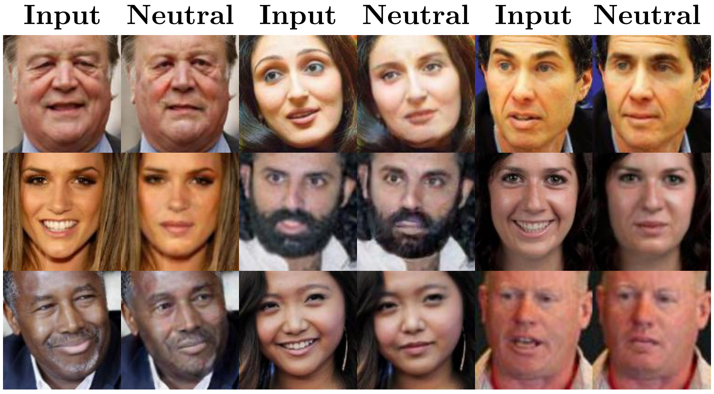


Figure 4.11: Neutralisation of "in-the-wild" images of arbitrary expression. The neutralisation takes place by setting all blendshape parameter values to zero.

Table 4.2: Expression recognition results by applying the off-the-self expression recognition system [244] of images generated by GANimation [20], SliderGAN-WGP and SliderGAN-RaD. Accuracy scores from both SliderGAN models outperform those of GANimation, while SliderGAN-RaD achieves thehighest accuracy in all epressions.

Method	Anger	Disgust	Fear	Happiness	Sadness	Surprise	Neutral	Average
GANimation [20]	0.552	0.446	0.517	0.658	0.632	0.622	0.631	0.579
SliderGAN-WGP	0.550	0.463	0.514	0.762	0.633	0.678	0.702	0.614
SliderGAN-RaD	**0.591**	**0.481**	**0.531**	**0.798**	**0.654**	**0.689**	**0.708**	**0.636**

anger, disgust, fear, happiness, sadness, surprise, neutral and passed them to the expression recognition network. For comparison, we repeated the same experiment with SliderGAN-WGP and GANimation using the same image set. In Table 4.2 we report accuracy scores for each expression class separately, as well as the average accuracy score for the three methods. The classification results are similar for the three models, with both implementations of SliderGAN producing slightly higher scores, which denotes that GANimation's results include more failure cases.

4.3.6 Comparison with Ganimation conditioned on blendshape parameters

It would be reasonable to be assumed that by just substituting AUs with blendshapes, Ganimation could be used to manipulate images based on blendshape conditions. However, this is not the case because Ganimation cannot handle errors of the expression parameters.

3DMM fitting, being an inverse graphics approach to 3D reconstruction, often produces errors related to mistakenly explaining identity and pose of faces as expression and the opposite. For example, a face with a long chin in a slightly side pose might be partially explained by a 3DMM fitting algorithm as a slightly open and shifted mouth or some other similar expression. This is the case for 3D mesh projection (as in the case of recovering parameters from the 4DFAB meshes), too, with which identity can be mistakenly reconstructed to an extend by the linear 3D expression model. This makes the extracted expression parameters to be associated with more attributes of images than only expression.

In the setting of Ganimation, these errors have a negative impact on the robustness and generalisation ability of the model. Particularly, the discriminator becomes dependent on more facial attributes than just expression in regressing the 3DMM parameters. This motivates the generator to reproduce the

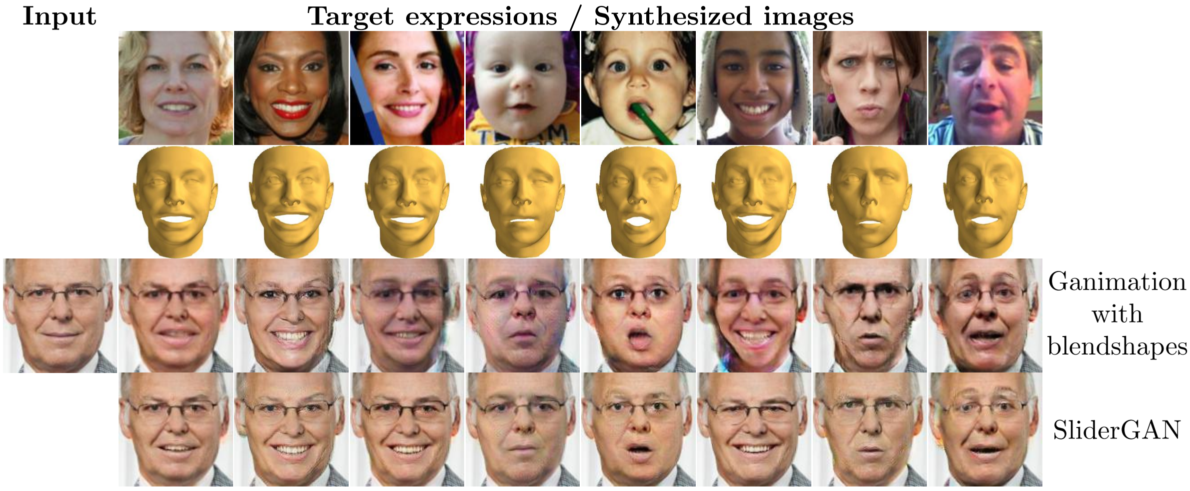

Input
Target expressions / Synthesized images
Ganimation with blendshapes
SliderGAN

Figure 4.12: Evaluation of Ganimation when switching AUs with blenddshapes. Ganimation is not able to handle the errors in expression parameters extracted from 3DMM fitting. The synthesised data, as well as the additional identity loss enables SliderGAN to better translate input images to target expressions.

identity, pose and style of the training images rather than only the target expression, as the two modules compete in the min-max optimisation problem of the GAN.

This problem is handled in SliderGAN by two of the main contributions of our work. First, the 3D warped images used for the 20 first epochs of the training, help the generator produce expressions consistent with the expression blendshapes, even though realistic texture deformations are missing at this stae (e.g wrinkles when smiling). Second, the face recognition error $\mathcal{L}_{id}$ substantially supports retaining the identity between input and generated images, making SliderGAN robust to the errors of 3DMM fitting. The contribution of both losses in training is further examined in Section 4.3.9. As it can be seen in Figure 4.12, the results produced by Ganimation include significant artifacts which are directly related to the identity pose and style of the target images. Contrarily, images generated from SliderGAN do not present such artifacts in most cases and when such artifacts are visible they exist to a considerably lesser extent.

4.3.7 Combined Expression and Speech Synthesis and Transfer

Blendshape coding of facial deformations allows modelling arbitrary deformations (e.g. deformations due to identity, speech, non-human face morphing etc.) which are not limited to facial expressions, unlike AUs coding which is a system that taxonomises the human facial muscles [79]. Even though

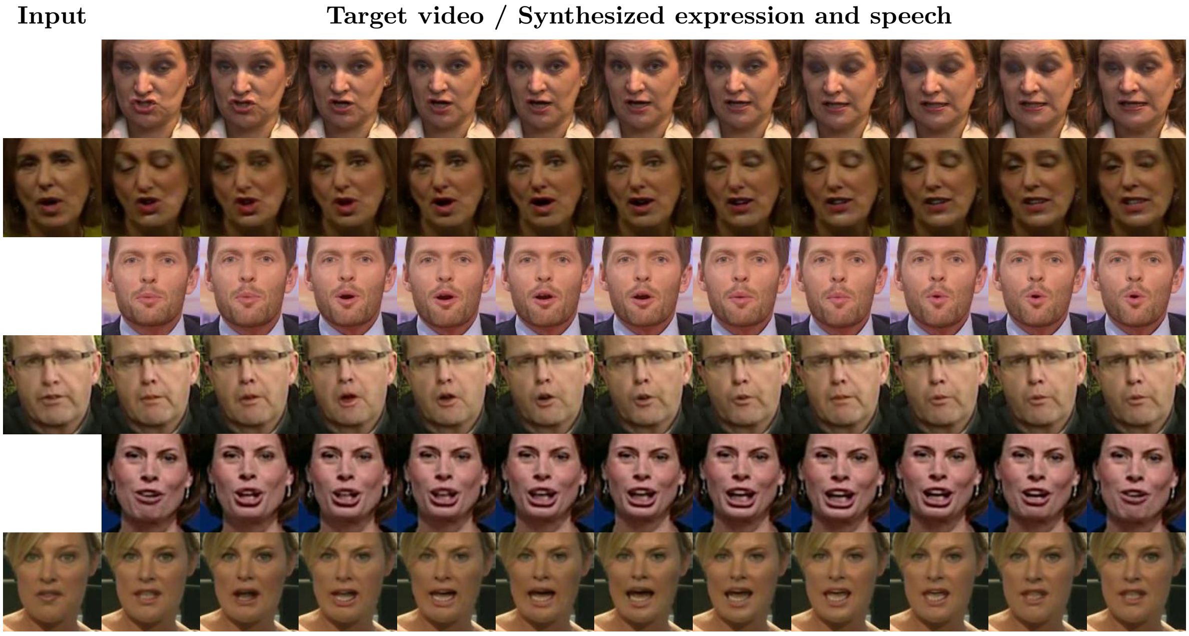


Figure 4.13: Combined expression and speech animation from a single input image. We utilise as targets the expression and speech blendshape parameters of consecutive frames of videos of LRW, to synthesise sequences of expression and speech from a single input image.

AUs 10-28 model mouth and lip motion, not all the details of lip motion that takes place during speech can be captured by these AUs. Moreover, only 10 (AUs 10, 12, 14, 15, 17, 20, 23, 25, 26 and 28) out of these 18 AUs can automatically be recognised, which is achieved only with low accuracy. On the contrary, a blendshape model of the 3D motion of the human mouth and lips would better capture motion during speech, while it would allow the recovery of robust representations from images and videos of human speech.

We capitalise on this fact and employ the mouth and lips blendshape model of [34], $\mathcal{S}_{speech}(\mathbf{q}) = \bar{\mathbf{s}} + \mathbf{U}_{speech}\mathbf{q}$, to perform speech synthesis from a single image with SliderGAN. Particularly, we employ the LRW-3D database which contains speech blendshape parameters annotations for the 500 words of LRW [241], to perform combined expression and speech synthesis and transfer, which we evaluate both qualitatively and quantitatively.

LRW contains videos with both expression and speech. Thus, to completely capture the smooth face motion across frames we employed for each frame 30 expression parameters recovered by 3DMM fitting and 10 speech parameters of LRW-3D which correspond to the 10 most significant components

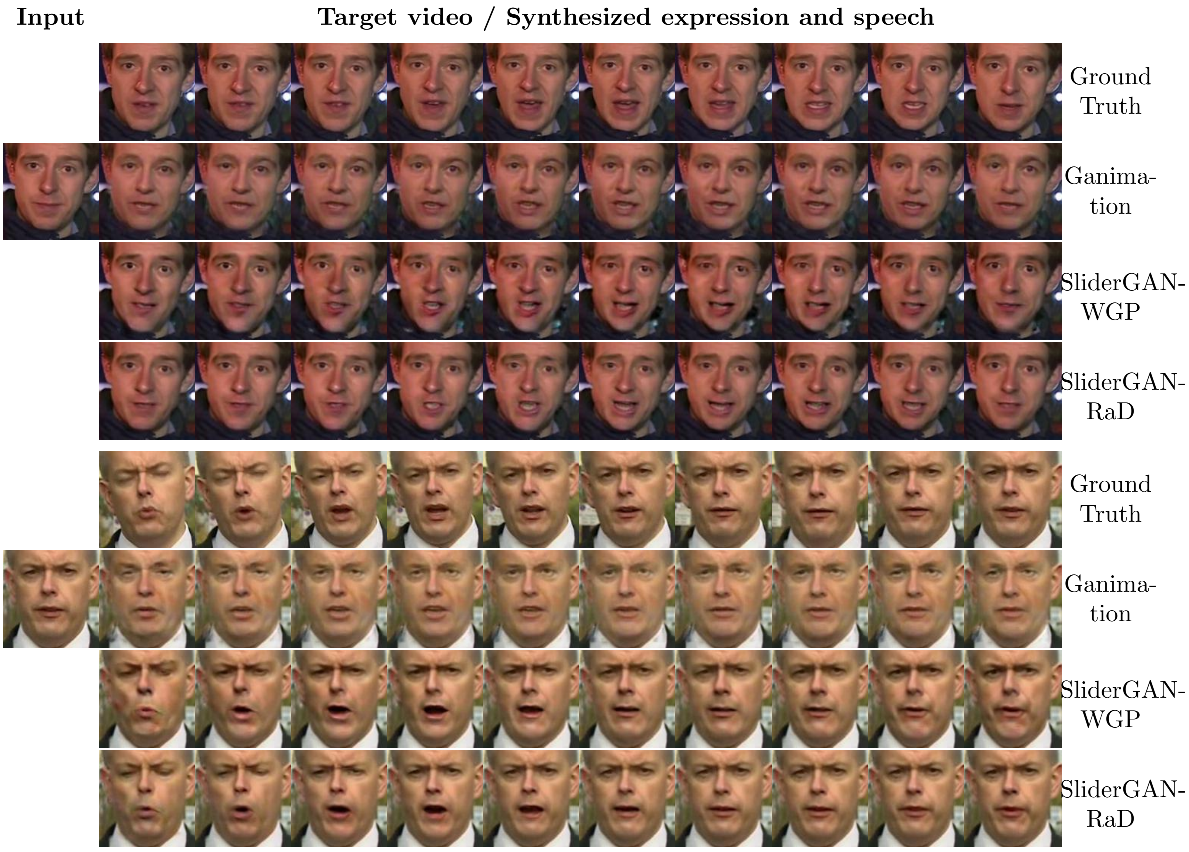


Figure 4.14: Comparison of combined expression and speech animation from a single input image between GANimation [20], SliderGAN-WGP and SliderGAN-RaD. We utilise as targets the expression and speech blendshape parameters of consecutive frames of a video of LRW. Then we reconstruct the expression and speech from a single input frame of the same video. Both SliderGAN implementations reconstruct face motion more accurately than GANimation. Also, the texture quality of the results is higher in SliderGAN-RaD than in SliderGAN-WGP as expected. (Please, zoom in the images to notice the differences in texture quality.)

of the 3D speech model $\mathcal{S}_{speech}$. That is we combined the parameters of two separate 3D blendshape models, $\mathcal{S}_{exp}$ and $\mathcal{S}_{speech}$, under our SliderGAN framework by stacking all 40 parameters in a single vector, to train a model which can generate frame sequences where both facial expression and lip/mouth motion varies. Simply stacking the parameters in one vector is a reasonable way to combine them in this case because $\mathcal{S}_{exp}$ and $\mathcal{S}_{speech}$ are linear models and have the same mean component (the LSFM mean face), which means that simple addition of instances of the two models yields possible 3D faces. Also, both include values in the interval $[-1, 1]$. We trained SliderGAN with 180,000 frames of LRW, after training with the warped images, without leveraging the temporal characteristics of the database, that is we shuffled the frames and trained our model with random target vectors to avoid

Table 4.3: Image Euclidean Distance (IED), calculated between ground truth images of LRW and corresponding generated images by Ganimation [20], SliderGAN-WGP and SliderGAN-RaD. Results from SliderGAN-RaD produce the lowest IED between the three methods, which indicates the robustness of blendshape coding for speech utlised by SliderGAN.

Method	IED
GANimation [20]	$3.07e-02$
SliderGAN-WGP	$1.14e-02$
SliderGAN-RaD	$\mathbf{9.35e-03}$

learning person specific deformations.

Qualitative Evaluation: Results of performing expression and speech synthesis from a video using a single image are presented in Figure 4.13 where the the parameters and the input frame belong to the same video (ground truth frames are available) and in Figure 4.14 where the parameters and the input frame belong to different videos of LRW.

For comparison we trained GANimation on the same dataset with AU activations obtained by OpenFace. As can be seen by Figure 4.14, GANimation is not able to accurately simulate the lip motion of the target video. On the contrary, SliderGAN-WGP simulates mouth and lip motion well, but produces textures that look less realistic. SliderGAN-RaD produces higher quality results that look realistic in terms of accurate deformation and texture.

Quantitative Evaluation: To measure the performance of our model we employ Image Euclidean Distance (IED) [243] to evaluate the results of expression and speech synthesis when the input frame and target parameters belong to the same video sequence. Due to changes in pose in the target videos, we align all target frames with the corresponding output ones before calculating IED. The results are presented in Table 4.3, where it can be seen that SliderGAN-RaD achieves the lowest error.

4.3.8 3D Expression Reconstruction

As also described in Section 4.3.4, a by-product of SliderGAN is the discriminator's ability to map images to expression parameters $D_{\mathbf{p}}$ that reconstruct the 3D expression as $\mathcal{S}_{exp}(D_{\mathbf{p}})$. We test the accuracy of the regressed parameters on images of Emotionet in two scenarios: a) we calculate the error between parameters recovered by 3DMM fitting and those regressed by D on the same image

Table 4.4: Expression representation results on SLiderGAN-RaD (blendshape parameters coding) and Ganimation (AUs activations coding). SliderGAN is capable to accurately and robustly recover expression representations, while GANimation fails to detect AUs activations.

	SliderGAN	GANimation [20]
$\frac{1}{N}\sum_{i=1}^{N}\frac{\|\mathbf{p}_{3DMM,i}-\mathbf{p}_{D,i}\|}{\|\mathbf{p}_{3DMM,i}\|}$	**0.131**	0.427
$\frac{1}{N}\sum_{i=1}^{N}\frac{\|\mathbf{p}_{trg,i}-\mathbf{p}_{D,i}\|}{\|\mathbf{p}_{trg,i}\|}$	**0.258**	0.513

(Table 4.4 row 1) and b) we test the consistency of our model and calculate the error between some target parameters $\mathbf{p}_{trg}$ and those regressed by D on a manipulated image which was translated to expression $\mathbf{p}_{trg}$ by SliderGAN-RaD (Table 4.4 row 2).

For comparison, we repeated the same experiment with GANimation for which we calculated the errors in AUs activations. For both experiments we employed 10000 images from our test set. The results demonstrate that the discriminator of SliderGAN-RaD extracts expression parameters from images with high accuracy compared to 3DMM fitting. On the contrary, GANimation's discriminator is less consistent in recovering AU annotations when compared to those of OpenFace. This, also, illustrates that the robustness of blendshape coding of expression over AUs, makes SliderGAN more suitable than GANimation for direct expression transfer.

Nevertheless, as it is reasonable to assume, 3DMM fitting is more stable and accurate in recovering expression parameters from images, than the trained discriminator. The superiority of 3DMM fitting is mostly evident in images with difficult faces and extreme expressions. As it can be seen in Figure 4.15, D produces substantially close 3D reconstruction results to those of 3DMM fitting for the easier image cases, which result in almost identical translated images. Contrarily, the regressed 3D expression reconstructions of D are obviously less accurate for the harder cases, which affects the quality of expression transfer between input and target images.

Lastly, D does not achieve state-of-the art results in 3D reconstruction of expression but allows our model to be independent from additional 3DMM fitting during testing, which is clearly an advantage. Alternatives, for more stable expression transfer between images would be to employ different DCNN-based models dedicated to blendshape parameters regression, or 3DMM fitting but with a higher cost in required resources and execution time.

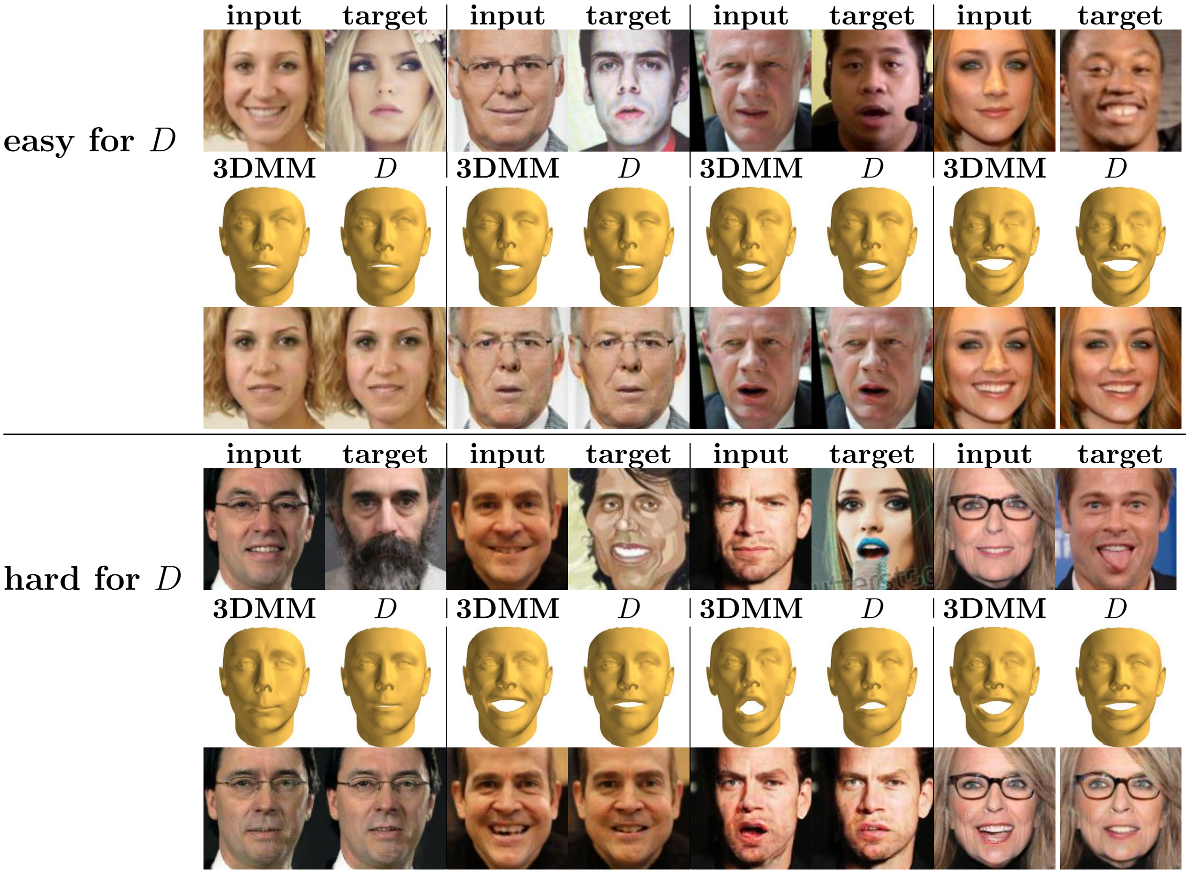


Figure 4.15: Comparison of image translation with expression parameters recovered from 3DMM fitting and the discriminator of SliderGAN. D recovers expressions adequately close to those of 3DMM fitting for most images which are noted as ”easy”. Then, the image translation in the two cases is almost identical. However, on ”hard” cases the accuracy of D drops, as also does the quality of expression editing.

4.3.9 Ablation Study

In this section, we investigate the effect of the different losses that constitute the total loss functions $\mathcal{L}_G$ and $\mathcal{L}_D$ of our algorithm. As discussed in Section 4.2.1, both training in a semi-supervised manner with loss $\mathcal{L}_{gen}$ and employing a face recognition loss $\mathcal{L}_{id}$ between the original and the generated images, contribute significantly in the training process of the generator G. In fact, we only focus on these two terms as they are essential for making SliderGAN robust against errors in expression parameters used as ground truth during training. These errors, caused by limitations of 3DMM fitting, make parameters to be mistakenly associated to more attributes of images than just expression (e.g pose, identity), as further discussed in Section 4.3.6. The rest loss terms of SliderGAN are either essential in

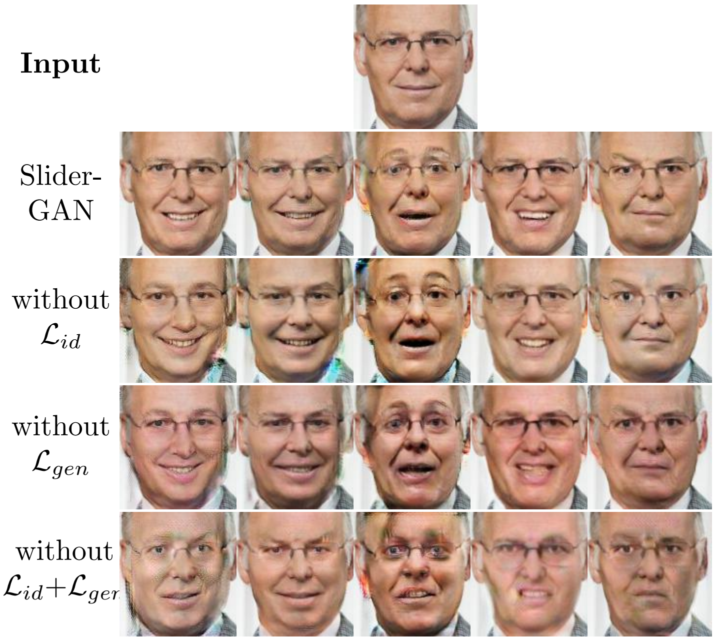

Input
Slider-GAN
without $\mathcal{L}_{id}$
without $\mathcal{L}_{gen}$
without $\mathcal{L}_{id}$+$\mathcal{L}_{ger}$

Figure 4.16: Results from the ablation study on SliderGAN's loss function components. It is evident that both losses $\mathcal{L}_{id}$ and $\mathcal{L}_{gen}$ have significant impact on the training of the model.

GAN training ($\mathcal{L}_{adv}$), or common in similar architectures such as the StarGAN and Ganimation ($\mathcal{L}_{rec}$, $\mathcal{L}_{exp,D}$, $\mathcal{L}_{exp,G}$, $\mathcal{L}_{att}$) and thus are not explicitly discussed.

To explore the extend at which these losses affect the performance of G, we consider three different models trained with variations of the loss function of SliderGAN which are: a) $\mathcal{L}_G$ does not include $\mathcal{L}_{id}$, b) $\mathcal{L}_G$ does not include $\mathcal{L}_{gen}$ and c) $\mathcal{L}_G$ does not include both $\mathcal{L}_{id}$ and $\mathcal{L}_{id}$. Figure 4.16 depicts results for the same subject generated by the three models as well as SliderGAN. As it can be observed in row ”without $\mathcal{L}_{id}$”, the absence of $\mathcal{L}_{id}$ results in images that clearly reflect the target expressions, but with changed identity and artifacts. Thus, $\mathcal{L}_{id}$ substantially supports retaining the identity between input and generated images. As it is shown in row ”without $\mathcal{L}_{gen}$”, training our model utilising $\mathcal{L}_{id}$ and not $\mathcal{L}_{gen}$ results in images with only slightly changed identity between input and output images, that however reflect other attributes of the target images along with expression such as pose, head shape and color.

When both $\mathcal{L}_{id}$ and $\mathcal{L}_{gen}$ are omitted as in row ”without $\mathcal{L}_{id} + \mathcal{L}_{gen}$”, both the identity preservation and the expression accuracy decrease drastically. Generally, the GAN loss is responsible for generating realistic images with higher frequency details that an l_1 or l_2 reconstruction loss cannot produce. How-

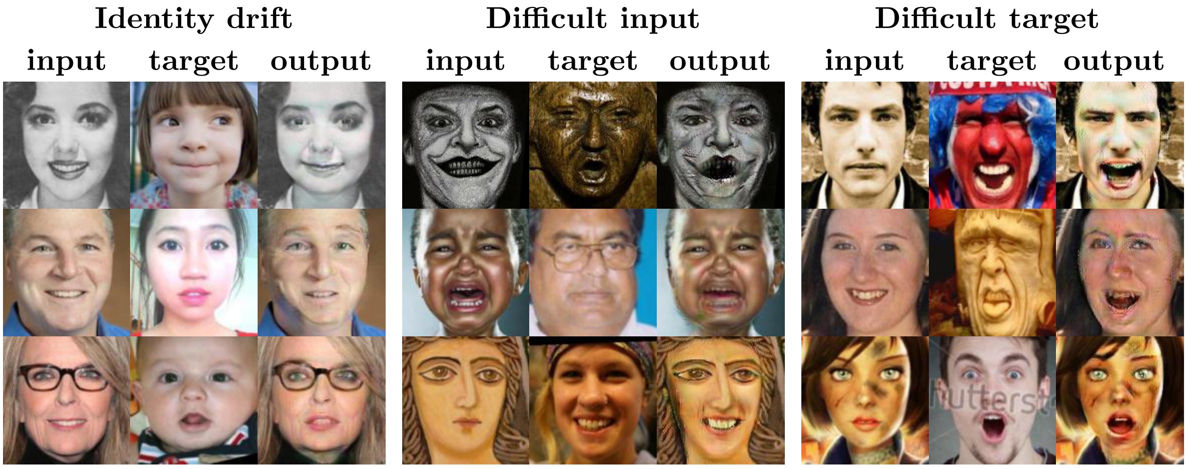


Figure 4.17: Limitations of SliderGAN. The main limitations are the identity transfer from target images to the output, the unsuccessful manipulation of non-natural images and the compromised generation of extreme expressions.

ever, in this case the GAN loss is not enough, because of the inconsistency of expression parameters which makes image generation problematic.

Finally, including both loss functions in training, enables SliderGAN to produce images that preserve all attributes of the input images but expression, which is manipulated according to the target expressions.

4.3.10 Limitations

In this section, we discuss the main limitations of our proposed model to indicate possible directions for improvement.

One important limitation is that SliderGAN does not always maintain the identity of the input images completely unchanged as can be seen in Figure 4.17. This happens mainly, in cases of extreme expressions or expressions with few close samples in the training set of real images. Thus, in those cases SliderGAN over-fits to specific images, reproducing the identity in the generator's output. This could probably be solved if a more balanced database in terms of expressions was employed. It is worth noting that the identities are perfectly maintained in the case of training with 4DFAB, which is a controlled database and includes lots of images for every expression.

Another limitation is generating extreme expressions or manipulating images with extreme expressions. In both cases, images often present a lot of artifacts as shown in Figure 4.17. This is because extreme expressions are not well represented in the training dataset and of course, bigger parts of the image have to be edited which makes it a more difficult task for the generator.

Lastly, editing non real faces, such as sketches of faces, faces of character models, faces with makeup etc., most often produces artifacts as shown in Figure 4.17, for the same reasons as editing extreme expressions.

4.4 Conclusion

In this Chapter, we presented SliderGAN, a very flexible way for manipulating the expression (i.e., expression transfer etc.) in facial images driven by a set of statistical blendshapes. To this end, a novel generator based on Deep Convolutional Neural Networks (DCNNs) is proposed, as well as a learning strategy that makes use of adversarial learning. Motivated by the success of relativistic discriminators in the task of super-resolution, a relativistic discriminator was employed to challenge our generator and enhance the resolution of the produced images. Moreover, a by-product of the learning process is a very powerful regression network that maps images into a number of blendshape parameters, which can be directly applied on target images to drive expression transfer without relying on any external models or 3DMM fitting. Lastly, we demonstrated that SliderGAN is able to edit images not only with regards to expresison, but any deformation type that can be expressed with blendshape models, such as speech.

CHAPTER 5

Weakly-Supervised Gaze Estimation from Synthetic Views

Contents

5.1 Introduction

Eye gaze serves as a cue for understanding human behavior and intents, including attention, communication and mental state. As a consequence, gaze information has been exploited by a lot of applications of various fields of interest, ranging from medical and psychological analysis [245, 246, 247] to human-computer interaction [248, 249], efficient rendering in VR/AR headset systems [250, 251, 252], virtual character animation [253, 254] and driver state monitoring [255, 256].

Undoubtedly, the most common approach to tackle gaze estimation has been by learning a direct mapping between eye or face images and few gaze coordinates or angles. To this end, numerous model design settings have been investigated recently, including the face region to use as input [22, 21, 25], the model architecture [257, 258, 24] and what external stimuli to utilise to improve performance [259]. Nevertheless, much effort has also been made to design models that generalise well to unseen subjects and environments, by employing either few labeled samples [260, 261, 262] or completely unlabeled data of the target domain [263, 264, 265]. Better yet, in recent works it has been shown that learning gaze from images can be achieved in fully unsupervised settings. Particularly, valuable gaze representations can be extracted from image encoder-decoder architectures by applying gaze redirection [266] or disentanglement [267] constraints. In addition, [268] shown that it is possible to train gaze estimation by employing geometric constraints in scenes depicting social interaction and particularly scenes of people looking at each other (LAEO).

Differently from the above, sparse or semantic representations of the eye geometry have also been employed by some methods to infer gaze from images [269, 23, 270, 271, 258]. However, such representations do not convey information about the 3D substance of eyes and are prone to noisy predictions. In contrast, by predicting 3D eye meshes we are able to learn a much more robust representation, from which we can retrieve any other sparse or semantic one just by indexing. Recovering dense 3D geometry of the eye region from images by fitting parametric models of the shape and texture has been previously proposed [269]. However, restrictions posed by building large-scale parametric models and fitting to "in-the-wild" images have resulted in low gaze accuracy compared to learning-based methods.

Most of the above methods predict gaze as either a 3D vector or spherical coordinates indicating the direction that someone is looking at, without considering any geometric representation of the eyes. Nevertheless, it has been shown that unconstrained face and body pose estimation from single images benefit from replacing predicting few pose or model parameters by directly predicting dense 3D geometry [272, 273, 274, 275]. To our knowledge, this observation has not been leveraged for eyes, and thus recovering gaze as a by-product of 3D eye reconstruction remains open for investigation.

Training to predict 3D geometry from images requires supervision from related ground truth. In [276] the authors have proposed a dataset of IR images and 3D eyes parameterised by the radius and eye center. However, IR images cannot be directly employed for RGB based methods. In addition, several gaze datasets have become recently available [277, 259, 278, 279, 280, 21, 281, 282, 283]. A straightforward approach to obtain 3D ground truth for these data, is to fit an eyeball using sparse eye landmarks and the available gaze labels. Still, collecting gaze datasets is a costly and challenging process which restricts them being captured in controlled environments and often consisting of limited different identities. This causes the most common challenge in gaze estimation, which is cross-domain generalisation. Nevertheless, images and videos of people "in-the-wild" are abundantly available in the internet. Thus, a reasonable question would be: *"Is it possible to utilise "in-the-wild" face images for improving generalisation of eye 3D reconstruction and thus, gaze-estimation?"*.

In this work, we propose to tackle gaze estimation as end-to-end 3D reconstruction of eyes using a dense coordinate regression approach. We acquire compatible 3D ground truth by defining a unified eye representation for all employed datasets, i.e. a 3D eyeball template (Fig. 5.3 (a)), which we fit on existing gaze datasets based on sparse landmarks and the available gaze labels. Additionally, we tackle the challenge of cross-domain generalisation by taking advantage of largely available "in-the-wild" face data and recent advances in weak-supervision of training CNNs for human perception tasks [39, 38, 284, 285, 40]. An overview of our method is presented in Fig. 5.1.

To obtain viable supervision from face data, we combine multiple geometric and multi-view consistency constraints. Particularly, we enforce geometric constraints which drive the outputs to follow the geometry of our defined 3D eyeball template. Additionally, to extract meaningful gaze information, we implement a weak-supervision, multi-view constraint which encourages our model to maintain consistency between the 3D eyes across multiple synthetic views of the same subject. We acquire novel views of a face by employing HeadGAN [3], a recently proposed method for face reenactment, which

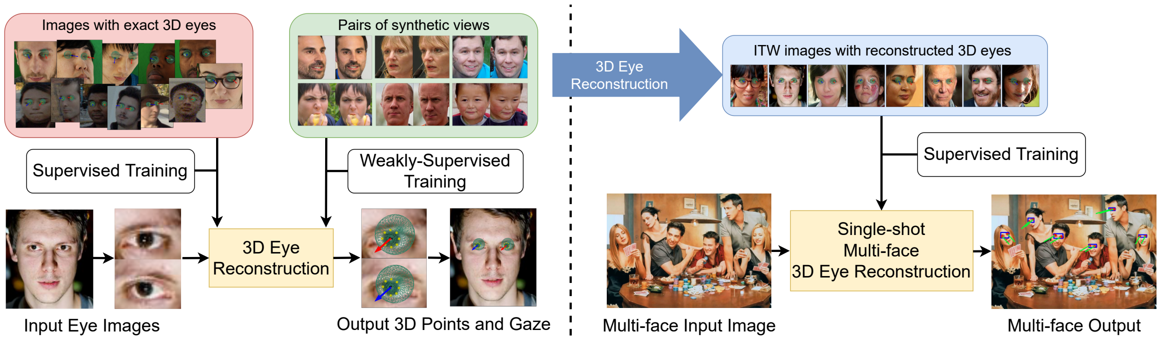


Figure 5.1: Overview of our 3D eye reconstruction approach to gaze estimation. a) Using weak-supervision from synthetic views of faces "in-the-wild", or semi-supervision by additionally employing data with exact ground truth, we are able to train 3D eye reconstruction. b) We exploit pseudo ground truth generated by our model to train a network for single-shot, multi-face 3D reconstruction of eyes "in-the-wild".

enables us to animate single images. HeadGAN manages to synthesise novel head poses while maintaining the relative difference between the gaze direction and head orientation in the generated image. This is because in HeadGAN image synthesis is conditioned on dense 3D representations of the face, which includes the eye regions.

We evaluate our 3D eye reconstruction method on common gaze estimation datasets including the "in-the-wild" Gaze360 [281]. We demonstrate that learning meaningful gaze information from "in-the-wild" face images is possible by our weakly supervised training approach and that including this loss in gaze estimation improves generalisation. Particularly, we demonstrate improvements in semi-supervised scenarios, where utilising "in-the-wild" face data helps to close the gap between different domains. Lastly, we prove the validity of our "in-the-wild" reconstruction results, by proposing and tackling the novel task of single-shot 3D reconstruction of eyes from multiple faces of an image or video frame.

To summarise, the key contributions of our work are:

- We revise the common approach of tackling 3D gaze estimation and propose to learn gaze as a by-product of dense 3D reconstruction of eyes from images. To the best of our knowledge, we are the first to adopt an end-to-end, regression-based approach to 3D eye reconstruction for gaze estimation.

- We propose a weakly-supervised framework to train 3D reconstruction of eyes, based on unlabeled images of faces "in-the-wild". We employ synthetic views of face images and design specific, multi-view geometric constraints which allows us to effectively learn gaze.
- We introduce the novel task of single-shot gaze estimation for all faces depicted in a particular frame, which we tackle based on robust gaze predictions extracted by our weakly-supervised framework. We demonstrate that we are able to achieve similar results with the state-of-the art, in O(1) regarding the number of faces in an image.
- We demonstrate the effectiveness of robust gaze pseudo-labels for the task of 3D gaze editing in images "in-the-wild" and showcase results similar to utilizing ground truth-supervised gaze estimation models.

5.2 Methodology

5.2.1 Problem Definition and Motivation

It is well known from previous work on "in-the-wild" face and body 3D reconstruction [272, 273, 274, 275], that accuracy and robustness benefit from predicting dense coordinates. To our knowledge, this observation has not been leveraged for estimating the geometry of eyes, replacing training for sparse points or few pose parameters with dense 3D coordinates. In this work, our goal is to learn to extract 3D gaze from images "in-the-wild", as a by-product of estimating dense 3D eye meshes. In more detail, we aim to design a method which given a face image $\mathbf{I}$, it estimates $2 \times N_v$ 3D coordinates $\mathbf{V} = [\mathbf{V}_l^\mathsf{T}, \mathbf{V}_r^\mathsf{T}]^\mathsf{T}$, where $\mathbf{V}_l \in \mathcal{R}^{N_v \times 3}$ are coordinates corresponding to the left eye while $\mathbf{V}_r \in \mathcal{R}^{N_v \times 3}$ to the right, from the subject's point of view.

Inspired by recent work in self-supervised 3D body pose estimation [38, 39, 40], we adopt multi-view constrains to train 3D reconstruction of eyes based on face images "in-the-wild". By enforcing additional geometric constraints, we are able to recover coordinates that adhere to a common 3D eye representation. To the best of our knowledge, this is the first time that "in-the-wild" face data without any gaze related annotation have been employed for eye 3D reconstruction and gaze estimation.

To train our model using multi-view losses, we assume that images of the same subject with different face poses and the same gaze direction relatively to the face are available. For example, this condition

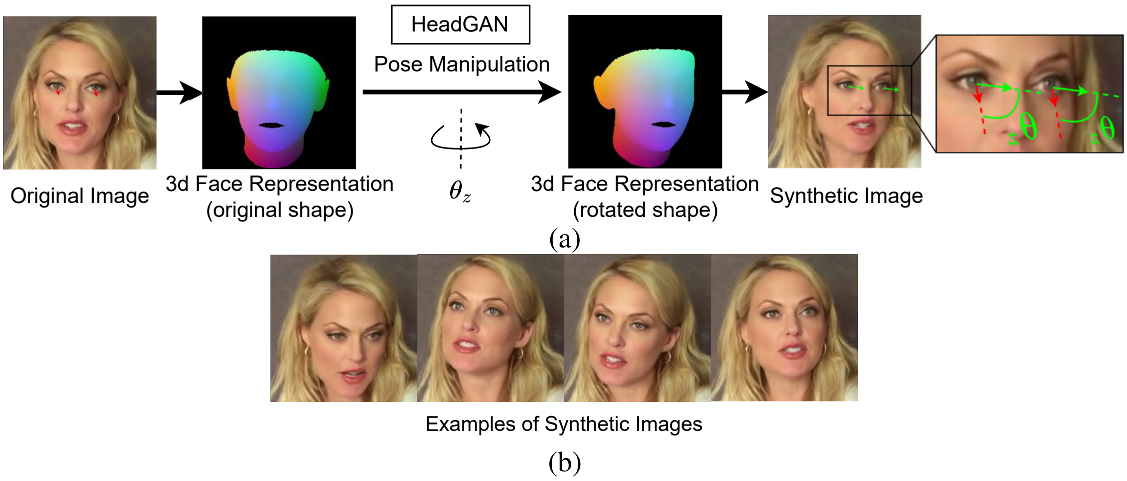

HeadGAN
Pose Manipulation
Original Image
3d Face Representation (original shape)
θ_z
3d Face Representation (rotated shape)
Synthetic Image
(a)
Examples of Synthetic Images
(b)

Figure 5.2: (a) We use HeadGAN [3] to generate novel views by manipulating the 3D pose of the face. During synthesis the face rotation angle θ_z is transferred to all facial parts, including the eyes, thus the relative angle between the head and eyes is maintained. (b) Synthetic examples generated with HeadGAN by rotating the face depicted in the original image.

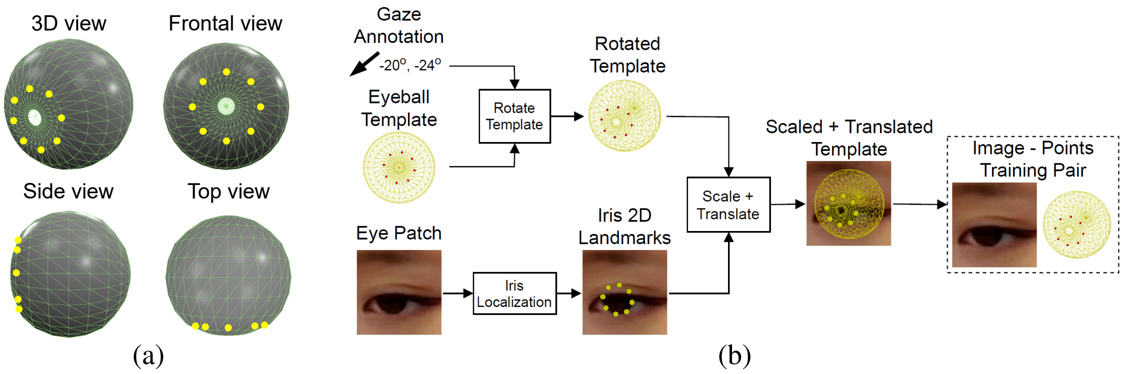


Figure 5.3: (a) Our eyeball mesh consisted of $N = 481$ vertices and $T = 928$ triangles. (b) Ground truth data generation pipeline, applied on samples of gaze estimation datasets for which gaze ground truth is available. The eyeball template is first rotated according the a 3D gaze annotation. Then, iris landmarks are employed to align the rotated eyeball in the image space, maintaining the original proportions in the depth axis.

is satisfied when a face picture is taken from different angles at the same time. As such images are not commonly available for "in-the-wild" datasets, we employ HeadGAN [3], a recent face reenactment method, to generate novel face poses from existing images. HeadGAN is able to synthesise face animations, using dense face geometry as driving signal and single source images. Using dense geometry guaranties that the relative angle between the head and eyes is maintained when synthesising novel poses, as it is shown in Fig. 5.2.

5.2.2 Unified 3D Eye Representation

Learning meaningful and consistent eye geometry across different images and datasets, requires establishing a unified 3D representation of eyes. To that end, we define a 3D eyeball template as a 3D triangular mesh with spherical shape, consisting of $N_v = 481$ vertices and $N_t = 928$ triangles. We create two mirrored versions, $\mathbf{M}_l$ and $\mathbf{M}_r$, of the above mesh to represent a left and a right reference eyeball respectively. This representation allows us to allocate semantic labels to different sets of vertices of the eyeball, such as the cornea and iris, as well as retrieving sparse point sets, such as the iris border (Fig. 5.3 (a)).

When gaze labels are available, as for example in gaze estimation datasets, exact supervision can be acquired by automatically fitting the eyeball template on face images, based on sparse landmarks around the iris and the available gaze labels, as also described in Fig. 5.3 (b). To create such ground truth data for our experiments, we employed the method of [23] to extract sparse iris landmarks from images, but any similar method could have been used.

5.2.3 Weakly-supervised 3D Eye Reconstruction

Given an input face image $\mathbf{I}$, we utilise 5 face detection landmarks to crop patches around each one of the two eyes. We resize the patches to shape $128 \times 128 \times 3$ and stack them channel wise, making sure that the first three channels correspond to the left eye while the next three to the right. We employ a simple model architecture consisting of a ResNet-34 [286] to extract features from the eye images, followed by a fully connected layer which maps features to eye coordinates in the image space. We aim to train the above network relying on supervision by pairs of images of the same subject with different face pose, but the same relative angle between the face and gaze direction. By enforcing additional geometric constraints to ensure that the output will adhere to the eyeball templates $\mathbf{M}_l$ and $\mathbf{M}_r$, we are able to recover 3D meshes that correctly represent the shape and size of eyes in images but also provide meaningful gaze predictions. A visual representation of our approach, as well as the data flow and losses used n training can be seen in Figure 5.4. In the rest of this section we detail the multiple losses employed by our algorithm.

Pair Supervision Loss: Recovering dense 3D face geometry and pose from images has recently been quite reliable [272, 272, 12, 287]. Having a pair of images $\mathbf{I}_1$ and $\mathbf{I}_2$ of the same subject and their

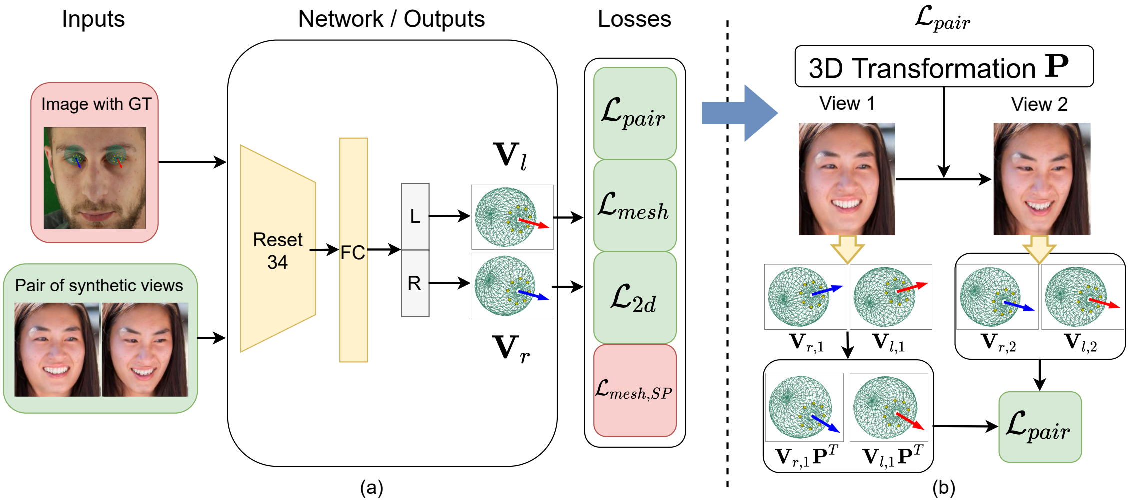


Figure 5.4: Overview of our weakly-supervised approach to 3D eye reconstruction from pairs of synthetic views of faces "in-the-wild". a) Overview of the main training components employed by our method. Input can be either pairs of synthetic images of the same subject with different head poses or single images with ground truth gaze annotations. The designed model outputs both left and right 3D eyes in image space in a single network pass. Different sets of losses are employed depending on the type of supervision. b) Detailed demonstration of $\mathcal{L}_{pair}$. 3D transformation $\mathbf{P}$ which maps view 1 to view 2, is employed to transform points $\mathbf{V}_{l,1}$ and $\mathbf{V}_{r,1}$, before calculating an $L1$ distance loss against $\mathbf{V}_{l,2}$ and $\mathbf{V}_{r,2}$.

reconstructed 3D faces, we can compute a transformation matrix $\mathbf{P} \in \mathcal{R}^{3\times4}$ which aligns the two faces in image space. Assuming that gaze direction in both images remains still relatively to the face, as is the case with images created by HeadGAN, we are able to supervise 3D reconstruction of eyes without depending on ground truth. That is, we are able to restrict our model's reconstruction to be consistent over the image pair, as output vertices should coincide when transformation $\mathbf{P}$ is applied to one of the pair's outputs. Particularly, we form the following pair vertex reconstruction loss:

$$\mathcal{L}_{pair} = \frac{1}{N_v} \sum_{j=\{l,r\}} \sum_{i=1}^{N_v} \left\| \mathbf{V}_{1,j,i}\mathbf{P}^\mathsf{T} - \mathbf{V}_{2,j,i} \right\|_1, \tag{5.1}$$

where $\mathbf{V}_{1,j}, \mathbf{V}_{2,j} \in \mathcal{R}^{N_v\times4}$ for $j = \{l,r\}$ are the output matrices for left and right eyes, which correspond to input images $\mathbf{I}_1$ and $\mathbf{I}_2$. $\mathbf{V}_{1,j,i}, \mathbf{V}_{2,j,i} \in \mathcal{R}^4$ are the specific homogeneous 3D coordinates indexed by i in the above matrices. We add an extra column of ones on both output matrices to ensure coordinates are homogeneous and thus, compatible with transformation $\mathbf{P}$.

Mesh Reconstruction Losses: To obtain meaningful eye geometry on our network's output, we augment our overall loss with two mesh reconstruction losses. In particular we employ a vertex loss and an edge length loss between the model outputs and reference meshes, both of which regulate the locations of vertices to be close to the reference topology.

As we need eye reconstructions to follow gaze direction, direct comparison of the vertices between outputs and the frontal facing reference meshes is not possible. To overcome this obstacle, we simply compute a transformation $\mathbf{P} \in \mathcal{R}^{3\times 4}$ between the model output and the reference eyeball template and apply it on the first, repeating it for both left and right eye. Then, our vertex reconstruction loss for each image of a training pair can be written as:

$$\mathcal{L}_{vert} = \frac{1}{N_v} \sum_{j=\{l,r\}} \sum_{i=1}^{N_v} \left\| \mathbf{V}_{j,i} \mathbf{P}_j^{\mathsf{T}} - \mathbf{M}_{j,i} \right\|_1, \tag{5.2}$$

where $\mathbf{V}_j \in \mathcal{R}^{N_v \times 4}$, $\mathbf{M}_j \in \mathcal{R}^{N_v \times 4}$, $\mathbf{P}_j \in \mathcal{R}^{3\times 4}$ for $j = \{l, r\}$ are the output matrices, the corresponding reference coordinates and the transformations between them for both left and right eyes. Finally, to compute the loss for a pair of images, we just add the two losses.

Similarly to the vertex loss, calculating the edge length loss requires aligning the output coordinates to the reference meshes to maintain consistent scaling. By employing the fixed mesh triangulation of our template meshes, we compute the following loss for each image:

$$\mathcal{L}_{edge} = \frac{1}{3N_t} \sum_{j=\{l,r\}} \sum_{i=1}^{3N_t} \left\| \mathbf{E}_{j,i} - \mathbf{E}_{M,i} \right\|_1 \tag{5.3}$$

where $\mathbf{E}_j \in \mathcal{R}^{3N_t}$ for $j = \{l, r\}$ are the edge lengths of the predicted eyes, $\mathbf{E}_M \in \mathcal{R}^{N_t}$ the edge lengths of the reference mesh and $3N_t$ is the number of edges of our eye template. As edge length we define the euclidean distance between two vertices of the same triangle. Finally, for a pair of training images we calculate the loss independently and add the two losses.

By combining the vertex and edge length losses, we get a mesh reconstruction loss written as:

$$\mathcal{L}_{mesh} = \mathcal{L}_{vert} + \lambda_e \mathcal{L}_{edge}, \tag{5.4}$$

where λ_e is a parameter which regularises the contribution of the two terms in the overall loss. From our experiments we have selected its value to be $\lambda_e = 1$.

Sparse 2D Landmarks Loss: Estimating eye coordinates in image space benefits by applying an additional sparse landmarks loss. In particular, we employ a 2D supervision loss between a set of

N_{iris} 2D landmarks on the iris contour and N_{iris} points picked from the output using specific indexes known by the eyeball template. The 2D landmarks loss can be written as:

$$\mathcal{L}_{2d} = \frac{1}{N_{iris}} \sum_{j=\{l,r\}} \sum_{i=1}^{N_{iris}} \left\| \mathbf{V}_{j,i}^{2d} - \mathbf{V}_{j,i}^{2d*} \right\|_1, \tag{5.5}$$

where $\mathbf{V}_j^{2d} \in \mathcal{R}^{N_{iris} \times 3}$ for $j = \{l, r\}$ are estimated 2d coordinates for the left and right eye and $\mathbf{V}_j^{2d*} \in \mathcal{R}^{N_{iris} \times 3}$ for $j = \{l, r\}$ are ground truth sparse landmarks obtained by an existing eye landmark localisation method. In our case we employed the method of [23]. $\mathcal{L}_{2d}$ constitutes the only ground truth supervision included in our weakly supervised training method. However, $\mathcal{L}_{2d}$ does not include any cues about gaze direction, which is learnt mainly by $\mathcal{L}_{pair}$.

Weakly-supervised and Semi-supervised Training We consider two scenarios for training our algorithm. One is to weakly-supervise training, utilising only pairs of synthetic images of faces “in-the-wild”. In that case, we calculate the overall loss as the weighted sum of the three losses described above:

$$\mathcal{L} = \mathcal{L}_{pair} + \lambda_m \mathcal{L}_{mesh} + \lambda_{2d} \mathcal{L}_{2d}, \tag{5.6}$$

where λ_m and λ_{2d} are parameters, regularising the contribution of the three terms in the overall loss. From our experiments we have selected the parameter values to be $\lambda_m = 0.1$ and $\lambda_{2d} = 10$.

The next training scenario is referred to semi-supervised training, in which pair supervision from synthetic images supports training with exact ground truth from gaze datasets. In more detail, we first train our network to convergence with full supervision from the exact eye coordinates. To this end, we optimise the mesh loss only, which we calculate between the predicted and the ground truth coordinates without aligning them as they already lie in the same space. We refer to this loss as supervised mesh loss, $\mathcal{L}_{mesh,SP}$, and keep the same value for the weighting parameter $\lambda_e = 1$. We then substitute the supervised loss by the loss in Eq. 5.6 and continue training as in the weakly-supervised case.

5.2.4 Single-shot 3D Reconstruction of Eyes from Multiple Faces

In Fig. 5.5, we present the framework of the proposed single-shot, multi-face gaze estimation method inspired by RetinaFace [272]. As can be seen, our model consists of two main components: the feature pyramid network and the multi-task loss. In the feature pyramid network, we use ResNet-34 [286]

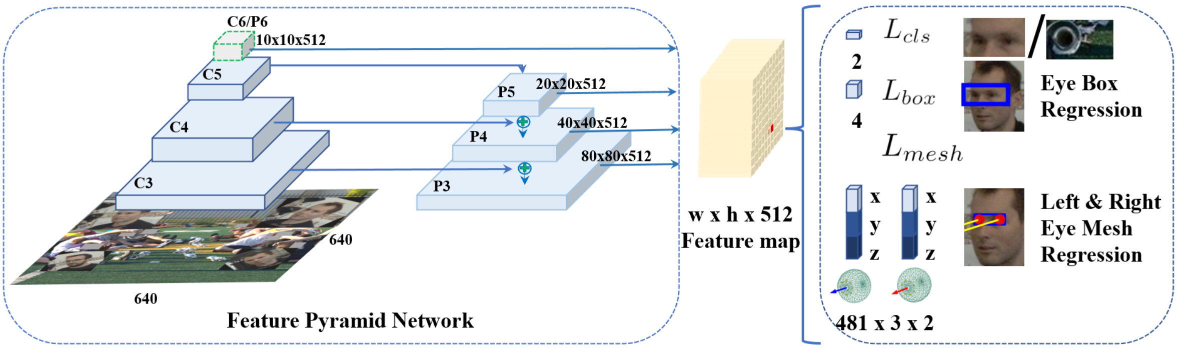


Figure 5.5: Network structure and multi-task loss function of the proposed single-shot multi-face gaze estimation. We synthesise multi-face images by putting weakly-supervised gaze data on the WIDER FACE images [290].

as the backbone, Path Aggregation Feature Pyramid Network (PAFPN) [288] as the neck, and two stacked 3×3 convolutional layers for the head. For the anchor setting, we tile multi-scale anchors of 32, 64, 128 and 256 on the feature maps of stride 8, 16, 32, and 64, respectively. The anchor ratio is set as 0.5. For each training anchor i, we minimise the following multi-task loss:

$$\mathcal{L} = \mathcal{L}_{cls}(p_i, p_i^*) + \lambda_1 p_i^* \mathcal{L}_{box}(t_i, t_i^*) + \lambda_2 p_i^* \mathcal{L}_{mesh}(v_i, v_i^*). \tag{5.7}$$

where t_i, v_i are eye region box and 3D eye vertices predictions, t_i^*, v_i^* is the corresponding ground-truth generated by the proposed weakly supervised learning, p_i is the predicted probability of anchor i being an eye region, and p_i^* is 1 for the positive anchor and 0 for the negative anchor. The classification loss $\mathcal{L}_{cls}$ is the softmax loss for binary classes (eye/not eye). For eye box regression and eye vertices regression, we follow [289] and use the smooth-L$_1$ loss. The loss-balancing parameters λ_1 and λ_2 are both set to 1. The proposed method employs fully convolutional neural networks, thus it can be easily trained in an end-to-end way.

5.3 Experiments

In this section, we evaluate our gaze estimation approach under full, weak and semi supervision settings, comparing against state-of-the-art methods. We, then, provide an ablation study on the loss terms of our weak supervision algorithm. Lastly, we evaluate the performance of our single-shot, multi-face gaze estimator and demonstrate its application in the task of Looking At Each Other (LAEO).

5.3.1 Datasets

Gaze Datasets: To build gaze datasets, typically, subjects are captured by multi-camera set-ups, under controlled lighting conditions while their gaze is tracked by specialised gaze tracking hardware. Collected by this method, ETH-XGaze [278] includes large variation in face pose and gaze and consists of 756K frames of 80 subjects. Similarly, UTMV [282], for which face pose and gaze variation is acquired by a reconstruction and synthesis pipeline, consists of 64K real frames of 50 subjects. Additionally, Columbia [277] consists of 5880 images of 56 subjects. Another common approach for collecting gaze datasets is by asking subjects to follow visual targets on the screen of a phone or tablet, while being captured by the device's camera. Collected in this way, MPIIGaze [280] includes smaller face pose and gaze variation and consists of 213,659 frames of 15 subjects, while GazeCapture [21] contains almost 2M frontal face images of 1474 subjects. In contrast to the above datasets, which have been collected in indoor environments, Gaze360 [281], is the only gaze dataset captured both indoors and outdoors and include large variation in face pose and gaze as well as lighting and backgrounds. Besides, it consists of 127K training sequences from 365 subjects.

In-The-Wild Face Dataset: In contrast to gaze datasets, face datasets "in-the-wild" consist of significantly more unique subjects and capturing environments. Incorporating variation of face data in gaze estimation could be valuable for improving generalisation to unseen and "in-the-wild" scenarios. To this end, we employed VGGFace2 [291], a large-scale dataset for face recognition, which includes 3.31M images of 9131 subjects downloaded from the internet, containing large variations in pose, age, illumination and ethnicity. To train our weakly-supervised method we synthesised one novel head pose from each image using HeadGAN, sampling the pitch and yaw angles, relatively to the original ones, by Gaussians with zero mean and 20 degrees standard deviation. We name this collection of images as "In-The-Wild Gaze" dataset (ITWG) and employ it in our experiments to improve generalisation of gaze estimation and create robust gaze annotations to build our single-shot, "in-the-wild" gaze estimator.

Social Interaction Datasets: To prove the effectiveness of our single-shot method, we employ it to predict the Looking At Each Other (LAEO) task in the wild. To that end, we leverage the large-scale human activity dataset AVA [292] with LAEO annotations [293, 294]. It contains 40,166 and 10,631 frames in its train and validation subsets respectively. The annotations of these frames are formed by pairs of head bounding boxes, with one of the three labels: $\{$LAEO, not-LAEO, ambiguous$\}$. In total, AVA-LEAO consists of 19K LAEO pairs and 118K not-LAEO pairs in the training subset while 5.8K LAEO pairs and 28K pairs not-LAEO pairs in the validation subset. Apart from the LAEO task, we employ the visible faces of AVA for weak supervision, in Section 5.3.4, for which we acquire novel views using HeadGAN. Similarly, we employ CMU Panoptic [295], which captures interactions of multiple people in the same scene.

5.3.2 Implementation Details.

Training Details of Single-face Gaze Estimation: For the proposed weakly-supervised training and semi-supervised training, we initialise ResNet-34 with weights pre-trained using ImageNet [296]. We use a batch size of 32 pairs/images to train our network with weak/full supervision. We train using the Adam optimiser [242] with a learning rate of 10^{-4}. We stop weakly-supervised training when $\mathcal{L}_{pair}$ converges, which is usually after 10-15 epochs for semi-supervision and 15-20 epochs for weak-supervision.

For the training of single-shot multi-face gaze estimation, we adopt the SGD optimiser (momentum 0.9, weight decay 2e-4) with a batch size of 8×4 and train on four Tesla V100 GPUs. The learning rate is linearly warmed up to 0.01 within the first epoch, and then multiplied by 0.1 at the 10-th and 16-th epochs. The learning process terminates at the 20-th epoch.

Training Details of Single-shot Multi-face Gaze Estimation: For the training of the proposed single-shot multi-face gaze estimation, we employ the open-source MMDetection [297], which is implemented in PyTorch. We first synthesise multi-face images by putting weakly-supervised gaze data on the WIDER FACE images [290]. The eye region box is the bounding box of the left and right eye mesh. For the scale augmentation, square patches are cropped from the original images with a random size ([0.3, 1.5]), and then these patches are resized to 640×640 for training. Besides scale augmentation, the training data are also augmented by color distortion and random horizontal flipping, with a probability of 0.5. Inspired by RetinaFace [272] and SCRFD [298], we employ Adaptive Training Sample Selection (ATSS) [299] for positive anchor matching. In the detection head, weight sharing and Group Normalisation [300] are used. The losses of classification and regression branches are Generalised Focal Loss (GFL) [301] and DIoU loss [302], respectively.

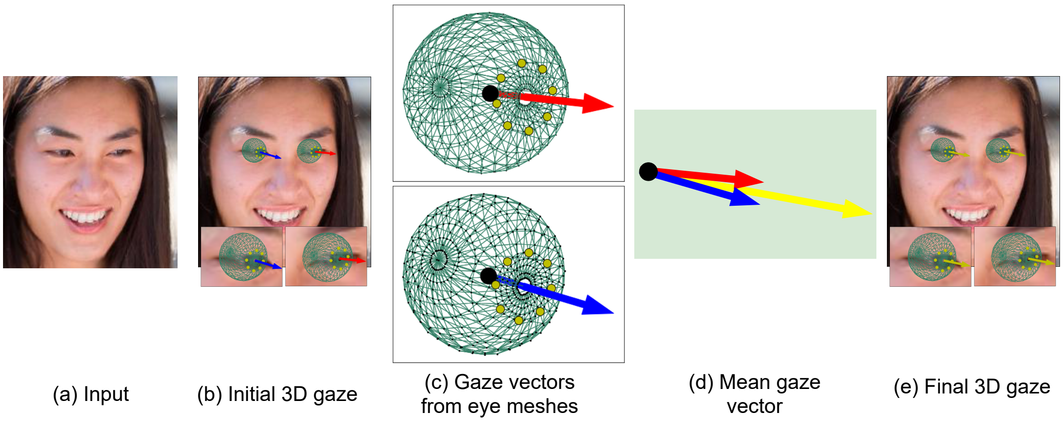


Figure 5.6: Calculating 3D gaze from eye meshes. Given 3D eye meshes extracted by our method, we calculate gaze direction as the mean of the two independent gaze vectors from the left and right eyes.

For the look at each other task, we train a three-layer fully connected network taking the gaze estimation results of two persons as the input. The detailed settings of the MLP are as follows: (1) 3 layers; (2) hidden layer 1024; (3) PRelu; (4) BN; (5) no dropout; (6) batch size 4096; and (7) shortcut connection in the middle layer.

Calculating gaze direction from 3D eye meshes: In this work, we have proposed a method to estimate 3D eye meshes from images and employ them for gaze estimation. During test time, having recovered a 3D eyeball meshes for both eyes, with topology adhering to our 3D eyeball template, we calculate gaze from the orientation of the central axis of the eyeballs. Particularly, we calculate 3D gaze vectors using the centre of each eyeball and the centre of the iris as shown in Fig. 5.6 (c). After obtaining 3D gaze vectors from both left and right eyes, we add the two vectors to retrieve a final gaze prediction, Fig. 5.6 (d).

5.3.3 Gaze Estimation via 3D Eye Reconstruction

Here we experimentally evaluate our suggestion that gaze estimation performance benefits from replacing the training target from gaze vectors or angles to 3D dense eye coordinates. To this end we employ the fully supervised version of our model, utilising data with exact ground truth and $\mathcal{L}_{mesh,SP}$ for training. We conduct within-dataset, cross-subject experiments on 5 commonly utilised gaze databases, namely Columbia [277], MPIIGaze [280], UTMV [282], and Gaze360 [281] and GazeCapture [21], for which specific data split for training and testing are provided. Additionally, for Gaze360

Table 5.1: Comparison between state-of-the-art gaze estimation methods, our 3D reconstruction approach (mesh) and 3D gaze vector regression (vector), on within-dataset experiments. Training with mesh targets leads to lower gaze error in all experiments. We also report the performance of our ”Unified” gaze estimation model, which achieves further decrease in the gaze error, due to the larger variation of the combined datasets. In all experiments, the gaze error is measured in degrees.

Dataset	Other methods									Within-dataset		Unified	
	[303]	[266]	[271]	[263]	[25]	[267]	[260]	[281]	[268]	vector	mesh	vector	mesh
Columbia	1.3	3.4	3.6	-	-	3.5	-	-	-	3.8	3.7	3.6	**3.4**
MPIIGaze	5.3	-	4.6	3.7	4.1	-	5.3	-	-	4.1	4.0	4.4	**4.0**
UTMV	-	5.5	-	-	-	4.8	-	-	-	5.6	4.1	5.5	**3.9**
Gaze360	-	-	-	-	-	-	-	11.1	10.1	12.7	10.4	12.4	**9.8**
GazeCapture	-	-	-	-	-	-	3.49	-	-	3.3	3.1	3.1	**2.7**

we consider only the frontal facing images as our method operates on eye patches. Additionally, we employ all 5 datasets under our unified 3D eye representation to train a ”Unified” gaze estimation model and report results on testing on the test set of each dataset.

We compare against state-of-the-art methods [303, 266, 271, 263, 25, 267, 260, 268, 281] and demonstrate that by simply utilising 3D dense coordinates instead of gaze vectors or angles, we are able to get close or even beat their performance. We report results in Table 5.1. For reference, we also report results of our fully-supervised design, trained on predicting 3D gaze vectors instead of coordinates (vector). The reason behind the lower gaze error achieved when training with 3D mesh targets (mesh) is that the final gaze is calculated from a large number of predicted parameters (the dense 3D eye coordinates), which makes predictions robust to small errors. On the other hand, when regressing few pose or sparse shape parameters, small prediction errors might lead to large errors in gaze direction. This finding is in line with results on dense face prediction [272], where the motivation of this work comes from. It also is worth noting here that our method employs a simple network architecture and training pipeline, while most methods consist of elaborate models or training schemes, designed to improve gaze accuracy. Lastly, with our unified 3D eye representation we are able to achieve further improvement in gaze accuracy because of integrating variation from multiple datasets which is particularly important for shape regression applications.

Table 5.2: Weakly-supervised method evaluation across three experimental settings. In all three settings we calculate the gaze error in degrees, on the test split of Gaze360. Particularly, we consider only the frontal facing images of Gaze360 (yaw angle of head pose from -90 to 90 degrees). For weak supervision we employ our In-The-Wild Gaze dataset (ITWG), as well as CMU and AVA to provide a clearer comparison with [268]. Throughout the experiments we investigate the effect of different supervision and datasets in the gaze error. In all cases, our method achieves the best performance, outperforming [268], when training with the large scale ITWG dataset, leveraging the wide variation of "in-the-wild" faces and capturing conditions.

(a) Within-dataset
Gaze + Synthetic Views

Dataset	[281]	[268]	Ours
G360	11.1	10.1	10.4
G360+AVA	-	10.2	**9.8**
G360+ITWG	-	-	**9.0**

(b) Cross-dataset
Synthetic Views

Dataset	[268]	Ours
CMU	29.0	30.3
AVA	26.0	28.4
CMU+AVA	22.5	25.4
ITWG	-	**21.2**

(c) Cross-dataset
Gaze + Synthetic Views

Dataset	[278]	[268]	Ours
EXG	27.3	20.5	22.3
EXG+AVA	-	16.9	18.9
EXG+ITWG	-	-	**16.2**
GC	30.2	29.2	29.5
GC+AVA	-	19.5	20.9
GC+ITWG	-	-	**18.1**

5.3.4 Semi-supervised Method Evaluation

Within-dataset and Cross-dataset Evaluation

In this section, we evaluate our approach in both the weakly-supervised and semi-supervised settings. We believe that [268] is the most similar method to ours, as it improves gaze estimation generalisation based on weak-supervision from "in-the-wild" data of social interaction between people.

We design three experiments to test the performance of our method on the "in-the-wild" gaze dataset Gaze360 (G360) and report the results on Table 5.2. Particularly, the experiments are the following: a) within-dataset evaluation on Gaze360 with additional weak supervision from AVA and ITWG, b) cross-dataset evaluation in which we train on the CMU, AVA and ITWG datasets in a purely weakly-supervised approach and test on Gaze360 and c) cross-dataset evaluation on Gaze360 with ground truth supervision from ETH-XGaze (EXG) or GazeCapture (GC) and weak supervision from AVA and ITWG (semi supervision experiment).

From the above experiments, it becomes obvious that weak supervision from multiple views effectively supports gaze estimation generalisation to unseen domains, even without any available gaze annotation. From Table 5.2, it can be seen that our method always outperforms [268] when our ITWG

Table 5.3: Ablation study on the losses of our weakly-supervised method, in which we train on CMU+AVA as well as ITWG and evaluate on the test set of Gaze360 (frontal facing images of Gaze360 with yaw angle of head pose from -90 to 90 degrees). In all cases the gaze error is measured in degrees. Results show that $\mathcal{L}_{mesh}$ and $\mathcal{L}_{pair}$ are crucial to learn any meaningful gaze information as leaving any of these out of the training process leads to large gaze errors. Especially, without $\mathcal{L}_{mesh}$ there is no supervision about the shape of eyes, leading to output meshes of random shapes.

Dataset	$\mathcal{L}_{pair}$+$\mathcal{L}_{2d}$	$\mathcal{L}_{mesh}$+$\mathcal{L}_{2d}$	$\mathcal{L}_{mesh}$+$\mathcal{L}_{pair}$	all
CMU+AVA	55.4	39.1	34.2	25.4
ITWG	60.3	41.6	35.0	21.2

dataset is employed. This proves the ability of our method to benefit from large scale “in-the-wild” face datasets, without requiring LAEO or gaze annotations. In experiments (b) and (c), [268] performs better when AVA or CMU datasets are employed for weak supervision. This is because [268] benefits from the LAEO labels of these datasets, as well as the fact that our method employs only the visible faces and not the ones turned more that 90 degrees away from the camera.

As expected, ground truth gaze labels allow for better gaze estimation performance in cases (a) and (c) when compared to (b), while in (a) training with data from the same domain leads to the highest accuracy. Moreover, based on our experimentation, even though novel synthetic views are useful for weak supervision, similarly augmenting existing gaze datasets for full supervision in within-dataset experiments does not improve gaze estimation accuracy, even though exact gaze labels are available. This is probably due to slight inaccuracies in face synthesis.

Ablation Study of Weakly-supervised Losses

To better understand the effect of each term of Eq. 5.6, we repeat the weakly-supervised experiments (b), in which we employ different subsets of the three loss terms, and present results in Table 5.3. It is natural to expect $\mathcal{L}_{mesh}$ to be crucial for our gaze estimation algorithm, as it reassures possible output eye meshes, without which $\mathcal{L}_{pair}$ cannot operate as designed. Besides, removing $\mathcal{L}_{pair}$ removes any gaze learning capabilities from our algorithm. The increased gaze accuracy is only due to having possible eyeballs on the model’s output. Contrary to the above, our method maintains some gaze learning potential even without $\mathcal{L}_{2d}$, which however is far from optimal. $\mathcal{L}_{2d}$ is also important for convergence as removing it led many times to unstable training in our experiments.

Table 5.4: Comparison between single- and multi-face gaze estimation models on the Gaze360 test set (frontal facing images of Gaze360 with yaw angle of head pose from -90 to 90 degrees). "Unified" refers to all five datasets with gaze annotations. In all experiments the gaze error is measured in degrees. Results show that the two methods are adequately close in gaze error, while the multi-face one provides the benefit of unchanged processing time regardless of the number of faces in a particular image.

Training Data	Single-face	Multi-face
Unified	9.8	10.3
Unified+ITWG	8.9	9.5

Table 5.5: Comparison of LAEO results on the AVA-LAEO dataset. We report Average Precision (AP) at the pair@frame level for AVA-LAEO. Results show that combining LAEO-NET++ with outputs of our multi-face gaze estimation model, leads to the highest AP.

Method	AP
LAEO-Net (pre-trained) [293]	50.6
LAEO-Net++ (self-supervised) [294]	68.7
Gaze Estimation	42.5
LAEO-Net++ & Gaze Estimation	**70.6**

5.3.5 Evaluation of Multi-face Gaze Estimation

In this section, we experimentally evaluate the effectiveness of the proposed single-shot multi-face gaze estimation method. As shown in Tab. 5.4, we first compare the accuracy of gaze estimation models trained under single-face and multi-face settings. When the unified labelled data are used for training, the multi-face gaze estimation model achieves 10.3.

Even though the single-face gaze model is slightly better than the multi-face gaze model, the complexity of the multi-face gaze model is O(1) regarding the number of faces in an image. By employing the proposed large-scale ITWG, the multi-face gaze model obtains 9.5, which indicates that the proposed weakly-supervised gaze data is effective for improving gaze estimation in the wild. In Fig. 5.7, we show some multi-face gaze estimation results on AFW [304] and PASCAL [305]. As we can see from the last row, the proposed single-shot method is robust under pose variations and occlusions (e.g., sunglasses and hats), which indicates that our model also takes the context information to estimate the gaze. Regarding the performance of eye detection, we associate the face boxes and eye boxes by using RetinaFace [272], and the APs on AFW and PASCAL are 89.05% and 84.52%, respectively. Under the input resolution of 640×640, the proposed multi-face gaze estimation model can run in real-time (23.5ms) on GPU-2080ti.

Besides, we also evaluate the multi-face gaze estimation model for the task of Looking At Each Other (LAEO). A detected pair is correct if both heads are correctly localised and its label (LAEO/

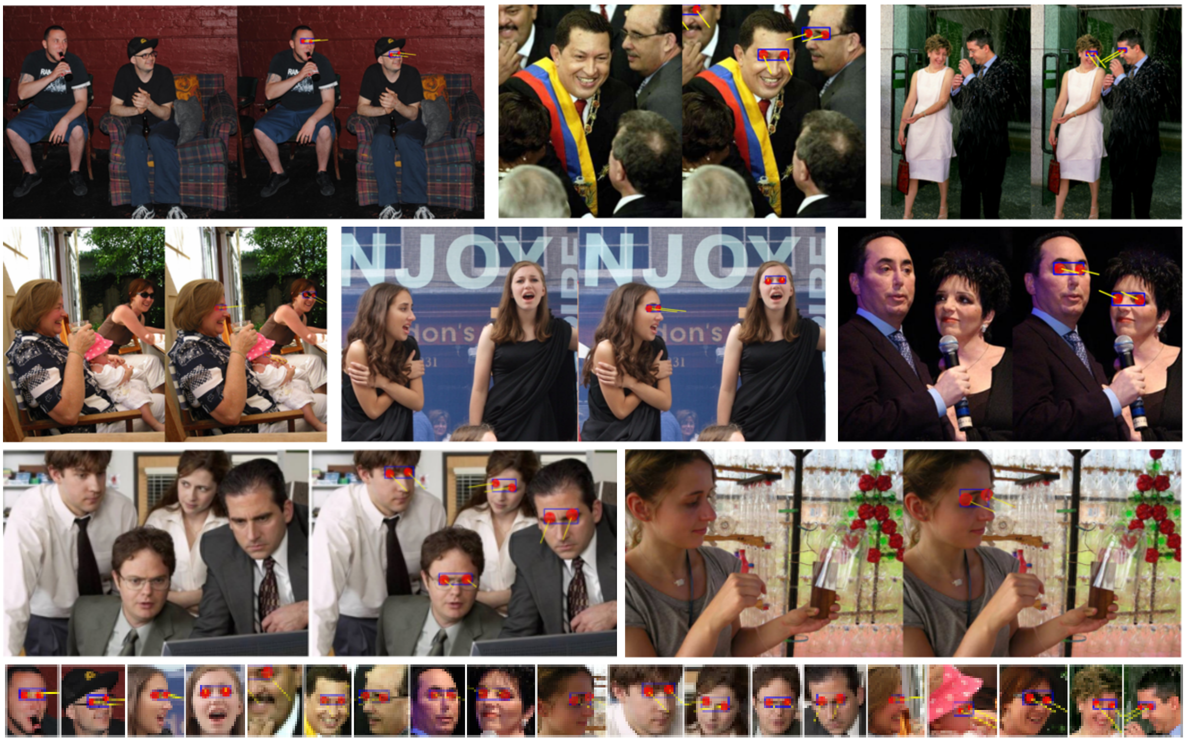


Figure 5.7: Visualisation of multi-face gaze estimation on AFW [304] and PASCAL [305]. Last row shows the face crops by zooming in the eye meshes. Our method estimates 3D gaze sa well as 3D eye meshes for all faces in an image in a single network pass.

not-LAEO) is correct. The evaluation metric is Average Precision (AP) computed as the area under the Precision-Recall (PR) curve. Here, we choose the open-source method, LEAO-Net++ [294], as our baseline. LAEO-Net++ is a three-branch track network, which takes two head tracks and the relative position between the two heads encoded by a head-map as the input. By fusing these temporal information, LAEO-Net++ determines a confidence score on whether the two people are looking at each other or not on each frame. We first run our single-shot gaze estimator on each frame of the AVA-LAEO dataset. Then, we train an MLP network taking eye meshes of two persons as input. Without using the temporal information, our gaze estimation can achieve an AP of 42.5% based on the frame-wise inference. As shown in Fig. 5.8, we visualise the eye mesh prediction results from our single-shot gaze estimator. Our method is robust under large-pose variations. When the eye region is totally not visible, the estimated gaze can be inaccurate as shown in the second case of Fig. 5.8. Besides the frame-wise training, we also exploit the gaze estimation results into the fusion block of LEAO-Net++. As given in Tab. 5.5, the gaze information can obviously improve the AP by 1.9%, confirming the

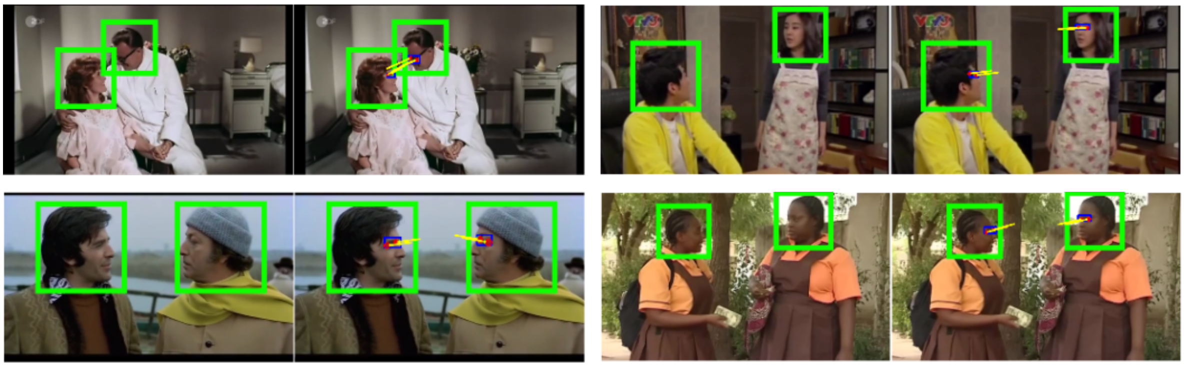


Figure 5.8: Visualisation results of our multi-face gaze estimation on the AVA-LAEO dataset. Our single pass gaze estimation method, trained with a large collection of “in-the-wild” and controlled images, is useful for improving detection accuracy of the LAEO task in real conditions.

effectiveness of the proposed single-shot gaze estimation in the wild.

5.3.6 Qualitative Results for Single-Face 3D Eye Reconstruction

Here we demonstrate a qualitative comparison of results retrieved from “in-the-wild” face images for which gaze labels are not available. We compare two versions of our model one with only ground truth supervision (GT-sup) and one with semi-supervision (Semi-sup). The first one is trained on ETH-XGaze [278] and Gaze360 [281] datasets combined until convergence. The second one is further trained on our ITWG dataset with weak supervision. The quantitative results on Fig. 5.9 demonstrate that weak supervision improves results on domains for which ground truth is not available.

5.3.7 3D Gaze Editing Application

To demonstrate the benefits of our method in a facial attribute editing context, we design a gaze redirection experiment in which we aim to train neural networks to manipulate 3D gaze direction in face images “in-the-wild”. To this end “in-the-wild” training data with gaze labels are required and thus, most gaze datasets are not suitable for the task as they are captured in controlled conditions or include limited environment and identity variation.

Therefore, in our experiment we employ arbitrary “in-the-wild” facial images and consider two training scenarios regarding the source of gaze labels used for supervision. In particular, in the first scenario we use pseudo-labels of gaze from “in-the-wild” images, extracted by the fully-supervised version of our model. In the contrary, in the second scenario we employ robust pseudo-labels which

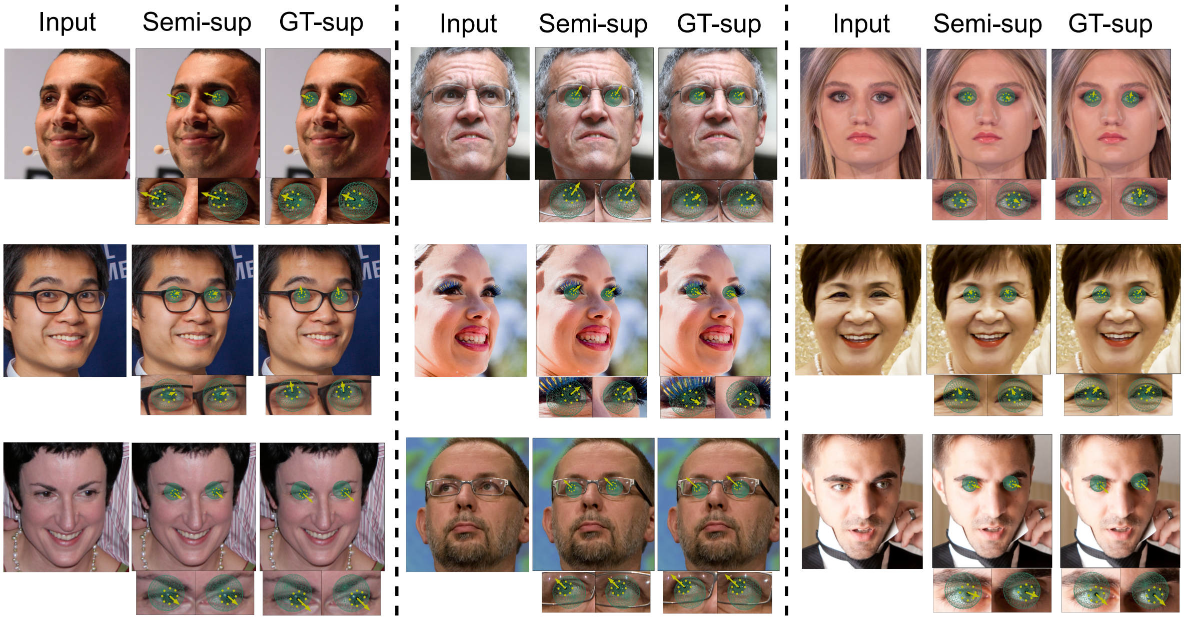


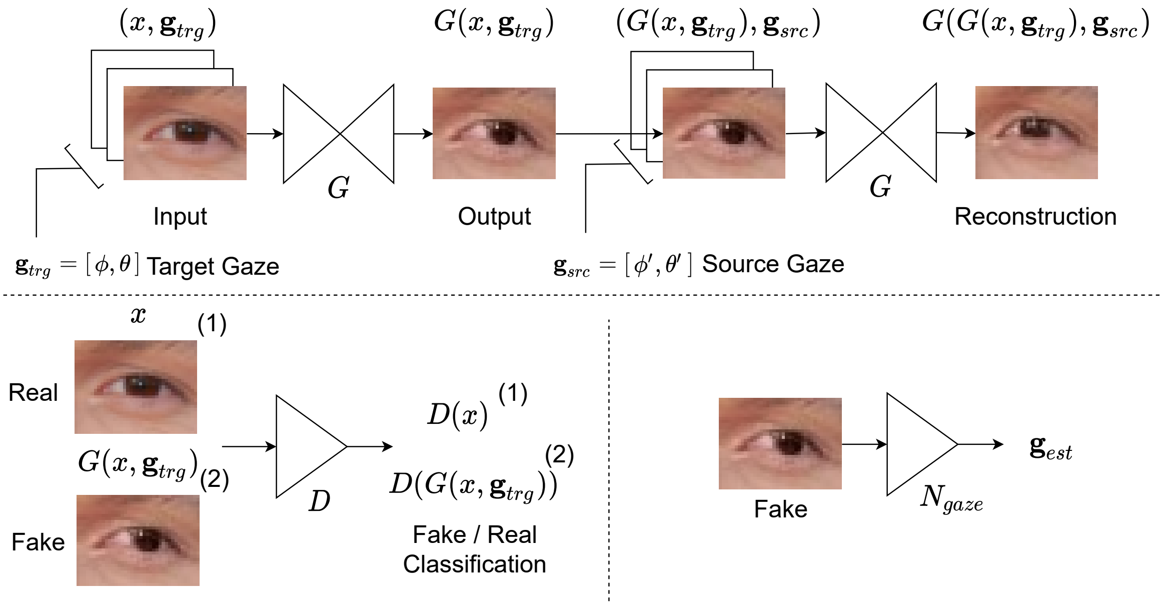
Figure 5.9: Results from applying our weakly-supervised and fully-supervised models on faces "in-the-wild". Employing semi-supervision from arbitrary face images improves the generalisation capabilities of gaze estimation.

we extract with our weakly-supervised model, which is trained without any gaze annotation. By these two settings, we aim to examine the validity of pseudo-labels extracted based on weak-supervision and evaluate their applicability to the gaze redirection task, when compared to labels extracted by traditional fully-supervised estimators.

To implement gaze redirection in images we follow the training paradigm of SliderGAN [29], which is based on the one of StarGAN [19], changing certain aspects of it, such as the conditioning signal, to adapt to the problem at hand. That is, given an input eye image x and a target gaze vector $\mathbf{g}_{trg}$, a generator network G produces image y which contains the desired 3D gaze direction. To ensure that, we employ our gaze estimation network N_{gaze} to obtain gaze labels $\mathbf{g}_{est} = N_{gaze}(y)$ from synthesised images and calculate a loss between the desired and estimated gaze vectors as:

$$\mathcal{L}_{gaze} = (180/\pi)\arccos(\mathbf{g}_{trg}^{\mathsf{T}}\mathbf{g}_{est}) \tag{5.8}$$

Following the standard adversarial training approach, we employ a discriminator network D to ensure that the generated images are photo-realistic. D is trained to distinguish real images from generated, driving the generator to produce images as close as possible to the real ones. To achieve



$(x, \mathbf{g}_{trg})$
$G(x, \mathbf{g}_{trg})$
$(G(x, \mathbf{g}_{trg}), \mathbf{g}_{src})$
$G(G(x, \mathbf{g}_{trg}), \mathbf{g}_{src})$
Input
G
Output
G
Reconstruction
$\mathbf{g}_{trg} = [\phi, \theta]$ Target Gaze
$\mathbf{g}_{src} = [\phi', \theta']$ Source Gaze
x (1)
Real
$G(x, \mathbf{g}_{trg})$ (2)
Fake
D
$D(x)$ (1)
(2) $D(G(x, \mathbf{g}_{trg}))$
Fake / Real Classification
Fake
N_{gaze}
$\mathbf{g}_{est}$

Figure 5.10: Gaze editing model based on the architecture of SliderGAN. An additional gaze estimation model is employed to make sure that the generated images adhere to the correct gaze inputs.

that we utilize the WGAN-GP loss [36] as:

$$\mathcal{L}_{adv} = \mathbb{E}_x[D(x)] - \mathbb{E}_{x,\mathbf{g}_{trg}}[D(G(x, \mathbf{g}_{trg}))] - \lambda_{gp}\mathbb{E}_y[(\|\nabla_x D(y)\|_2 - 1)^2]. \tag{5.9}$$

Moreover, to make sure that the eye identity is maintained between the input and generated images, we employ an image reconstruction loss between images x and $G(y, \mathbf{g}_{src})$, where $\mathbf{g}_{src}$ is the gaze label of image x according to the training set. The reconstruction loss is calculated for images of size $W \times H$ as:

$$\mathcal{L}_{rec} = \frac{1}{W \times H}\|x - G(y, \mathbf{g}_{src})\|_1, \tag{5.10}$$

Lastly, assuming λ_{rec} and λ_{gaze} are a reguralisation parameters, we optimise the following problems for the generator and discriminator respectively:

$$\min_G \mathcal{L}_{adv} + \lambda_{rec}\mathcal{L}_{rec} + \lambda_{gaze}\mathcal{L}_{gaze}, \tag{5.11}$$

$$\max_D \; - \mathcal{L}_{adv}. \tag{5.12}$$

A diagram of the model architecture, as well as the data flow during training is presented in Figure 5.10.

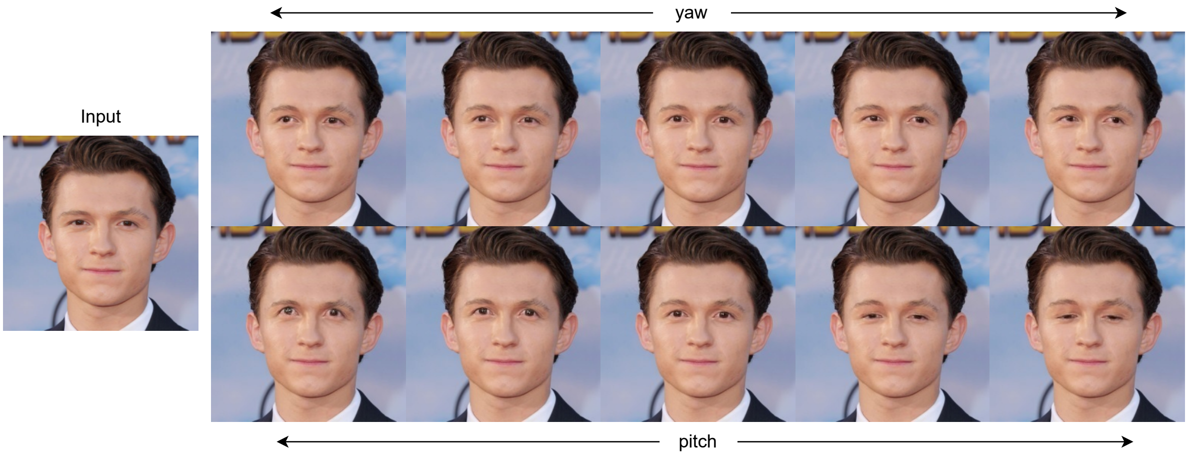


Figure 5.11: Continuous editing of gaze in pitch and yaw angles independently. The gaze editing generator has been trained with robust gaze labels from our weakly-supervised gaze estimation method.

To train the method described above, we employ a subset of 100K images of the VGGFace2 [291] dataset and extract gaze labels using two different methods. One is by applying our unified gaze estimation model, trained on gaze datasets using exclusively ground truth supervision. We name this model as M_{gt}. The other method is by applying our weakly-supervised training algorithm directly on the 100K images, based on synthetic views acquired by employing HeadGAN. We name the model trained with weak gaze labels as M_{weak}. We employ eye patches of size 64×64 and train only with left eyes, vertically flipping the right ones. In both cases, we train the models for 40 epochs, with batch size of 32, using the Adam optimiser [242] with $\beta_1 = 0.5, \beta_2 = 0.999$.

To validate our models we perform two different tasks, namely *continuous gaze redirection* and *gaze transfer between source and target images*. In Figure 5.11, we present synthetic images obtained by applying model M_{weak} to manipulate gaze across the pitch and yaw angles demonstrating the effectiveness of robust gaze labels coming from weakly-supervised models. In Figure 5.12, we present gaze transfer results between source and target images. Particularly, after applying models M_{gt} and M_{weak} on the tested samples, we obtain similar gaze transfer results, meaning that weak labels are not far off from labels obtained from fully-supervised models.

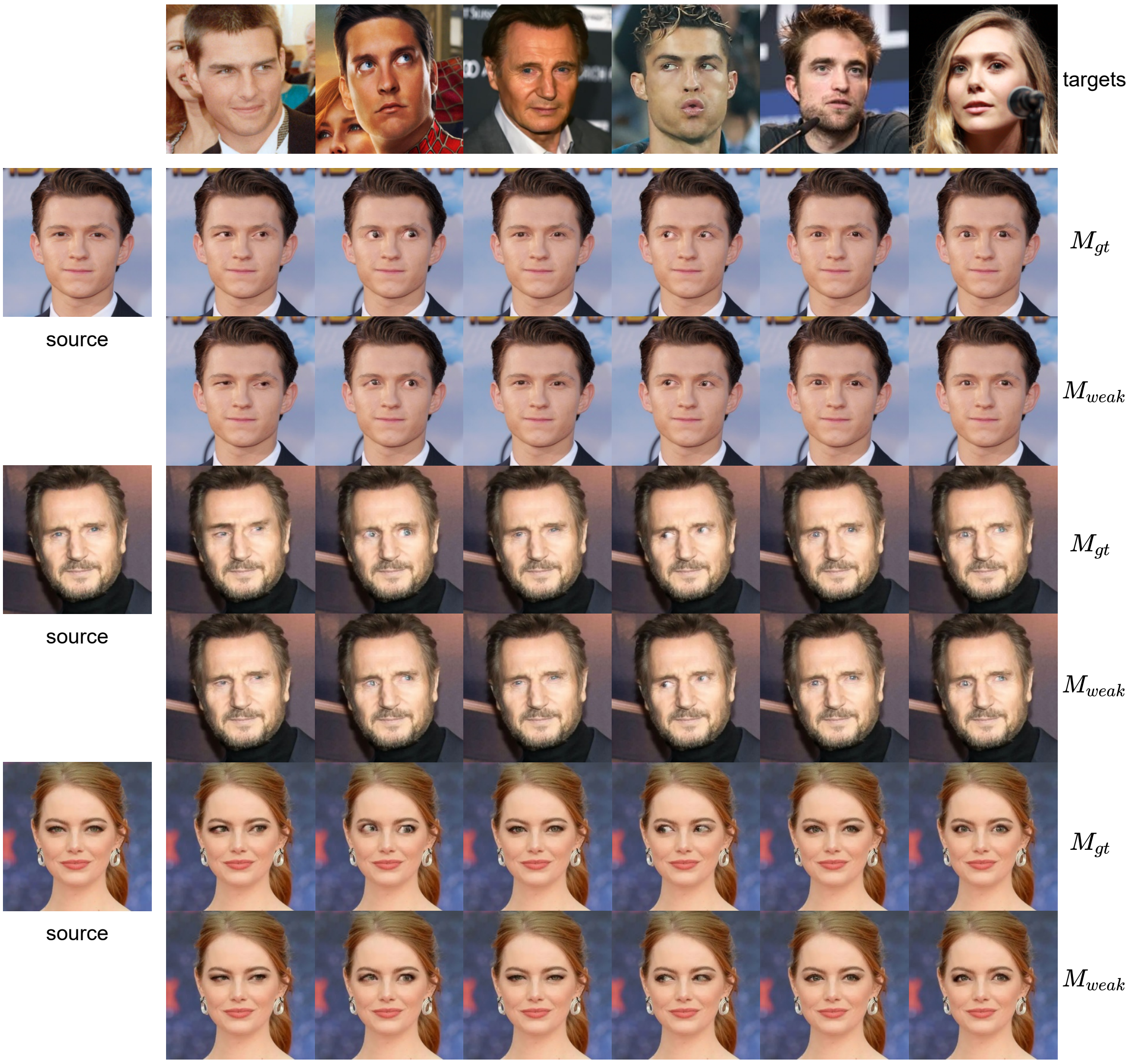


Figure 5.12: Gaze transfer on images "in-the-wild". The generator trained with robust labels from our weakly-supervised gaze estimation method (M_{weak}) performs similarly to the one that uses the ground truth-supervised model for extracting gaze labels (M_{gt}).

5.3.8 Limitations

In this section, we discuss limitations of our work, as well as possible solutions. One common challenge in 3D reconstruction systems is handling occlusions. In our case the biggest challenge comes from faces in profile pose and cases of people wearing glasses. Some examples of applying our model on such cases are depicted in Figure 5.13. While transparent glasses do not pose a significant challenge, sunglasses and profile faces make the eyes completely invisible, which causes our model's accuracy to drop. This is most probably due to inaccurate training data because of compromised performance

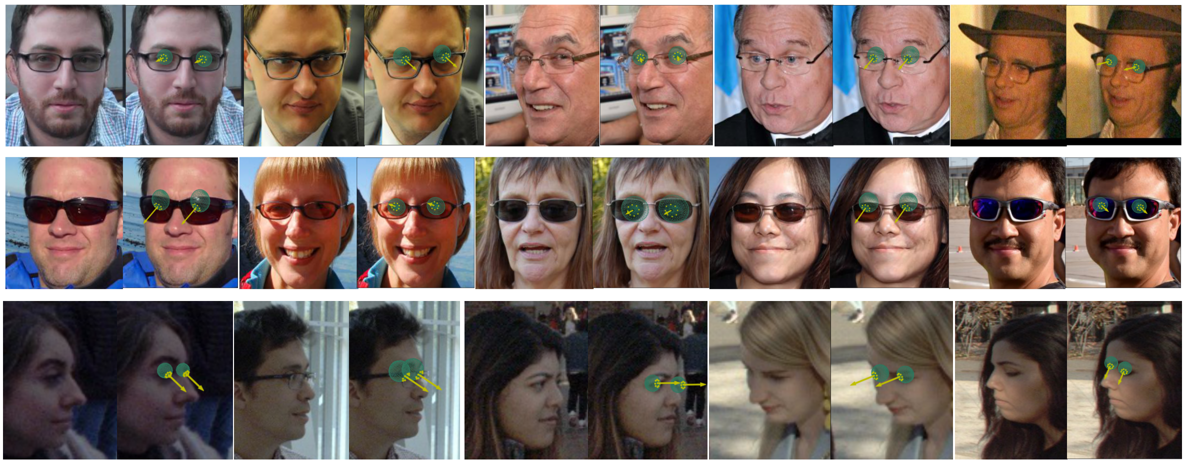


Figure 5.13: Results from applying our model on faces with glasses (1st row), sunglasses (2nd row) and faces in near-profile pose (3rd row). While normal glasses do not pose a significant challenge, sunglasses and profile poses are more difficult to handle.

of 2D iris localisation and face manipulation on such faces. Improving accuracy of those tasks could improve accuracy of our method, too. Additionally, profile images have only recently been included in gaze datasets [281, 278]. Having more ground truth cases of varying face poses would benefit any gaze estimation system including ours.

Another limitation of our method lies in the use of synthetic images for weakly-supervised training. For this algorithm, we assume that images of the same subject with different pose but the same difference between head and eye orientation are available. To acquire such data we employ a face reenactment method, HeadGAN [3], which animates the human head given single input images. However, relying on synthetic data for training means that performance is compromised by the quality of image generation. Higher quality of face image synthesis, could lead to easier optimisation and better performance for our method.

Lastly, another limitation of our model is that it does not consider the anatomical differences of eyes between people. In more detail, an offset angle exists between the optical and visual axes of eyes according to their real anatomy as shown in Fig. 5.14. This angle is subject-dependent and usually mentioned as the kappa coefficient of the eyes. Some methods have attempted to model this offset or incorporate it in their models' parameters [306, 259]. In our method, 3D gaze predictions are calculated by the orientation of the central axis of our 3D eyeball template, which coincides with the optical axis

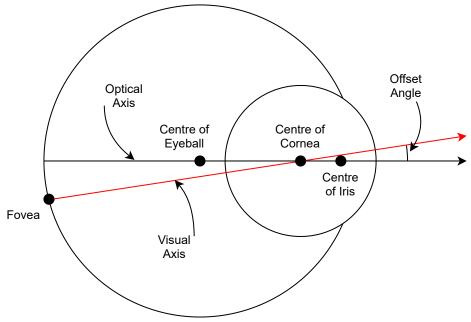


Figure 5.14: Eyeball anatomy demonstrating the offset between the optical and visual axis. This offset is often modeled by methods that solve controlled gaze estimation or train person-specific models. In our case, gaze generalisation to unseen and "in-the-wild" domains is not heavily affected by the 3^o possible offset, as gaze errors are much larger than that. The aim of our method is to provide a simple, yet effective way for robust gaze estimation that can be employed without any re-training or fine-tuning to real world scenarios.

of the human eyes. To make our system robust to variations of face identity, we rely on large "in-the-wild" face datasets. However, employing an anatomically aware 3D eyeball template or designing a strategy for personalising our model constitutes an interesting direction for further research.

5.4 Conclusion

In this Chapter, we presented a novel weakly-supervised method for gaze estimation, based on 3D eye mesh reconstruction. We demonstrated that by simply replacing the training target from few gaze parameters to dense 3D eye coordinates we can improve prediction accuracy. Additionally, for the first time, we explored the possibility of exploiting the abundantly available "in-the-wild" face data for improving gaze estimation generalisation. By enforcing specific multi-view geometric constraints, we have been able to successfully utilise such data and achieve improvements in cross-dataset and within-dataset experiments. Moreover, we proposed a method for single-shot, multi-face gaze estimation "in-the-wild", which we employed for predicting the task of Looking At Each Other (LAEO). Using this method, we demonstrates improvement in the AVA-LAEO dataset. Lastly, we demonstrated the effectiveness of weak gaze labels acquired by our weakly-supervised algorithm for the task of gaze editing in images "in-the-wild".

CHAPTER 6

Conclusion

Contents

6.1 Conclusion

In this Thesis, we presented novel techniques for analysis and editing of facial attributes in images, ranging from Component Analysis methods with solid mathematical formulations which result in meaningful outputs, to Deep Learning ones which are capable to address more complex problems and produce more realistic results regarding image generation and editing.

More specifically, in Chapter 3 we presented a CA method for recovering joint and individual variations from facial data in multiple scenarios. Our developed algorithm Robust Joint and Individual Variation Explained (RJIVE) and its variants, is capable of discovering joint and individual structures between an arbitrary number of datasets or views of the same dataset. Moreover, unlike previous methods, RJIVE can handle data contaminated by gross, sparse, non-Gaussian errors, such as the salt-and-pepper noise in imaging devices, occlusions in facial images, registration errors, or errors due incorrect localisation and tracking. Additionally, we presented a variant of RJIVE which is tailored for use with UV texture maps acquired by 3DMM fitting on arbitrary facial images. Through quantitative and qualitative experiments with both synthetic and real data, we demonstrated that our method outperforms the compared ones in 2D and 3D face age progression and expression transfer.

In Chapter 4, we presented an image-to-image translation method for face editing in images regarding motion due to expression and speech. Previous works were trained using images annotated for a specific number of discrete expressions or based on the Facial Action Coding System (FACS) which requires particular expertise. This caused their generation capabilities to be limited. Unlike those methods, we addressed face editing in images using continuous codes of expression, adapted from 3D blendshape modelling. Those codes are (a) very easily and with high accuracy recoverable from "in-the-wild" face images, (b) intuitive in the sense that their effect can be directly replicated by a 3D face model and (c) universal as any facial motion can be expressed through blendshapes (e.g. expression, speech). In our experiments, we demonstrated the usefulness of our technique in various expression editing and expression/speech transfer applications. Moreover, we showed that our method outperforms the compared ones through our quantitative and qualitative experiments.

In Chapter 5, we introduced a method to improve gaze estimation from monocular facial images "in-the-wild". In particular, unlike previous methods which predict gaze through few output parameters, we consider the 3D structure of eyes and propose to predict gaze via 3D eye reconstruction. Moreover, for the first time, to the best of our knowledge, we showed that it is possible to harness arbitrary, unlabelled face images to improve gaze estimation generalisation to unseen domains. To this end, we developed a weakly-supervised algorithm and designed particular multi-view, geometric losses to train our 3D eye reconstruction models. Through our experiments, we showed that our algorithm outperforms the compared ones in both within- and cross- domain gaze estimation, under various supervision settings. Lastly, we demonstrated the validity of weak gaze labels acquired by our methods from "in-the-wild" images, in two applications, namely, single-shot, multi-face gaze estimation and gaze direction editing in images.

6.2 Considerations for Future Work

The works presented in this Thesis focus on the problem of facial attribute editing, with the ones of Chapters 3 and 4 dealing with attributes such as facial expression and age, while the work of Chapter 5 is related to efficient gaze learning and editing. In all three works, there can be identified direct extensions and improvements that if implemented could lead to better performance or to a wider scope of applications. However, the significant breakthroughs which have been achieved recently in face image re-animation and synthetic media creation [2, 307, 3, 308], have pointed to new exciting research directions which are worth following.

As presented in Chapter 3, RJIVE is a CA method which discovers joint and individual structures in data annotated regarding a specific attribute. That is, for example, if a dataset of facial images is annotated with regards to the age of the subjects, RJIVE can be employed to recover age-specific (individual) components, as well as joint components including universal information about faces. However, one issue which arises from the formulation of RJIVE is that even if a dataset is annotated with regards to multiple attributes (for example, a dataset of faces might include labels about age, expression, identity, etc.), they cannot be simultaneously considered by the algorithm, meaning that in case we need to learn components for all available attributes, we would have to execute RJIVE multiple times. This is not only time-inefficient but also sub-optimal, in terms of learning from all available information during training. In [309], the authors have proposed a CA method to handle multiple attributes by learning generic and specific components for each attribute along with componentns for universal face structure. One drawback of [309] is that the generic components learned for each attribute are of rank 1. To overcome this issue, RJIVE could be extended to a multi-attribute setting and model both generic and specific components as linear subspaces of rank > 1 in order to include more information.

SliderGAN, introduced in Chapter 4, is an i2i translation method which employs blendshape parameters as codes to transfer images to target expressions. As shown by our experiments and particularly in Section 4.3.10 were we discus the limitations of the method, SliderGAN can be affected by mistakes in recovering blendshape parameters through 3DMM fitting. In particular, during 3DMM fitting identity and pose information can be mistakenly explained as expression, causing SliderGAN to over-fit to specific training examples. This effect is partially handled by incorporating synthetic data pairs in the training process, however, it is not completely eliminated. An extension of SliderGAN which could possibly overcome this issue, would be to utilise dense 3D information as condition of the generator, instead of few parameter values. That is, target 3D face shapes could be generated by combining the identity of test images and the target expression and be rendered appropriately before passed to the generator. Additionally, another possible extension direction would be to train the model to produce more types of animations, including free-head movement as shown in [3].

In Chapter 5, we presented a weakly-supervised method for monocular 3D gaze estimation. Even thought our method is able to extract gaze information from arbitrary, unlabelled face images and improve cross-domain generalisation, it is still restricted by the quality and accuracy of the synthetic images. In [268], the authors proposed to leverage datasets of social interaction to recover pseudo-labels about gaze. Indeed, discovering ways to supervise gaze estimation from abundantly available, real image and video data is an exciting line of research to follow. In particular, combining single-shot, multi-face gaze estimation with data regarding interaction between multiple subjects, or interaction between subjects and objects, could constitute a possible path worth exploring. Regarding eye editing in images, an obvious extension would be to train our model to edit additional attributes, such as eye colour, the size of the iris and pupil, and the overall shape of the eyelids. Of course, this can only be possible if available data become available or weakly-supervised techniques are devised, to extract related information from arbitrary face data.

Finally, in recent years, significant breakthroughs have been achieved in synthetic media creation using machine learning techniques. For example, StyleGAN [2] was among the first methods to have shown incredible results in unconditional face synthesis. In contrast to methods of Chapters 3 and 4, recent methods focus on full head animation, which is also referred to as re-enactment, and have shown remarkable results in synthetic image and video creation [199, 307, 310, 3, 308]. The high quality offered by these methods, have the potential to revolutionise numerous digital media fields including social media, teleconference systems, automatic image and video production, virtual/augmented reality and video games. This is directly reflected to the fact that multiple companies have recently emerged, which focus on synthetic media production such as images, video, speech and text, building on top of recent machine learning methods and developing new ones. Therefore, the current state-of-the-art clearly point to new exciting research directions to follow in facial attribute editing and face animation.

Bibliography

[1] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in Neural Information Processing Systems (NIPS)*, vol. 27, pp. 2672–2680, 2014. 20, 25, 47, 49, 52, 96, 102

[2] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 4401–4410, 2019. 20, 48, 49, 56, 98, 155, 156

[3] M. C. Doukas, S. Zafeiriou, and V. Sharmanska, "Headgan: One-shot neural head synthesis and editing," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 14398–14407, 2021. 20, 55, 56, 129, 132, 151, 155, 156

[4] R. Gottumukkal and V. K. Asari, "An improved face recognition technique based on modular pca approach," *Pattern Recognition Letters*, vol. 25, no. 4, pp. 429–436, 2004. 20

[5] C. Liu, "Gabor-based kernel pca with fractional power polynomial models for face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 26, no. 5, pp. 572–581, 2004. 20

[6] J. Yang, D. Zhang, A. F. Frangi, and J.-y. Yang, "Two-dimensional pca: a new approach to appearance-based face representation and recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 26, no. 1, pp. 131–137, 2004. 20

[7] M. Kaur, R. Vashisht, and N. Neeru, "Recognition of facial expressions with principal component analysis and singular value decomposition," *International Journal of Computer Applications*, vol. 9, no. 12, pp. 36–40, 2010. 20

[8] L. Oliveira, M. Mansano, A. Koerich, and A. de Souza Britto, "2d principal component analysis for face and facial-expression recognition," *Computing in Science & Engineering*, vol. 13, no. 3, pp. 9–13, 2010. 20

[9] S.-J. Wang, W.-J. Yan, G. Zhao, X. Fu, and C.-G. Zhou, "Micro-expression recognition using robust principal component analysis and local spatiotemporal directional features," in *Proceedings of European Conference on Computer Vision (ECCV)*, pp. 325–338, Springer, 2014. 20

[10] M. Scholz, F. Kaplan, C. L. Guy, J. Kopka, and J. Selbig, "Non-linear pca: a missing data approach," *Bioinformatics*, vol. 21, no. 20, pp. 3887–3895, 2005. 20

[11] S. J. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 1–8, IEEE, 2007. 20

[12] B. Gecer, S. Ploumpis, I. Kotsia, and S. Zafeiriou, "Ganfit: Generative adversarial network fitting for high fidelity 3d face reconstruction," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 1155–1164, 2019. 20, 33, 41, 133

[13] J. Booth, A. Roussos, E. Ververas, E. Antonakos, S. Ploumpis, Y. Panagakis, and S. Zafeiriou, "3d reconstruction of "in-the-wild" faces in images and videos," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 40, no. 11, pp. 2638–2652, 2018. 20, 25, 33, 36, 38, 39, 41, 95, 106, 107

[14] E. F. Lock, K. A. Hoadley, J. S. Marron, and A. B. Nobel, "Joint and individual variation explained (jive) for integrated analysis of multiple data types," *The annals of applied statistics*, vol. 7, no. 1, p. 523, 2013. 20, 25, 44, 45, 62, 67, 77, 78

[15] G. Zhou, A. Cichocki, Y. Zhang, and D. P. Mandic, "Group component analysis for multiblock data: Common and individual feature extraction," *IEEE Transactions on Neural Networks and Learning Systems (TNNLS)*, vol. 27, no. 11, pp. 2426–2439, 2015. 20, 25, 44, 45, 62, 77, 78

[16] Y. Panagakis, M. A. Nicolaou, S. Zafeiriou, and M. Pantic, "Robust correlated and individual component analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 38, no. 8, pp. 1665–1678, 2015. 20, 25, 44, 46, 62, 73, 77, 78

[17] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 1125–1134, 2017. 21, 25, 48, 49, 50, 51, 94, 97, 105

[18] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 2223–2232, 2017. 21, 25, 48, 52, 94, 97, 103, 105, 115, 116

[19] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "Stargan: Unified generative adversarial networks for multi-domain image-to-image translation," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 8789–8797, 2018. 21, 25, 48, 53, 54, 94, 97, 103, 105, 115, 116, 147

[20] A. Pumarola, A. Agudo, A. M. Martinez, A. Sanfeliu, and F. Moreno-Noguer, "Ganimation: Anatomically-aware facial animation from a single image," in *Proceedings of European Conference on Computer Vision (ECCV)*, pp. 818–833, 2018. 21, 25, 48, 94, 97, 100, 113, 115, 116, 117, 120, 121, 122

[21] K. Krafka, A. Khosla, P. Kellnhofer, H. Kannan, S. Bhandarkar, W. Matusik, and A. Torralba, "Eye tracking for everyone," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 2176–2184, 2016. 21, 128, 129, 138, 140

[22] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling, "It's written all over your face: Full-face appearance-based gaze estimation," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition, Workshops (CVPR-W)*, pp. 51–60, 2017. 21, 128

[23] S. Park, X. Zhang, A. Bulling, and O. Hilliges, "Learning to find eye region landmarks for remote gaze estimation in unconstrained settings," in *ACM Symposium on Eye Tracking Research and Applications (ETRA)*, pp. 1–10, 2018. 21, 128, 133, 136

[24] Y. Cheng, F. Lu, and X. Zhang, "Appearance-based gaze estimation via evaluation-guided asymmetric regression," in *Proceedings of European Conference on Computer Vision (ECCV)*, pp. 100–115, 2018. 21, 43, 128

[25] Y. Cheng, S. Huang, F. Wang, C. Qian, and F. Lu, "A coarse-to-fine adaptive network for appearance-based gaze estimation," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 34, pp. 10623–10630, 2020. 21, 128, 141

[26] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014. 25, 48, 49, 50

[27] C. Sagonas, E. Ververas, Y. Panagakis, and S. Zafeiriou, "Recovering joint and individual components in facial data," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 40, no. 11, pp. 2668–2681, 2018. 25, 44

[28] D. P. Bertsekas, *Constrained optimization and Lagrange multiplier methods*. Academic press, 2014. 25, 64, 67

[29] E. Ververas and S. Zafeiriou, “Slidergan: Synthesizing expressive face images by sliding 3d blendshape parameters,” *International Journal of Computer Vision (IJCV)*, vol. 128, no. 10, pp. 2629–2650, 2020. 25, 34, 41, 48, 147

[30] E. Richardson, M. Sela, R. Or-El, and R. Kimmel, “Learning detailed face reconstruction from a single image,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 1259–1268, 2017. 25, 33, 95

[31] A. Tewari, M. Zollhöfer, P. Garrido, F. Bernard, H. Kim, P. Pérez, and C. Theobalt, “Self-supervised multi-level face model learning for monocular reconstruction at over 250 hz,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 2549–2559, 2018. 25, 33, 95

[32] L. Tran and X. Liu, “Nonlinear 3d face morphable model,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 7346–7355, 2018. 25, 33, 95

[33] J. Booth, E. Antonakos, S. Ploumpis, G. Trigeorgis, Y. Panagakis, and S. Zafeiriou, “3d face morphable models” in-the-wild”,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 48–57, 2017. 25, 39, 85, 86

[34] P. Tzirakis, A. Papaioannou, A. Lattas, M. Tarasiou, B. Schuller, and S. Zafeiriou, “Synthesising 3d facial motion from “in-the-wild” speech,” in *Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition (FG)*, pp. 265–272, IEEE, 2020. 25, 95, 107, 119

[35] A. Jolicoeur-Martineau, “The relativistic discriminator: a key element missing from standard GAN,” in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019. 25, 96, 101, 102

[36] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, “Improved training of wasserstein gans,” *Advances in Neural Information Processing Systems (NIPS)*, vol. 30, pp. 5767–5777, 2017. 26, 49, 96, 102, 148

[37] E. Ververas, P. Gkagkos, J. Deng, J. Guo, M. C. Doukas, and S. Zafeiriou, "Generalizing gaze estimation with weak-supervision from synthetic views," *arXiv preprint arXiv:2212.02997*, 2022. 26

[38] Y. Li, K. Li, S. Jiang, Z. Zhang, C. Huang, and R. Y. Da Xu, "Geometry-driven self-supervised method for 3d human pose estimation," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 34, pp. 11442–11449, 2020. 26, 129, 131

[39] U. Iqbal, P. Molchanov, and J. Kautz, "Weakly-supervised 3d human pose learning via multi-view images in the wild," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 5243–5252, 2020. 26, 129, 131

[40] B. Wandt, M. Rudolph, P. Zell, H. Rhodin, and B. Rosenhahn, "Canonpose: Self-supervised monocular 3d human pose estimation in the wild," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 13294–13304, 2021. 26, 129, 131

[41] 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*, pp. 187–194, 1999. 32, 33, 38

[42] V. Blanz and T. Vetter, "Face recognition based on fitting a 3d morphable model," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 25, no. 9, pp. 1063–1074, 2003. 32, 33

[43] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter, "A 3d face model for pose and illumination invariant face recognition," in *IEEE international conference on advanced video and signal based surveillance*, pp. 296–301, IEEE, 2009. 32

[44] J. Booth, A. Roussos, A. Ponniah, D. Dunaway, and S. Zafeiriou, "Large scale 3d morphable models," *International Journal of Computer Vision (IJCV)*, vol. 126, no. 2, pp. 233–254, 2018. 32, 38, 43, 86

[45] H. Dai, N. Pears, W. A. Smith, and C. Duncan, "A 3d morphable model of craniofacial shape and texture variation," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 3085–3093, 2017. 32

[46] S. Ploumpis, E. Ververas, E. O'Sullivan, S. Moschoglou, H. Wang, N. Pears, W. A. Smith, B. Gecer, and S. Zafeiriou, "Towards a complete 3d morphable model of the human head," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 43, no. 11, pp. 4142–4160, 2020. 32

[47] T. Li, T. Bolkart, M. J. Black, H. Li, and J. Romero, "Learning a model of facial shape and expression from 4d scans.," *ACM Transactions on Graphics (TOG)*, vol. 36, no. 6, pp. 194–1, 2017. 33, 35

[48] A. Ranjan, T. Bolkart, S. Sanyal, and M. J. Black, "Generating 3d faces using convolutional mesh autoencoders," in *Proceedings of European Conference on Computer Vision (ECCV)*, pp. 704–720, 2018. 33

[49] 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," *ACM Transactions on Graphics (TOG)*, vol. 39, no. 5, pp. 1–38, 2020. 33

[50] L. Jiang, J. Zhang, B. Deng, H. Li, and L. Liu, "3d face reconstruction with geometry details from a single image," *IEEE Transactions on Image Processing (TIP)*, vol. 27, no. 10, pp. 4756–4770, 2018. 33

[51] X. Zhu, Z. Lei, X. Liu, H. Shi, and S. Z. Li, "Face alignment across large poses: A 3d solution," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 146–155, 2016. 33

[52] A. Tuan Tran, T. Hassner, I. Masi, and G. Medioni, "Regressing robust and discriminative 3d morphable models with a very deep neural network," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 5163–5172, 2017. 33

[53] S. Sanyal, T. Bolkart, H. Feng, and M. J. Black, "Learning to regress 3d face shape and expression from an image without 3d supervision," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 7763–7772, 2019. 33, 41

[54] K. Genova, F. Cole, A. Maschinot, A. Sarna, D. Vlasic, and W. T. Freeman, "Unsupervised training for 3d morphable model regression," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 8377–8386, 2018. 33, 41

[55] A. Tewari, M. Zollhoefer, F. Bernard, P. Garrido, H. Kim, P. Perez, and C. Theobalt, "High-fidelity monocular face reconstruction based on an unsupervised model-based face autoencoder," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 42, no. 2, pp. 357–370, 2018. 33

[56] A. Tewari, M. Zollhofer, H. Kim, P. Garrido, F. Bernard, P. Perez, and C. Theobalt, "Mofa: Model-based deep convolutional face autoencoder for unsupervised monocular reconstruction," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 1274–1283, 2017. 33

[57] D. Cosker, E. Krumhuber, and A. Hilton, "A facs valid 3d dynamic action unit database with applications to 3d dynamic morphable facial modeling," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 2296–2303, IEEE, 2011. 34

[58] C. Cao, Y. Weng, S. Zhou, Y. Tong, and K. Zhou, "Facewarehouse: A 3d facial expression database for visual computing," *IEEE Transactions on Visualization and Computer Graphics*, vol. 20, no. 3, pp. 413–425, 2013. 34, 86

[59] S. Cheng, I. Kotsia, M. Pantic, and S. Zafeiriou, "4dfab: A large scale 4d database for facial expression analysis and biometric applications," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 5117–5126, 2018. 34, 35, 40, 95, 106

[60] T. Neumann, K. Varanasi, S. Wenger, M. Wacker, M. Magnor, and C. Theobalt, "Sparse localized deformation components," *ACM Transactions on Graphics (TOG)*, vol. 32, no. 6, pp. 1–10, 2013. 34, 35

[61] S. J. Wright, R. D. Nowak, and M. A. Figueiredo, "Sparse reconstruction by separable approximation," *IEEE Transactions on Signal Processing*, vol. 57, no. 7, pp. 2479–2493, 2009. 35

[62] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski, "Optimization with sparsity-inducing penalties," *Foundations and Trends in Machine Learning*, vol. 4, pp. 1–106, Jan. 2012. 35

[63] F. I. Parke and K. Waters, "Computer facial animation," 2008. 35

[64] C.-H. Hjortsjö, "Man's face and mimic language," 1970. 35

[65] B. Choi, H. Eom, B. Mouscadet, S. Cullingford, K. Ma, S. Gassel, S. Kim, A. Moffat, M. Maier, M. Revelant, *et al.*, "Animatomy: an animator-centric, anatomically inspired system for 3d facial modeling, animation and transfer," in *SIGGRAPH Asia 2022 Conference Papers*, pp. 1–9, 2022. 35, 36

[66] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-squares fitting of two 3-d point sets," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, no. 5, pp. 698–700, 1987. 36

[67] P. J. Besl and N. D. McKay, "Method for registration of 3-d shapes," in *Sensor fusion IV: control paradigms and data structures*, vol. 1611, pp. 586–606, Spie, 1992. 36

[68] B. Amberg, S. Romdhani, and T. Vetter, "Optimal step nonrigid icp algorithms for surface registration," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 1–8, IEEE, 2007. 36

[69] S. Cheng, I. Marras, S. Zafeiriou, and M. Pantic, "Statistical non-rigid icp algorithm and its application to 3d face alignment," *Image and Vision Computing (IMAVIS)*, vol. 58, pp. 3–12, 2017. 36

[70] J. Booth, A. Roussos, S. Zafeiriou, A. Ponniah, and D. Dunaway, "A 3d morphable model learnt from 10,000 faces," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 5543–5552, 2016. 38, 106

[71] F. Shang, Y. Liu, J. Cheng, and H. Cheng, "Robust principal component analysis with missing data," in *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pp. 1149–1158, 2014. 38

[72] S. Baker and I. Matthews, "Lucas-kanade 20 years on: A unifying framework," *International Journal of Computer Vision (IJCV)*, vol. 56, no. 3, pp. 221–255, 2004. 39

[73] I. Matthews and S. Baker, "Active appearance models revisited," *International Journal of Computer Vision (IJCV)*, vol. 60, no. 2, pp. 135–164, 2004. 39, 79

[74] G. Papandreou and P. Maragos, “Adaptive and constrained algorithms for inverse compositional active appearance model fitting,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 1–8, IEEE, 2008. 39

[75] G. Tzimiropoulos and M. Pantic, “Optimization problems for fast aam fitting in-the-wild,” in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 593–600, 2013. 39

[76] E. Antonakos, J. Alabort-i Medina, G. Tzimiropoulos, and S. P. Zafeiriou, “Feature-based lucas–kanade and active appearance models,” *IEEE Transactions on Image Processing (TIP)*, vol. 24, no. 9, pp. 2617–2632, 2015. 39

[77] J. Alabort-i Medina and S. Zafeiriou, “A unified framework for compositional fitting of active appearance models,” *International Journal of Computer Vision (IJCV)*, vol. 121, no. 1, pp. 26–64, 2017. 39

[78] E. Antonakos, J. Alabort-i Medina, G. Tzimiropoulos, and S. Zafeiriou, “Hog active appearance models,” in *Proceedings of IEEE International Conference on Image Processing (ICIP)*, pp. 224–228, IEEE, 2014. 39

[79] P. Ekman and W. V. Friesen, “Facial action coding system,” *Environmental Psychology & Nonverbal Behavior*, 1978. 39, 94, 118

[80] P. Yang, Q. Liu, and D. N. Metaxas, “Boosting coded dynamic features for facial action units and facial expression recognition,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 1–6, IEEE, 2007. 39

[81] P. Lucey, J. F. Cohn, I. Matthews, S. Lucey, S. Sridharan, J. Howlett, and K. M. Prkachin, “Automatically detecting pain in video through facial action units,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 41, no. 3, pp. 664–674, 2010. 39

[82] Z. Zafar and N. A. Khan, “Pain intensity evaluation through facial action units,” in *Proceedings of International Conference on Pattern Recognition (ICPR)*, pp. 4696–4701, IEEE, 2014. 39

[83] A. Ruiz, J. Van de Weijer, and X. Binefa, "From emotions to action units with hidden and semi-hidden-task learning," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 3703–3711, 2015. 39

[84] D. Kollias and S. Zafeiriou, "A multi-task learning & generation framework: Valence-arousal, action units & primary expressions," *arXiv preprint arXiv:1811.07771*, 2018. 39

[85] B. Amos, B. Ludwiczuk, M. Satyanarayanan, *et al.*, "Openface: A general-purpose face recognition library with mobile applications," *CMU School of Computer Science*, vol. 6, no. 2, p. 20, 2016. 40, 95, 112

[86] C. F. Benitez-Quiroz, R. Srinivasan, and A. M. Martinez, "Discriminant functional learning of color features for the recognition of facial action units and their intensities," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 41, no. 12, pp. 2835–2845, 2018. 40, 95

[87] C. F. Benitez-Quiroz, Y. Wang, A. M. Martinez, *et al.*, "Recognition of action units in the wild with deep nets and a new global-local loss.," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 3990–3999, 2017. 40, 95

[88] T. Kanade, J. F. Cohn, and Y. Tian, "Comprehensive database for facial expression analysis," in *Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition (FG)*, pp. 46–53, IEEE, 2000. 40

[89] J. Booth and S. Zafeiriou, "Optimal uv spaces for facial morphable model construction," in *Proceedings of IEEE International Conference on Image Processing (ICIP)*, pp. 4672–4676, IEEE, 2014. 42

[90] K. Pearson, "Liii. on lines and planes of closest fit to systems of points in space," *The London, Edinburgh, and Dublin philosophical magazine and journal of science*, vol. 2, no. 11, pp. 559–572, 1901. 44

[91] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 586–587, IEEE Computer Society, 1991. 44

[92] H. Abdi and L. J. Williams, "Principal component analysis," *Wiley interdisciplinary reviews: computational statistics*, vol. 2, no. 4, pp. 433–459, 2010. 44

[93] H. Hotelling, "Relations between two sets of variates," in *Breakthroughs in statistics*, pp. 162–190, Springer, 1992. 44, 46, 47, 62

[94] L. R. Tucker, "An inter-battery method of factor analysis," *Psychometrika*, vol. 23, no. 2, pp. 111–136, 1958. 44, 62

[95] P. J. Huber, *Robust statistics*, vol. 523. John Wiley & Sons, 2004. 45, 63

[96] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?," *Journal of the ACM (JACM)*, vol. 58, no. 3, pp. 1–37, 2011. 46, 64, 66

[97] M. Fazel, *Matrix rank minimization with applications*. PhD thesis, PhD thesis, Stanford University, 2002. 46, 64

[98] D. L. Donoho, "For most large underdetermined systems of linear equations the minimal `[U+FFFD][U+FFFD]`1-norm solution is also the sparsest solution," *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, vol. 59, no. 6, pp. 797–829, 2006. 46, 64

[99] G. Perarnau, J. Van De Weijer, B. Raducanu, and J. M. Álvarez, "Invertible conditional gans for image editing," *arXiv preprint arXiv:1611.06355*, 2016. 48, 98, 115, 116

[100] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms," *arXiv preprint arXiv:1708.07747*, 2017. 49, 50

[101] V. Laschos, J. Tinapp, and K. Obermayer, "Training generative networks with general optimal transport distances," *arXiv preprint arXiv:1910.00535*, 2019. 49

[102] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015. 49

[103] D. Berthelot, T. Schumm, and L. Metz, "Began: Boundary equilibrium generative adversarial networks," *arXiv preprint arXiv:1703.10717*, 2017. 49

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

[105] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *International conference on machine learning (ICML)*, pp. 214–223, PMLR, 2017. 49, 96

[106] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. Paul Smolley, "Least squares generative adversarial networks," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 2794–2802, 2017. 49

[107] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241, Springer, 2015. 51

[108] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proceedings of European Conference on Computer Vision (ECCV)*, pp. 694–711, Springer, 2016. 52

[109] P. Garrido, L. Valgaerts, O. Rehmsen, T. Thormahlen, P. Perez, and C. Theobalt, "Automatic face reenactment," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 4217–4224, 2014. 55

[110] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Nießner, "Face2face: Real-time face capture and reenactment of rgb videos," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 2387–2395, 2016. 55, 99

[111] J. Thies, M. Zollhöfer, C. Theobalt, M. Stamminger, and M. Nießner, "Headon: Real-time reenactment of human portrait videos," *ACM Transactions on Graphics (TOG)*, vol. 37, no. 4, pp. 1–13, 2018. 55

[112] Y. Deng, J. Yang, D. Chen, F. Wen, and X. Tong, "Disentangled and controllable face image generation via 3d imitative-contrastive learning," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 5154–5163, 2020. 55, 56

[113] O. Fried, A. Tewari, M. Zollhöfer, A. Finkelstein, E. Shechtman, D. B. Goldman, K. Genova, Z. Jin, C. Theobalt, and M. Agrawala, "Text-based editing of talking-head video," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 4, pp. 1–14, 2019. 55, 99

[114] H. Kim, P. Garrido, A. Tewari, W. Xu, J. Thies, M. Niessner, P. Pérez, C. Richardt, M. Zollhöfer, and C. Theobalt, "Deep video portraits," *ACM Transactions on Graphics (TOG)*, vol. 37, no. 4, pp. 1–14, 2018. 55, 99

[115] Y. Ren, G. Li, Y. Chen, T. H. Li, and S. Liu, "Pirenderer: Controllable portrait image generation via semantic neural rendering," in *ICCV*, pp. 13759–13768, 2021. 55

[116] A. Tewari, M. Elgharib, G. Bharaj, F. Bernard, H.-P. Seidel, P. Pérez, M. Zollhofer, and C. Theobalt, "Stylerig: Rigging stylegan for 3d control over portrait images," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 6142–6151, 2020. 55, 56, 98

[117] J. Thies, M. Zollhöfer, and M. Nießner, "Deferred neural rendering: Image synthesis using neural textures," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 4, pp. 1–12, 2019. 55, 99

[118] S. Xu, J. Yang, D. Chen, F. Wen, Y. Deng, Y. Jia, and X. Tong, "Deep 3d portrait from a single image," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 7710–7720, 2020. 55

[119] E. Zakharov, A. Shysheya, E. Burkov, and V. Lempitsky, "Few-shot adversarial learning of realistic neural talking head models," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 9459–9468, 2019. 55

[120] Z. Chen, C. Wang, B. Yuan, and D. Tao, "Puppeteergan: Arbitrary portrait animation with semantic-aware appearance transformation," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 13518–13527, 2020. 55

[121] A. Siarohin, S. Lathuilière, S. Tulyakov, E. Ricci, and N. Sebe, "First order motion model for image animation," *NIPS*, vol. 32, 2019. 55, 59

[122] M. C. Doukas, E. Ververas, V. Sharmanska, and S. Zafeiriou, “Free-headgan: Neural talking head synthesis with explicit gaze control,” *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2023. 56

[123] R. Abdal, P. Zhu, N. J. Mitra, and P. Wonka, “Styleflow: Attribute-conditioned exploration of stylegan-generated images using conditional continuous normalizing flows,” *ACM Transactions on Graphics (TOG)*, vol. 40, no. 3, pp. 1–21, 2021. 56

[124] E. Härkönen, A. Hertzmann, J. Lehtinen, and S. Paris, “Ganspace: Discovering interpretable gan controls,” *Advances in Neural Information Processing Systems (NIPS)*, vol. 33, pp. 9841–9850, 2020. 56

[125] Y. Shen, J. Gu, X. Tang, and B. Zhou, “Interpreting the latent space of gans for semantic face editing,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 9243–9252, 2020. 56

[126] H. Ling, K. Kreis, D. Li, S. W. Kim, A. Torralba, and S. Fidler, “Editgan: High-precision semantic image editing,” *Advances in Neural Information Processing Systems (NIPS)*, vol. 34, pp. 16331–16345, 2021. 56

[127] J. Liu, Y. Zou, and D. Yang, “Semanticgan: Generative adversarial networks for semantic image to photo-realistic image translation,” pp. 2528–2532, 2020. 56

[128] T. Park, M.-Y. Liu, T.-C. Wang, and J.-Y. Zhu, “Semantic image synthesis with spatially-adaptive normalization,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 2337–2346, 2019. 56

[129] P. Zhu, R. Abdal, Y. Qin, and P. Wonka, “Sean: Image synthesis with semantic region-adaptive normalization,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 5104–5113, 2020. 56

[130] S.-Y. Chen, F.-L. Liu, Y.-K. Lai, P. L. Rosin, C. Li, H. Fu, and L. Gao, “Deepfaceediting: Deep generation of face images from sketches,” *ACM Transactions on Graphics (TOG)*, vol. 40, no. 4, p. 90, 2021. 56

[131] S.-Y. Chen, W. Su, L. Gao, S. Xia, and H. Fu, “Deepfacedrawing: Deep generation of face images from sketches,” *ACM Transactions on Graphics (TOG)*, vol. 39, no. 4, pp. 72–1, 2020. 56

[132] E. R. Chan, C. Z. Lin, M. A. Chan, K. Nagano, B. Pan, S. De Mello, O. Gallo, L. J. Guibas, J. Tremblay, S. Khamis, *et al.*, “Efficient geometry-aware 3d generative adversarial networks,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 16123–16133, 2022. 56

[133] Y. Bai, Y. Fan, X. Wang, Y. Zhang, J. Sun, C. Yuan, and Y. Shan, “High-fidelity facial avatar reconstruction from monocular video with generative priors,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 4541–4551, 2023. 56

[134] E. R. Chan, M. Monteiro, P. Kellnhofer, J. Wu, and G. Wetzstein, “pi-gan: Periodic implicit generative adversarial networks for 3d-aware image synthesis,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 5799–5809, 2021. 56

[135] J. Gu, L. Liu, P. Wang, and C. Theobalt, “Stylenerf: A style-based 3d-aware generator for high-resolution image synthesis,” *arXiv preprint arXiv:2110.08985*, 2021. 56

[136] R. Or-El, X. Luo, M. Shan, E. Shechtman, J. J. Park, and I. Kemelmacher-Shlizerman, “Stylesdf: High-resolution 3d-consistent image and geometry generation,” in *CVPR*, pp. 13503–13513, 2022. 56

[137] P. Zhou, L. Xie, B. Ni, and Q. Tian, “Cips-3d: A 3d-aware generator of gans based on conditionally-independent pixel synthesis,” *arXiv preprint arXiv:2110.09788*, 2021. 56

[138] A. Bergman, P. Kellnhofer, W. Yifan, E. Chan, D. Lindell, and G. Wetzstein, “Generative neural articulated radiance fields,” *Advances in Neural Information Processing Systems (NIPS)*, vol. 35, pp. 19900–19916, 2022. 57

[139] J.-g. Kwak, Y. Li, D. Yoon, D. Kim, D. Han, and H. Ko, “Injecting 3d perception of controllable nerf-gan into stylegan for editable portrait image synthesis,” in *Proceedings of European Conference on Computer Vision (ECCV)*, pp. 236–253, 2022. 57

[140] A. Noguchi, X. Sun, S. Lin, and T. Harada, "Unsupervised learning of efficient geometry-aware neural articulated representations," in *Proceedings of European Conference on Computer Vision (ECCV)*, pp. 597–614, 2022. 57

[141] J. Sun, X. Wang, Y. Shi, L. Wang, J. Wang, and Y. Liu, "Ide-3d: Interactive disentangled editing for high-resolution 3d-aware portrait synthesis," *TOG*, vol. 41, no. 6, pp. 1–10, 2022. 57

[142] J. Sun, X. Wang, Y. Zhang, X. Li, Q. Zhang, Y. Liu, and J. Wang, "Fenerf: Face editing in neural radiance fields," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 7672–7682, 2022. 57

[143] J. Tang, B. Zhang, B. Yang, T. Zhang, D. Chen, L. Ma, and F. Wen, "Explicitly controllable 3d-aware portrait generation," *arXiv preprint arXiv:2209.05434*, 2022. 57

[144] Y. Wu, Y. Deng, J. Yang, F. Wei, Q. Chen, and X. Tong, "Anifacegan: Animatable 3d-aware face image generation for video avatars," *arXiv preprint arXiv:2210.06465*, 2022. 57

[145] J. Zhang, A. Siarohin, Y. Liu, H. Tang, N. Sebe, and W. Wang, "Training and tuning generative neural radiance fields for attribute-conditional 3d-aware face generation," *arXiv preprint arXiv:2208.12550*, 2022. 57

[146] A. Chen, R. Liu, L. Xie, Z. Chen, H. Su, and J. Yu, "Sofgan: A portrait image generator with dynamic styling," *ACM Transactions on Graphics (TOG)*, vol. 41, no. 1, pp. 1–26, 2022. 57

[147] K. Jiang, S.-Y. Chen, F.-L. Liu, H. Fu, and L. Gao, "Nerffaceediting: Disentangled face editing in neural radiance fields," in *SIGGRAPH Asia 2022 Conference Papers*, pp. 1–9, 2022. 57

[148] J. Zhang, Z. Jiang, D. Yang, H. Xu, Y. Shi, G. Song, Z. Xu, X. Wang, and J. Feng, "Avatargen: a 3d generative model for animatable human avatars," *arXiv preprint arXiv:2211.14589*, 2022. 57

[149] J. Sun, X. Wang, L. Wang, X. Li, Y. Zhang, H. Zhang, and Y. Liu, "Next3d: Generative neural texture rasterization for 3d-aware head avatars," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 20991–21002, 2023. 57

[150] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, "Deepsdf: Learning continuous signed distance functions for shape representation," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 165–174, 2019. 57

[151] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, 2021. 57

[152] G. Gafni, J. Thies, M. Zollhofer, and M. Nießner, "Dynamic neural radiance fields for monocular 4d facial avatar reconstruction," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 8649–8658, 2021. 57, 58

[153] Z. Wang, T. Bagautdinov, S. Lombardi, T. Simon, J. Saragih, J. Hodgins, and M. Zollhofer, "Learning compositional radiance fields of dynamic human heads," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 5704–5713, 2021. 57

[154] L. Liu, M. Habermann, V. Rudnev, K. Sarkar, J. Gu, and C. Theobalt, "Neural actor: Neural free-view synthesis of human actors with pose control," *ACM Transactions on Graphics (TOG)*, vol. 40, no. 6, pp. 1–16, 2021. 57

[155] A. Noguchi, X. Sun, S. Lin, and T. Harada, "Neural articulated radiance field," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 5762–5772, 2021. 57

[156] K. Park, U. Sinha, J. T. Barron, S. Bouaziz, D. B. Goldman, S. M. Seitz, and R. Martin-Brualla, "Nerfies: Deformable neural radiance fields," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 5865–5874, 2021. 57

[157] K. Park, U. Sinha, P. Hedman, J. T. Barron, S. Bouaziz, D. B. Goldman, R. Martin-Brualla, and S. M. Seitz, "Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields," *arXiv preprint arXiv:2106.13228*, 2021. 57, 58

[158] A. Raj, M. Zollhofer, T. Simon, J. Saragih, S. Saito, J. Hays, and S. Lombardi, "Pixel-aligned volumetric avatars," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 11733–11742, 2021. 57, 58

[159] P.-W. Grassal, M. Prinzler, T. Leistner, C. Rother, M. Nießner, and J. Thies, "Neural head avatars from monocular rgb videos," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 18653–18664, 2022. 57, 58

[160] J. Zhang, X. Li, Z. Wan, C. Wang, and J. Liao, "Fdnerf: Few-shot dynamic neural radiance fields for face reconstruction and expression editing," in *SIGGRAPH Asia 2022 Conference Papers*, pp. 1–9, 2022. 57, 58

[161] Y. Zheng, V. F. Abrevaya, M. C. Bühler, X. Chen, M. J. Black, and O. Hilliges, "Im avatar: Implicit morphable head avatars from videos," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 13545–13555, 2022. 57, 58

[162] W. Zielonka, T. Bolkart, and J. Thies, "Instant volumetric head avatars," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 4574–4584, 2023. 57

[163] S. Athar, Z. Xu, K. Sunkavalli, E. Shechtman, and Z. Shu, "Rignerf: Fully controllable neural 3d portraits," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 20364–20373, 2022. 58

[164] Y. Guo, K. Chen, S. Liang, Y.-J. Liu, H. Bao, and J. Zhang, "Ad-nerf: Audio driven neural radiance fields for talking head synthesis," in *ICCV*, pp. 5784–5794, 2021. 58

[165] X. Gao, C. Zhong, J. Xiang, Y. Hong, Y. Guo, and J. Zhang, "Reconstructing personalized semantic facial nerf models from monocular video," *ACM Transactions on Graphics (TOG)*, vol. 41, no. 6, pp. 1–12, 2022. 58

[166] Y. Zheng, W. Yifan, G. Wetzstein, M. J. Black, and O. Hilliges, "Pointavatar: Deformable point-based head avatars from videos," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 21057–21067, 2023. 58

[167] H. Yu, K. Niinuma, and L. A. Jeni, "Confies: Controllable neural face avatars," in *Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition (FG)*, pp. 1–8, IEEE, 2023. 58

[168] K. Kania, K. M. Yi, M. Kowalski, T. Trzciński, and A. Tagliasacchi, "Conerf: Controllable neural radiance fields," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 18623–18632, 2022. 58

[169] Y. Hong, B. Peng, H. Xiao, L. Liu, and J. Zhang, "Headnerf: A real-time nerf-based parametric head model," in *CVPR*, pp. 20374–20384, 2022. 58

[170] C. Cao, T. Simon, J. K. Kim, G. Schwartz, M. Zollhoefer, S.-S. Saito, S. Lombardi, S.-E. Wei, D. Belko, S.-I. Yu, *et al.*, "Authentic volumetric avatars from a phone scan," *ACM Transactions on Graphics (TOG)*, vol. 41, no. 4, pp. 1–19, 2022. 58

[171] D. Wang, P. Chandran, G. Zoss, D. Bradley, and P. Gotardo, "Morf: Morphable radiance fields for multiview neural head modeling," in *ACM SIGGRAPH 2022 Conference Proceedings*, pp. 1–9, 2022. 58

[172] Y. Zhuang, H. Zhu, X. Sun, and X. Cao, "Mofanerf: Morphable facial neural radiance field," in *Proceedings of European Conference on Computer Vision (ECCV)*, pp. 268–285, 2022. 58

[173] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *International conference on machine learning (ICML)*, pp. 2256–2265, PMLR, 2015. 58

[174] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in Neural Information Processing Systems (NIPS)*, vol. 33, pp. 6840–6851, 2020. 58, 59

[175] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," *arXiv preprint arXiv:2011.13456*, 2020. 58, 59

[176] P. Vincent, "A connection between score matching and denoising autoencoders," *Neural computation*, vol. 23, no. 7, pp. 1661–1674, 2011. 59

[177] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," *arXiv preprint arXiv:2010.02502*, 2020. 59

[178] C. Meng, Y. He, Y. Song, J. Song, J. Wu, J.-Y. Zhu, and S. Ermon, "Sdedit: Guided image synthesis and editing with stochastic differential equations," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021. 59

[179] B. Kawar, S. Zada, O. Lang, O. Tov, H. Chang, T. Dekel, I. Mosseri, and M. Irani, "Imagic: Text-based real image editing with diffusion models," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 6007–6017, 2023. 59

[180] N. Ruiz, Y. Li, V. Jampani, Y. Pritch, M. Rubinstein, and K. Aberman, "Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 22500–22510, 2023. 59

[181] H. Chung, J. Kim, M. T. Mccann, M. L. Klasky, and J. C. Ye, "Diffusion posterior sampling for general noisy inverse problems," *arXiv preprint arXiv:2209.14687*, 2022. 59

[182] A. Lugmayr, M. Danelljan, A. Romero, F. Yu, R. Timofte, and L. Van Gool, "Repaint: Inpainting using denoising diffusion probabilistic models," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 11461–11471, 2022. 59

[183] W. Harvey, S. Naderiparizi, V. Masrani, C. Weilbach, and F. Wood, "Flexible diffusion modeling of long videos," *arXiv preprint arXiv:2205.11495*, 2022. 59

[184] J. Ho, W. Chan, C. Saharia, J. Whang, R. Gao, A. Gritsenko, D. P. Kingma, B. Poole, M. Norouzi, D. J. Fleet, *et al.*, "Imagen video: High definition video generation with diffusion models," *arXiv preprint arXiv:2210.02303*, 2022. 59

[185] C. Saharia, J. Ho, W. Chan, T. Salimans, D. J. Fleet, and M. Norouzi, "Image super-resolution via iterative refinement," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2022. 59

[186] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," *Advances in Neural Information Processing Systems (NIPS)*, vol. 34, pp. 8780–8794, 2021. 59

[187] A. Nichol, P. Dhariwal, A. Ramesh, P. Shyam, P. Mishkin, B. McGrew, I. Sutskever, and M. Chen, “Glide: Towards photorealistic image generation and editing with text-guided diffusion models,” *arXiv preprint arXiv:2112.10741*, 2021. 59

[188] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, *et al.*, “Learning transferable visual models from natural language supervision,” in *International conference on machine learning (ICML)*, pp. 8748–8763, PMLR, 2021. 59

[189] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, “Hierarchical text-conditional image generation with clip latents,” *arXiv preprint arXiv:2204.06125*, 2022. 59

[190] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, “High-resolution image synthesis with latent diffusion models,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 10684–10695, 2022. 59

[191] C. Mou, X. Wang, L. Xie, J. Zhang, Z. Qi, Y. Shan, and X. Qie, “T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models,” *arXiv preprint arXiv:2302.08453*, 2023. 59

[192] K. Preechakul, N. Chatthee, S. Wizadwongsa, and S. Suwajanakorn, “Diffusion autoencoders: Toward a meaningful and decodable representation,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 10619–10629, 2022. 59

[193] G. Kim, T. Kwon, and J. C. Ye, “Diffusionclip: Text-guided diffusion models for robust image manipulation,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 2426–2435, 2022. 59

[194] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. L. Denton, K. Ghasemipour, R. Gontijo Lopes, B. Karagol Ayan, T. Salimans, *et al.*, “Photorealistic text-to-image diffusion models with deep language understanding,” *Advances in Neural Information Processing Systems (NIPS)*, vol. 35, pp. 36479–36494, 2022. 59

[195] B. Zeng, X. Liu, S. Gao, B. Liu, H. Li, J. Liu, and B. Zhang, “Face animation with an attribute-guided diffusion model,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 628–637, 2023. 59

[196] M. Kim, F. Liu, A. Jain, and X. Liu, "Dcface: Synthetic face generation with dual condition diffusion model," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 12715–12725, 2023. 59

[197] S. Shen, W. Zhao, Z. Meng, W. Li, Z. Zhu, J. Zhou, and J. Lu, "Difftalk: Crafting diffusion models for generalized audio-driven portraits animation," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 1982–1991, 2023. 59

[198] Z. Huang, K. C. Chan, Y. Jiang, and Z. Liu, "Collaborative diffusion for multi-modal face generation and editing," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 6080–6090, 2023. 59

[199] T.-C. Wang, M.-Y. Liu, A. Tao, G. Liu, B. Catanzaro, and J. Kautz, "Few-shot video-to-video synthesis," in *Advances in Neural Information Processing Systems (NIPS)*, vol. 32, 2019. 59, 156

[200] I. Kemelmacher-Shlizerman, S. Suwajanakorn, and S. M. Seitz, "Illumination-aware age progression," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 3334–3341, 2014. 62, 84, 90

[201] X. Shu, J. Tang, H. Lai, L. Liu, and S. Yan, "Personalized age progression with aging dictionary," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 3970–3978, 2015. 62, 84

[202] C. Nhan Duong, K. Luu, K. Gia Quach, and T. D. Bui, "Longitudinal face modeling via temporal deep restricted boltzmann machines," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 5772–5780, 2016. 62, 84

[203] N. Ramanathan and R. Chellappa, "Modeling age progression in young faces," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, vol. 1, pp. 387–394, IEEE, 2006. 62, 84

[204] C.-T. Shen, W.-H. Lu, S.-W. Shih, and H.-Y. M. Liao, "Exemplar-based age progression prediction in children faces," in *IEEE International Symposium on Multimedia (ISM)*, pp. 123–128, IEEE, 2011. 62, 84

[205] W. Wang, Z. Cui, Y. Yan, J. Feng, S. Yan, X. Shu, and N. Sebe, "Recurrent face aging," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 2378–2386, 2016. 62, 85

[206] A. Klami and S. Kaski, "Probabilistic approach to detecting dependencies between data sets," *Neurocomputing*, vol. 72, no. 1-3, pp. 39–46, 2008. 62

[207] M. A. Nicolaou, V. Pavlovic, and M. Pantic, "Dynamic probabilistic cca for analysis of affective behavior and fusion of continuous annotations," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 36, no. 7, pp. 1299–1311, 2014. 62

[208] A. Klami, S. Virtanen, E. Leppäaho, and S. Kaski, "Group factor analysis," *IEEE Transactions on Neural Networks and Learning Systems (TNNLS)*, vol. 26, no. 9, pp. 2136–2147, 2015. 62

[209] L. Vandenberghe and S. Boyd, "Semidefinite programming," *SIAM review*, vol. 38, no. 1, pp. 49–95, 1996. 64

[210] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM journal on computing*, vol. 24, no. 2, pp. 227–234, 1995. 64

[211] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on Optimization*, vol. 20, no. 4, pp. 1956–1982, 2010. 64

[212] J. Wright and Y. Ma, "Dense error correction via ℓ_1-minimization," *IEEE Transactions on Information Theory*, vol. 56, no. 7, pp. 3540–3560, 2010. 70

[213] C. Georgakis, Y. Panagakis, and M. Pantic, "Dynamic behavior analysis via structured rank minimization," *International Journal of Computer Vision (IJCV)*, vol. 126, no. 2, pp. 333–357, 2018. 73

[214] C. Sagonas, Y. Panagakis, S. Zafeiriou, and M. Pantic, "Robust statistical frontalization of human and animal faces," *International Journal of Computer Vision (IJCV)*, vol. 122, no. 2, pp. 270–291, 2017. 73

[215] C. Sagonas, Y. Panagakis, S. Zafeiriou, and M. Pantic, "Raps: Robust and efficient automatic construction of person-specific deformable models," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 1789–1796, 2014. 73

[216] Y. Peng, A. Ganesh, J. Wright, W. Xu, and Y. Ma, "Rasl: Robust alignment by sparse and low-rank decomposition for linearly correlated images," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 34, no. 11, pp. 2233–2246, 2012. 73

[217] J. Alabort-i Medina, E. Antonakos, J. Booth, P. Snape, and S. Zafeiriou, "Menpo: A comprehensive platform for parametric image alignment and visual deformable models," in *Proceedings of the ACM International Conference on Multimedia, Open Source Software Competition*, pp. 679–682, 2014. 79

[218] C. Sagonas, E. Antonakos, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "300 faces in-the-wild challenge: Database and results," *Image and Vision Computing (IMAVIS)*, vol. 47, pp. 3–18, 2016. 79

[219] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker, "Multi-pie," *Image and Vision Computing (IMAVIS)*, vol. 28, no. 5, pp. 807–813, 2010. 79

[220] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, "The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition, Workshops (CVPR-W)*, pp. 94–101, IEEE, 2010. 79

[221] D. Huang and F. D. l. Torre, "Bilinear kernel reduced rank regression for facial expression synthesis," in *Proceedings of European Conference on Computer Vision (ECCV)*, pp. 364–377, Springer, 2010. 79

[222] A. Mollahosseini, B. Hasani, M. J. Salvador, H. Abdollahi, D. Chan, and M. H. Mahoor, "Facial expression recognition from world wild web," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition, Workshops (CVPR-W)*, pp. 58–65, 2016. 82

[223] P. Pérez, M. Gangnet, and A. Blake, "Poisson image editing," in *ACM Transactions on Graphics (TOG)*, vol. 22, pp. 313–318, ACM, 2003. 83

[224] B.-C. Chen, C.-S. Chen, and W. H. Hsu, "Cross-age reference coding for age-invariant face recognition and retrieval," in *Proceedings of European Conference on Computer Vision (ECCV)*, pp. 768–783, Springer, 2014. 83

[225] R. Rothe, R. Timofte, and L. Van Gool, “Dex: Deep expectation of apparent age from a single image,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition, Workshops (CVPR-W)*, pp. 10–15, 2015. 83

[226] A. Lanitis, C. J. Taylor, and T. F. Cootes, “Toward automatic simulation of aging effects on face images,” *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 24, no. 4, pp. 442–455, 2002. 84

[227] “Face transformer.” http://morph.cs.standrews.ac.uk/Transformer/. 85

[228] G. B. Huang and E. Learned-Miller, “Labeled faces in the wild: Updates and new reporting procedures,” *Dept. Comput. Sci., Univ. Massachusetts Amherst, Amherst, MA, USA, Tech. Rep*, vol. 14, no. 003, 2014. 88

[229] S. Suwajanakorn, S. M. Seitz, and I. Kemelmacher-Shlizerman, “Synthesizing obama: learning lip sync from audio,” *ACM Transactions on Graphics (TOG)*, vol. 36, no. 4, pp. 1–13, 2017. 94, 99

[230] O. Wiles, A. Koepke, and A. Zisserman, “X2face: A network for controlling face generation using images, audio, and pose codes,” in *Proceedings of European Conference on Computer Vision (ECCV)*, pp. 670–686, 2018. 94, 98

[231] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, and C. Change Loy, “Esrgan: Enhanced super-resolution generative adversarial networks,” in *Proceedings of European Conference on Computer Vision (ECCV), Workshops*, pp. 0–0, 2018. 96

[232] M. Li, W. Zuo, and D. Zhang, “Deep identity-aware transfer of facial attributes,” *arXiv preprint arXiv:1610.05586*, 2016. 97, 115, 116

[233] B. Usman, N. Dufour, K. Saenko, and C. Bregler, “Puppetgan: Cross-domain image manipulation by demonstration,” in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 9450–9458, 2019. 98

[234] A. Voynov and A. Babenko, “Unsupervised discovery of interpretable directions in the gan latent space,” in *International conference on machine learning (ICML)*, pp. 9786–9796, 2020. 98

[235] C. Tzelepis, G. Tzimiropoulos, and I. Patras, “Warpedganspace: Finding non-linear rbf paths in gan latent space,” in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 6393–6402, 2021. 98

[236] M. Georgopoulos, J. Oldfield, G. G. Chrysos, and Y. Panagakis, “Cluster-guided image synthesis with unconditional models,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 11543–11552, 2022. 98

[237] Z. Geng, C. Cao, and S. Tulyakov, “3d guided fine-grained face manipulation,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 9821–9830, 2019. 98

[238] Y. Alami Mejjati, C. Richardt, J. Tompkin, D. Cosker, and K. I. Kim, “Unsupervised attention-guided image-to-image translation,” *Advances in Neural Information Processing Systems (NIPS)*, vol. 31, 2018. 100

[239] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, “Arcface: Additive angular margin loss for deep face recognition,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 4690–4699, 2019. 104

[240] C. Fabian Benitez-Quiroz, R. Srinivasan, and A. M. Martinez, “Emotionet: An accurate, real-time algorithm for the automatic annotation of a million facial expressions in the wild,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 5562–5570, 2016. 106

[241] J. S. Chung and A. Zisserman, “Lip reading in the wild,” in *Proceedings of Asian Conference on Computer Vision (ACCV)*, pp. 87–103, Springer, 2016. 107, 119

[242] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014. 108, 139, 149

[243] L. Wang, Y. Zhang, and J. Feng, “On the euclidean distance of images,” *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 27, no. 8, pp. 1334–1339, 2005. 113, 121

[244] S. Li, W. Deng, and J. Du, “Reliable crowdsourcing and deep locality-preserving learning for expression recognition in the wild,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 2852–2861, 2017. 115, 117

[245] C. L. Kleinke, “Gaze and eye contact: a research review.,” *Psychological bulletin*, vol. 100, no. 1, p. 78, 1986. 128

[246] N. Castner, T. C. Kuebler, K. Scheiter, J. Richter, T. Eder, F. Hüttig, C. Keutel, and E. Kasneci, “Deep semantic gaze embedding and scanpath comparison for expertise classification during opt viewing,” in *ACM Symposium on Eye Tracking Research and Applications (ETRA)*, pp. 1–10, 2020. 128

[247] M. Vidal, J. Turner, A. Bulling, and H. Gellersen, “Wearable eye tracking for mental health monitoring,” *Computer Communications*, vol. 35, no. 11, pp. 1306–1311, 2012. 128

[248] S. Chandra, G. Sharma, S. Malhotra, D. Jha, and A. P. Mittal, “Eye tracking based human computer interaction: Applications and their uses,” in *2015 International Conference on Man and Machine Interfacing (MAMI)*, pp. 1–5, IEEE, 2015. 128

[249] S. Andrist, X. Z. Tan, M. Gleicher, and B. Mutlu, “Conversational gaze aversion for humanlike robots,” in *2014 9th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pp. 25–32, IEEE, 2014. 128

[250] M. Chen, Y. Jin, T. Goodall, X. Yu, and A. C. Bovik, “Study of 3d virtual reality picture quality,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 14, no. 1, pp. 89–102, 2019. 128

[251] R. Konrad, A. Angelopoulos, and G. Wetzstein, “Gaze-contingent ocular parallax rendering for virtual reality,” *ACM Transactions on Graphics (TOG)*, vol. 39, no. 2, pp. 1–12, 2020. 128

[252] A. Burova, J. Mäkelä, J. Hakulinen, T. Keskinen, H. Heinonen, S. Siltanen, and M. Turunen, “Utilizing vr and gaze tracking to develop ar solutions for industrial maintenance,” in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–13, 2020. 128

[253] A. Richard, C. Lea, S. Ma, J. Gall, F. De la Torre, and Y. Sheikh, "Audio- and gaze-driven facial animation of codec avatars," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 41–50, 2021. 128

[254] J. Zhang, J. Chen, H. Tang, W. Wang, Y. Yan, E. Sangineto, and N. Sebe, "Dual in-painting model for unsupervised gaze correction and animation in the wild," in *Proceedings of the ACM International Conference on Multimedia*, pp. 1588–1596, 2020. 128

[255] C. Ahlstrom, K. Kircher, and A. Kircher, "A gaze-based driver distraction warning system and its effect on visual behavior," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 2, pp. 965–973, 2013. 128

[256] A. G. Mavely, J. Judith, P. Sahal, and S. A. Kuruvilla, "Eye gaze tracking based driver monitoring system," in *IEEE International Conference on Circuits and Systems (ICCS)*, pp. 364–367, IEEE, 2017. 128

[257] G. Liu, Y. Yu, K. A. F. Mora, and J.-M. Odobez, "A differential approach for gaze estimation with calibration.," in *Proceedings of British Machine Vision Conference (BMVC)*, vol. 2, p. 6, 2018. 128

[258] K. Wang, R. Zhao, and Q. Ji, "A hierarchical generative model for eye image synthesis and eye gaze estimation," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 440–448, 2018. 128

[259] S. Park, E. Aksan, X. Zhang, and O. Hilliges, "Towards end-to-end video-based eye-tracking," in *Proceedings of European Conference on Computer Vision (ECCV)*, pp. 747–763, Springer, 2020. 128, 129, 151

[260] S. Park, S. D. Mello, P. Molchanov, U. Iqbal, O. Hilliges, and J. Kautz, "Few-shot adaptive gaze estimation," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 9368–9377, 2019. 128, 141

[261] Y. Yu, G. Liu, and J.-M. Odobez, "Improving few-shot user-specific gaze adaptation via gaze redirection synthesis," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 11937–11946, 2019. 128

[262] J. He, K. Pham, N. Valliappan, P. Xu, C. Roberts, D. Lagun, and V. Navalpakkam, “On-device few-shot personalization for real-time gaze estimation,” in *Proceedings of the IEEE International Conference on Computer Vision, Workshops (ICCV-W)*, pp. 0–0, 2019. 128

[263] Z. Guo, Z. Yuan, C. Zhang, W. Chi, Y. Ling, and S. Zhang, “Domain adaptation gaze estimation by embedding with prediction consistency,” in *Proceedings of Asian Conference on Computer Vision (ACCV)*, 2020. 128, 141

[264] K. Wang, R. Zhao, H. Su, and Q. Ji, “Generalizing eye tracking with bayesian adversarial learning,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 11907–11916, 2019. 128

[265] Y. Liu, R. Liu, H. Wang, and F. Lu, “Generalizing gaze estimation with outlier-guided collaborative adaptation,” in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 3835–3844, 2021. 128

[266] Y. Yu and J.-M. Odobez, “Unsupervised representation learning for gaze estimation,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 7314–7324, 2020. 128, 141

[267] Y. Sun, J. Zeng, S. Shan, and X. Chen, “Cross-encoder for unsupervised gaze representation learning,” in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 3702–3711, 2021. 128, 141

[268] R. Kothari, S. De Mello, U. Iqbal, W. Byeon, S. Park, and J. Kautz, “Weakly-supervised physically unconstrained gaze estimation,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 9980–9989, 2021. 128, 141, 142, 143, 156

[269] E. Wood, T. Baltrušaitis, L.-P. Morency, P. Robinson, and A. Bulling, “A 3d morphable eye region model for gaze estimation,” in *Proceedings of European Conference on Computer Vision (ECCV)*, pp. 297–313, Springer, 2016. 128

[270] Y. Yu, G. Liu, and J.-M. Odobez, “Deep multitask gaze estimation with a constrained landmark-gaze model,” in *Proceedings of European Conference on Computer Vision (ECCV), Workshops*, pp. 0–0, 2018. 128

[271] S. Park, A. Spurr, and O. Hilliges, “Deep pictorial gaze estimation,” in *Proceedings of European Conference on Computer Vision (ECCV)*, pp. 721–738, 2018. 128, 141

[272] J. Deng, J. Guo, E. Ververas, I. Kotsia, and S. Zafeiriou, “Retinaface: Single-shot multi-level face localisation in the wild,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 5203–5212, 2020. 129, 131, 133, 136, 139, 141, 144

[273] D. Kulon, R. A. Guler, I. Kokkinos, M. M. Bronstein, and S. Zafeiriou, “Weakly-supervised mesh-convolutional hand reconstruction in the wild,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 4990–5000, 2020. 129, 131

[274] R. A. Guler, N. Neverova, and I. Kokkinos, “Densepose: Dense human pose estimation in the wild,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 7297–7306, 2018. 129, 131

[275] R. A. Guler and I. Kokkinos, “Holopose: Holistic 3d human reconstruction in-the-wild,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 10884–10894, 2019. 129, 131

[276] W. Fuhl, G. Kasneci, and E. Kasneci, “Teyed: Over 20 million real-world eye images with pupil, eyelid, and iris 2d and 3d segmentations, 2d and 3d landmarks, 3d eyeball, gaze vector, and eye movement types,” in *IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 367–375, IEEE, 2021. 129

[277] B. A. Smith, Q. Yin, S. K. Feiner, and S. K. Nayar, “Gaze locking: passive eye contact detection for human-object interaction,” in *Proceedings of the ACM symposium on User interface software and technology*, pp. 271–280, 2013. 129, 138, 140

[278] X. Zhang, S. Park, T. Beeler, D. Bradley, S. Tang, and O. Hilliges, “Eth-xgaze: A large scale dataset for gaze estimation under extreme head pose and gaze variation,” in *Proceedings of European Conference on Computer Vision (ECCV)*, pp. 365–381, Springer, 2020. 129, 138, 142, 146, 151

[279] K. A. Funes Mora, F. Monay, and J.-M. Odobez, "Eyediap: A database for the development and evaluation of gaze estimation algorithms from rgb and rgb-d cameras," in *ACM Symposium on Eye Tracking Research and Applications (ETRA)*, pp. 255–258, 2014. 129

[280] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling, "Appearance-based gaze estimation in the wild," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 4511–4520, 2015. 129, 138, 140

[281] P. Kellnhofer, A. Recasens, S. Stent, W. Matusik, and A. Torralba, "Gaze360: Physically unconstrained gaze estimation in the wild," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 6912–6921, 2019. 129, 130, 138, 140, 141, 142, 146, 151

[282] Y. Sugano, Y. Matsushita, and Y. Sato, "Learning-by-synthesis for appearance-based 3d gaze estimation," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 1821–1828, 2014. 129, 138, 140

[283] T. Fischer, H. J. Chang, and Y. Demiris, "Rt-gene: Real-time eye gaze estimation in natural environments," in *Proceedings of European Conference on Computer Vision (ECCV)*, pp. 334–352, 2018. 129

[284] Y. Deng, J. Yang, S. Xu, D. Chen, Y. Jia, and X. Tong, "Accurate 3d face reconstruction with weakly-supervised learning: From single image to image set," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition, Workshops (CVPR-W)*, pp. 0–0, 2019. 129

[285] Y. Cai, L. Ge, J. Cai, and J. Yuan, "Weakly-supervised 3d hand pose estimation from monocular rgb images," in *Proceedings of European Conference on Computer Vision (ECCV)*, pp. 666–682, 2018. 129

[286] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 770–778, 2016. 133, 136

[287] V. Albiero, X. Chen, X. Yin, G. Pang, and T. Hassner, "img2pose: Face alignment and detection via 6dof, face pose estimation," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 7617–7627, 2021. 133

[288] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 8759–8768, 2018. 137

[289] R. Girshick, "Fast r-cnn," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 1440–1448, 2015. 137

[290] S. Yang, P. Luo, C.-C. Loy, and X. Tang, "Wider face: A face detection benchmark," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 5525–5533, 2016. 137, 139

[291] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, "Vggface2: A dataset for recognising faces across pose and age," in *Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition (FG)*, pp. 67–74, IEEE, 2018. 138, 149

[292] C. Gu, C. Sun, D. A. Ross, C. Vondrick, C. Pantofaru, Y. Li, S. Vijayanarasimhan, G. Toderici, S. Ricco, R. Sukthankar, *et al.*, "Ava: A video dataset of spatio-temporally localized atomic visual actions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6047–6056, 2018. 138

[293] M. J. Marin-Jimenez, V. Kalogeiton, P. Medina-Suarez, and A. Zisserman, "Laeo-net: revisiting people looking at each other in videos," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 3477–3485, 2019. 138, 144

[294] M. J. Marin-Jimenez, V. Kalogeiton, P. Medina-Suarez, and A. Zisserman, "LAEO-Net++: revisiting people Looking At Each Other in videos," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2021. 138, 144, 145

[295] H. Joo, H. Liu, L. Tan, L. Gui, B. Nabbe, I. Matthews, T. Kanade, S. Nobuhara, and Y. Sheikh, "Panoptic studio: A massively multiview system for social motion capture," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 3334–3342, 2015. 139

[296] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 248–255, IEEE, 2009. 139

[297] K. Chen, J. Wang, J. Pang, Y. Cao, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Xu, *et al.*, "Mmdetection: Open mmlab detection toolbox and benchmark," *arXiv preprint arXiv:1906.07155*, 2019. 139

[298] J. Guo, J. Deng, A. Lattas, and S. Zafeiriou, "Sample and computation redistribution for efficient face detection," *arXiv preprint arXiv:2105.04714*, 2021. 139

[299] S. Zhang, C. Chi, Y. Yao, Z. Lei, and S. Z. Li, "Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 9759–9768, 2020. 139

[300] Y. Wu and K. He, "Group normalization," in *Proceedings of European Conference on Computer Vision (ECCV)*, pp. 3–19, 2018. 139

[301] X. Li, W. Wang, L. Wu, S. Chen, X. Hu, J. Li, J. Tang, and J. Yang, "Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection," *Advances in Neural Information Processing Systems (NIPS)*, vol. 33, pp. 21002–21012, 2020. 139

[302] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-iou loss: Faster and better learning for bounding box regression," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 34, pp. 12993–13000, 2020. 139

[303] Z. Wang, J. Zhao, C. Lu, F. Yang, H. Huang, Y. Guo, *et al.*, "Learning to detect head movement in unconstrained remote gaze estimation in the wild," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 3443–3452, 2020. 141

[304] X. Zhu and D. Ramanan, "Face detection, pose estimation, and landmark localization in the wild," in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 2879–2886, IEEE, 2012. 144, 145

[305] M. Mathias, R. Benenson, M. Pedersoli, and L. V. Gool, "Face detection without bells and whistles," in *Proceedings of European Conference on Computer Vision (ECCV)*, pp. 720–735, Springer, 2014. 144, 145

[306] Z. Chen and B. Shi, “Offset calibration for appearance-based gaze estimation via gaze decomposition,” in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 270–279, 2020. 151

[307] S. Ha, M. Kersner, B. Kim, S. Seo, and D. Kim, “Marionette: Few-shot face reenactment preserving identity of unseen targets,” in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 34, pp. 10893–10900, 2020. 155, 156

[308] T.-C. Wang, A. Mallya, and M.-Y. Liu, “One-shot free-view neural talking-head synthesis for video conferencing,” in *Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition (CVPR)*, pp. 10039–10049, 2021. 155, 156

[309] S. Moschoglou, E. Ververas, Y. Panagakis, M. A. Nicolaou, and S. Zafeiriou, “Multi-attribute robust component analysis for facial uv maps,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 6, pp. 1324–1337, 2018. 155

[310] M. R. Koujan, M. C. Doukas, A. Roussos, and S. Zafeiriou, “Head2head: Video-based neural head synthesis,” in *Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition (FG)*, pp. 16–23, IEEE, 2020. 156