Application of Synthetic Images in Face Alignment and Face 3D Reconstruction Studies

March 2022

Graduate School of Systems Engineering
Wakayama University

Haoqi Gao

顔の特徴点検出と **3** 次元復元における合成画像の利用

令和 4 年 3 月

和歌山大学大学院システム工学研究科

高皓琪

Abstract

This study shows the great potential of synthetic data in reducing the various biases impact of real datasets. In particular, we explored the complementary and availability of synthetic face images for the face alignment and the 3D reconstruction tasks.

First, the synthetic face images generated by the FaceGen generator, which can easily modify features(pose and expression) of synthetic face images. The multiple datasets S/R/RA/R+S/RA+S were created by combining the real datasets (R), real datasets after augmentation (RA), and synthetic datasets (S) and using them to train a facial landmark point detector in Chapter 3. Experimental results demonstrate that adding synthetic datasets improve the accuracy. It shows that FaceGen can generate face images with various poses and expressions, which is helpful for training.

Second, to address the differences between real-world and synthetic datasets. Inspired by GAN ideas, this paper introduced two transformed models ($R \to S$ and $S \to R$). In Chapter 4, an improved transfer $R \to S$ model based on the CycleGAN is proposed. It can convert the real test dataset to synthetic first and use the face alignment model trained with RA+S to make predictions. However, the improvement is limited for faces in complex backgrounds or under large poses and expressions. In Chapter 5, another improved transfer $S \to R$ model based on UGATIT is proposed. It can generate a synthetic face dataset G with a near-realistic appearance, various poses, and expressions. Then, we train a landmark point detector using the RA+G dataset and show experimentally that the accuracy is improved. This approach can alleviate the current challenging issues in facial image analysis regarding the data collection process.

3D face reconstruction needs more complex images, which is not satisfied for the FaceGen model. In Chapter 6, the synthetic dataset is used, which is rotating the real-world datasets in 3D space with variations in large poses and occlusion. An improved lightweight structure combining three achievements: 3DMM, ShufffleNetV2 Plus series of units and Squeeze-and excitation (SE) attention mechanism is proposed to estimate the intrinsic consistency between the predicted 3DMM coefficients and the corresponding face images. Experimental results illustrate the proposed algorithm's robustness even under multiple challenging conditions and lighten the network in size and speed while maintaining as much accuracy as possible.

概要

本研究では, 実データに加えて合成データを用いることによって, 実データセットに含まれる様々な偏りを解消できることを示す. 特に, 顔の特徴点検出タスクと 3 次元復元タスクを例にとり, 合成画像データの補完性および利用可能性に関して以下に述べる研究を行う.

まず, 合成顔画像データセットに含まれる顔の姿勢や表情などの特徴を簡単に変更してデータセットを定量的に評価できるようにするため, 顔画像生成器である FaceGen を用いて合成顔画像を生成する. 第 3 章では, 実画像データセット R, データ拡張後の実画像データセット RA，合成画像データセット S を組み合わせて複数のデータセット S/R/RA/R+S/RA+S を作成し, これらを用いてニューラルネットに基づく顔の特徴点検出器を学習する. そして, 合成画像を追加することによって特徴点の検出精度が向上することを実験によって示す. このことは, 顔画像生成器が学習に有用な多様な姿勢や表情の顔画像を生成することが可能であることを示している.

次に, 実画像データセットと合成画像データセットの間の差異を解消するため, GAN の考え方に基づき, 実画像を合成画像に変換するモデル $R \to S$ と合成画像を実画像に変換するモデル $S \to R$ を提案する. 第 4 章では, CycleGAN に基づく変換手法であるモデル $R \to S$ を提案し, これを用いて評価用の実画像データを合成画像データに変換する. そして, これを入力として, データセット RA+S を用いて学習した特徴点検出器の検出精度を評価する. しかし, 顔画像の背景が複雑な場合や顔の向きや表情が大きく変化する場合には精度の改善に限界がある. そこで, 第 5 章では, UGATIT に基づく変換手法であるモデル $S \to R$ を提案し, これを用いて実画像に近い見え, 多様な姿勢, 表情を有する合成顔画像データセット G を生成する. そして, データセット RA+G を用いて特徴点検出器を学習し, 実験により特徴点の検出精度が向上することを示す. 本手法は, 顔画像解析におけるデータ収集を容易にするものである.

顔の 3 次元復元タスクでは, FaceGen では生成することが難しいより複雑かつ大規模な画像データセットが必要になる. そこで, 第 6 章では, 3 次元実データを 2 次元平面に投影することによって様々な顔の向きや隠蔽度合いを網羅する合成画像データセットを生成して利用する. また, 3DMM, ShuffleNetV2 Plus および Squeeze-and-Excitation (SE) による注意機構の 3 つの手法を組み合わせたニューラルネットに基づく軽量な 3 次元復元器を提案し, 顔の特徴点検出および 3 次元復元実験において提案手法の頑健性を示す. 提案手法は, 高い精度を維持しながら, ネットワークモデルの計算量と必要メモリ量を削減することが可能である.

Acknowledgements

Twenty-three years of schooling from a child to an adult, left home step by step. The memory of my academic life and the experience of my doctoral studies will bury in my deepest heart.

First of all, I would like to express my deepest gratitude to Prof. Koichi Ogawara, who has taught me a lot in the past few years, and the successful completion of my dissertation is inseparable from my teacher's guidance. I want to thank Prof. Ogawara for his patience in guiding me in my studies and taking care of me in my daily life.

Second, I would like to thank Prof. Haiyuan Wu, Prof. Qian Chen for the many opportunities they have given me. Thanks to all my friends in the Ph.D. stage (like Dr.Yiqiang Qi, Dr. Peng Li, Dr.Yi Tian, Dr.Xinbo Ren, Dr.Yankun Lang, Mr.Xiang Zhou, Mr.Kai Wang, Mrs.Xiaosi Hu) who helped me a lot in life and study during these years. I need to thank Prof. Liangzhi Li and Prof. Hajime Nagahara of Osaka University for their help and inspiration during the writing paper. I want to thank my friend in China, Mr.Chong Xu, for solving my code problems during my hard time. Many thankfulness to my master's supervisor Prof. Huafeng Wang, who always cares about my study and life in Japan. I also thank all Japanese students in the same lab for their care and help in research and life. I truly wish all my friends and professors peace and happiness in their future lives.

All my love and thanks to my parents. Maybe words can't describe how thankful I am. Thank them for creating a warm and loving family for me. Their support for my studies and selfless love helped me to study smoothly until now. They have always been my strong backing, no matter the success or failure. Their selfless contribution is the prerequisite and foundation for the completion of my studies.

Finally, I would like to thank all the experts and professors who reviewed and commented on this thesis during their busy schedules.

Publications

Journal paper

1) **Haoqi, Gao**, and Koichi Ogawara. ”Bidirectional Mapping Augmentation Algorithm for Synthetic Images Based on Generative Adversarial Network.” IIEEJ Transactions on Image Electronics and Visual Computing, Vol.8, NO.2, pp.110-120, 2020.

International conference

1) **Haoqi, Gao**, and Koichi Ogawara. ”Face alignment using a GAN-based photorealistic synthetic dataset.” International Conference on Control and Robotics Engineering, IEEE. (Accepted)

2) **Haoqi, Gao**, and Koichi Ogawara. ”Face alignment by learning from small real datasets and large synthetic datasets.” Asia Conference on Cloud Computing, Computer Vision and Image Processing, IEEE. (Accepted)

3) **Haoqi, Gao**, and Koichi Ogawara. ”Generative adversarial network for bidirectional mappings between synthetic and real facial image.” Twelfth International Conference on Digital Image Processing, Vol. 11519, pp. 115190J, 2020.

4) **Haoqi, Gao**, and Koichi Ogawara. ”CGAN-based Synthetic Medical Image Augmentation between Retinal Fundus Images and Vessel Segmented Images.” 2020 5th International Conference on Control and Robotics Engineering, IEEE, pp. 218-223, 2020.

5) **Haoqi, Gao**, and Koichi Ogawara. ”Adaptive Data Generation and Bidirectional Mapping for Polyp Images.” 2020 IEEE Applied Imagery Pattern Recognition Workshop, IEEE, pp. 1-6, 2020.

6) **Haoqi, Gao**, and Wang, Huafeng and Feng, Zhou and Fu, Mingxia and Ma, Chennan and Pan, Haixia and Xu, Binshen and Li, Ning. ”A novel texture extraction method for the sedimentary structures’ classification of petroleum imaging logging.” Chinese Conference on Pattern Recognition. Springer, Singapore, pp.161-172, 2016.

Others

1) **Haoqi Gao**, Liangzhi Li, Bowen Wang, Yuta Nakashima, Ryo Kawasaki, Koichi Ogawara, Hajime Nagahara, ”Multi-label Image Classification with Visual Explanations,” Responsible Computer Vision CVPR 2021 Workshop, 2021.6.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In recent years, face-related research has had a wide variety of real-life applications. Uniqueness and non-reproducibility of the human face provide necessary prerequisites for identity recognition. In contrast to other biometric technologies (fingerprint, iris), fingerprint recognition requires the user to press his finger on the sensor, iris recognition requires the user to get close to the camera, and voice recognition requires the user to speak aloud. In contrast, conducting face-related research without disturbing the detected person in a non-contact setting has the advantage of the simplicity of operation and intuitive results. With this in mind, face recognition is the most user-friendly biometric method, which implies that potential applications of face recognition are much broader. For example, face-related research involves information security, transportation, education, health care, and other fields. In these fields, face-related research focuses on cellphones, intelligent door locks, access control, and attendance, medical beauty. It also can be widely used in high-precision scenarios such as airport and train station security checks, intelligent hospitals, bank payments, and other financial areas. With the rapid development of artificial intelligence, computer vision, big data, cloud computing, and other technologies, face-related research has made tremendous progress and has successfully applied in many scenarios.

Fig. 1.1 shows some real-life face applications such as face tracking, fatigue driving detection, face attendance, face payment, face gates, face recognition, face check-in, security monitoring, face beauty effects.

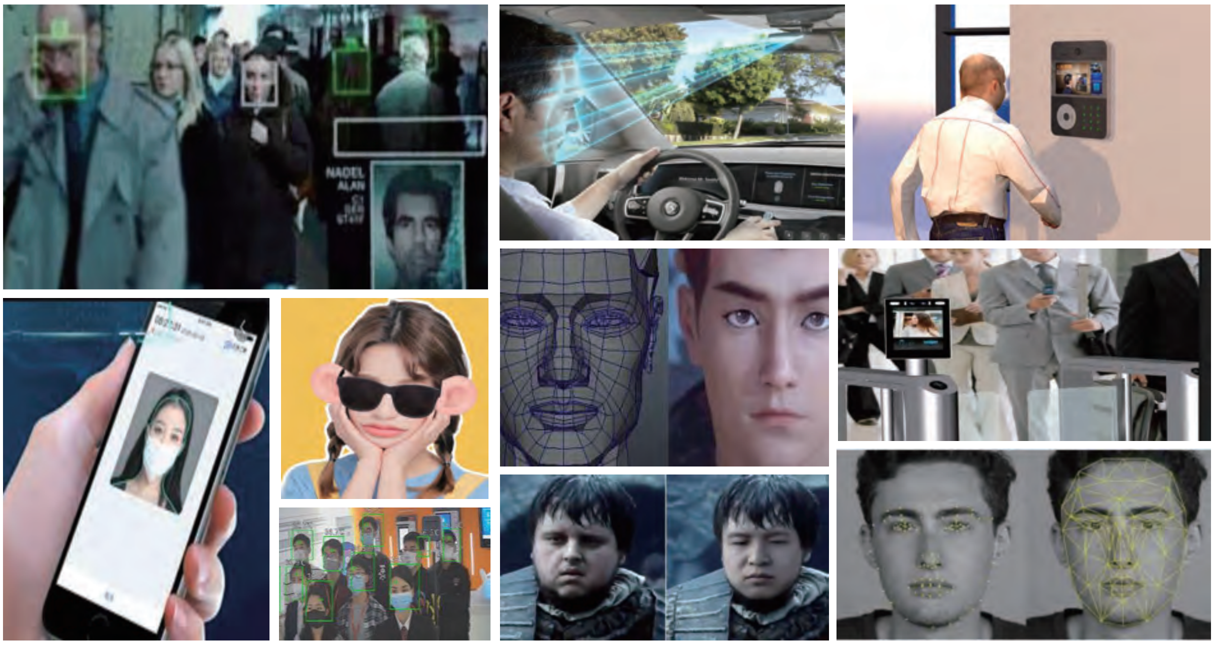


Figure 1.1: Examples of face-related technologies and applications in real life. (Note: images are from the urls ([a-k]) in A.3.

1.1 Background overview

Fig. 1.2 demonstrates some areas of face-related research (e.g., face detection, face alignment, and 3D face reconstruction). The red parts in the figure (face alignment & 3D reconstruction) are this research direction.

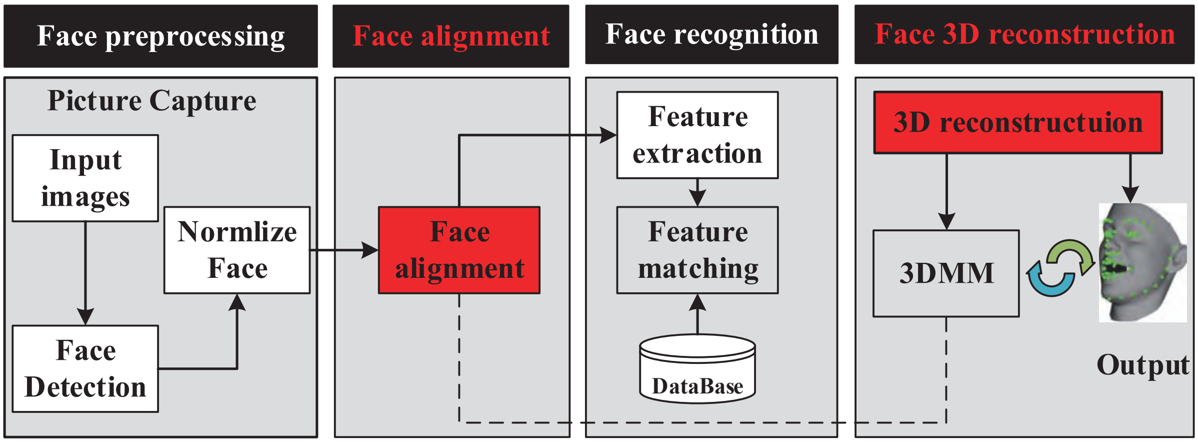


Figure 1.2: Some areas of face-related research.

Facial analysis has always been a hot spot in computer vision and neural networks due to its important theoretical significance and practical application value. With early works dating back 20-

30 years, there are various traditional face-related methods. Common face detection algorithms are AdaBoost classifiers [15], Scale-Invariant feature transform (SIFT), Haar feature, and Histograms of Oriented Gradients (HOG) [16, 17, 18]. A thorough survey [19] describes traditional face detection methods. Traditional facial landmark localization works have focused on generative methods: such as Active Shape Model (ASM) [20] and Active Appearance Models (AAM) [21] and discriminative methods [22, 23]. Some kinds of literature [24, 25] give some surveys about it. Face recognition methods are divided into two main categories: local-based methods, such as recognition using local descriptors Gabor [26], Local Binary Pattern (LBP) [27], etc, and global-based methods, such as Linear Discriminant Analysis (LDA) [28], and the Locality Preserving Projection algorithm (LPP) [29] and other popular learning algorithms [30, 31]. Masi et al. [32] and Guo et al. [33] introduced some face recognition surveys. The traditional 3D face reconstruction approaches employ 3D base shapes to capture a morphological model of face shape variations. Cootes et al. [20, 21] have shown shape variations can be modeled well with subspace analysis. Blanz et al. [34] proposed a 3DMM to represent the shape and texture of human faces by a linear combination of orthogonal basis vectors obtained by Principal Components Analysis (PCA) [35]. Traditional 3DMM reconstruction is an iterative fitting process, which is relatively inefficient.

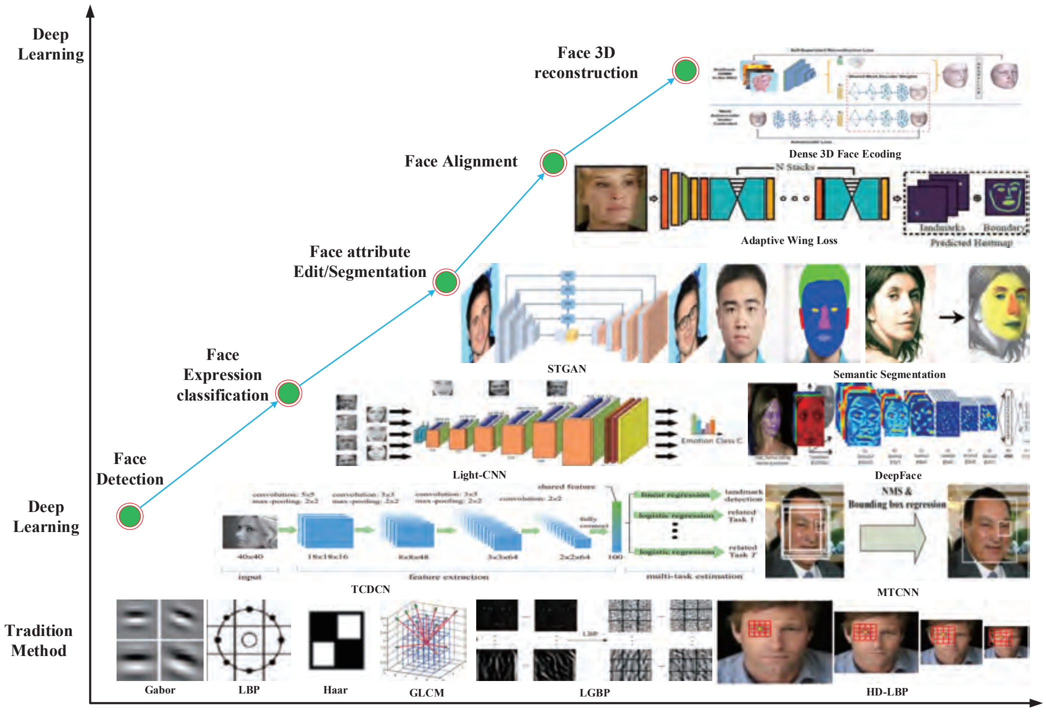


Figure 1.3: History of face-related technologies development [1-10].

Convolutional Neural Network (CNN) is the most common and effective algorithm [36, 37]. Recently, it has gained remarkable progress in a variety of computer vision tasks [38]. With the support of a large amount of face data, face-related research based on deep learning has far surpassed traditional methods in terms of speed and accuracy. For face analysis, the first step is face detection, which aims to locate face regions for given face images or frames of videos [8]. In the previous works, fine-tuned high-level CNN features with Support Vector Machine (SVM) [39, 40] have largely improved the object detection [41]. Fully Convolutional Neural (FCN) network [42] based methods [43, 44] bring a revolution to the field of face detection. With the development of deep CNNs, Cascaded CNN's [45, 46] and region-based CNNs (Faster RCNN) [47, 48], You Only Look Once (YOLO) [49], RestinaFace [50] are getting popular. Next, face landmark detection is an essential intermediate procedure for face analysis. Cascade Network [51] is one of the classic networks in face alignment tasks. Multi-task learning, which by adding facial attribute inference to aid the face alignment task [52, 53]. Other common facial landmark localization methods can be divided into three categories: coordinate regression-based approaches [54, 55, 56], heatmap regression-based approaches [57, 58, 59, 60] and 3D model fitting based [9, 61, 62, 63] approaches. Many deep learning approaches for face recognition included Single-CNN (DeepFace [64], Web-Scale [65]), Multi-CNN (like the series of DeepID [66, 67]), and other variants of CNN [68, 69] show outstanding results in image and speech applications. In contrast to nonlinear optimization, for 3D reconstruction, CNNs can be used as regressors to estimate 3DMM coefficients directly, which can significantly improve the reconstruction quality and efficiency. In the works of [62, 70, 71, 72], researchers proposed an end-to-end 3D face reconstruction method to estimate 3DMM coefficients. Others have attempted unsupervised and weakly supervised training [73, 74, 75].

1.2 Main challenge and purpose

As briefly reviewed above, face-related technology has matured over the past several decades. Despite past efforts, there is still room for improvement due to several problems in the current methods. The following introduces some challenges in face-related research. First, insufficient face datasets, some face datasets are generally not publicly available due to privacy reasons or commercial value. The lack of group-specific training sets makes the face data distribution unbalanced. Second, uncontrollable factors make the quality of captured face images differs significantly from that of training images. In real-life application scenarios, these factors include face deflection, motion blur, out-of-focus blur, occlusions (e.g., mask, sunglasses), low light intensity, low contrast, and loss of face information due to the encoding and decoding process of video transmission.

1.2.1 Insufficient Data

As we all know, deep learning emphasizes the ability to learn features automatically from the given training datasets. If there are not enough training samples, it is hard to provide enough information for the model to train. A small number of data sets make the model difficult to train stably and have weak generalization ability.

1) **Privacy reasons**: Nowadays, facial information is sensitive to personal information. It requires the explicit consent of the user when collecting and acquiring data. Also, it needs to prevent data leakage when storing them. Another reason is the commercial value issues that datasets in the papers or applications are generally not publicly available.

Table 1.1: A list of exists public face databases.

Name	Released Year	Image	Train	Test	Keypoint
AR Face [76]	1998	>4000	-	-	22 points
XM2VTS [77]	1999	2,360	-	-	68 points
BIOID [78]	2001	1,521	-	-	20 points
IMM [79]	2004	240	-	-	58 points
LFW [80]	2007	13,233	-	-	10 points
PUT [81]	2008	9,971	-	-	30 points
AFLW [2]	2011	24,386	20,000	4,386	21 points
LFPW [5]	2011	1,400	1,100	300	20 points
AFW [3]	2012	205	-	-	6 points
HELEN [4]	2012	2,330	2,000	330	194 points
300W [6]	2013	3,837	3,148	689	68 points
COFW [7]	2013	1,852	1,345	507	29 points
MENPO [1]	2017	28,273	12,014	16,259	68 points
WFPW [82]	2018	10,000	7,500	2,500	98 points
JD-landmark [83]	2019	15,393	13,393	2,000	106 points

Table 1.1 lists the basic information of existing face databases including LFW [80], AFLW [2], HELEN [4], WFPW [82], and JD-landmark [83] etc.

2) **Annotation label**: Usually, acquiring the training datasets is not only a matter of data

collection but also producing keypoints labels. Generally, we desire more landmark points in the annotation to capture richer facial structures [8]. Fewer keypoint annotations (e.g., 6 or 20 points) are insufficient for detailed face analysis tasks. This paper uses 68 points annotations for the facial landmark localization algorithm.

As the Table 1.1 shows, the 68 keypoints annotation datasets contain XM2VTS [77], 300-W [6] and MENPO [1]. But large dataset MENPO [1] needs to provide the password, which is not publicly datasets. Taking out the privacy factor, the dataset collection process by ourselves can be very time-consuming and expensive, especially since it is inherently difficult to automate. The manually labeled ground truth facial points will yield errors for the fuzzy or invisible facial landmarks location.

3) **Special group**: The lack of training sets for a particular group makes the trained model not work well for it. For example, some facial datasets across different age groups and different national races are distributed unevenly. As illustrated in Fig. 1.4, age ranging over the person's lifetime leads to wide variation in the face, and the 2nd-row pictures show facial differences (skin color or face contour) under different races.

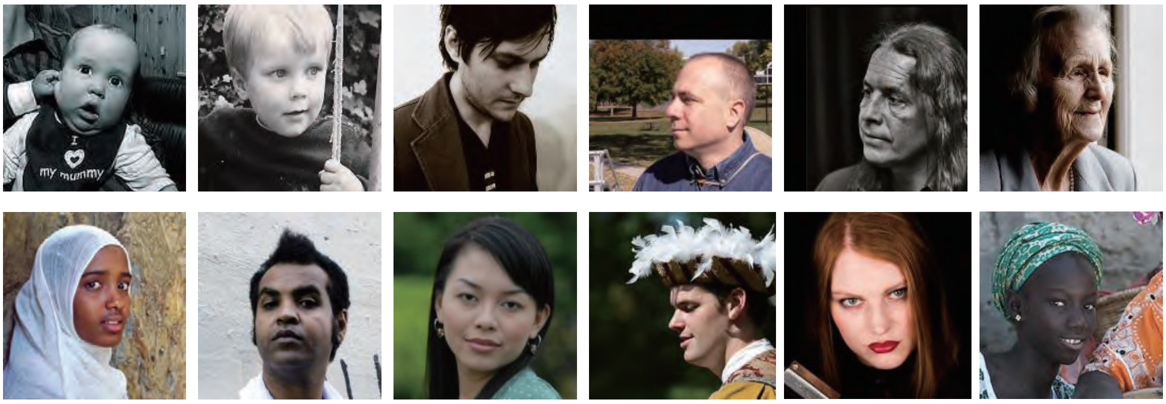


Figure 1.4: Examples for specific groups: faces under the age and race different ranges.(Note: images are from MENPO [1] and AFLW [2])

1.2.2 Unconstrained Situations

As shown in Fig. 1.5, most of the publicly available datasets have angles less than 60°. The publicly available datasets are a limited ability to represent large angular ranges and therefore do not cope well with large poses of faces.

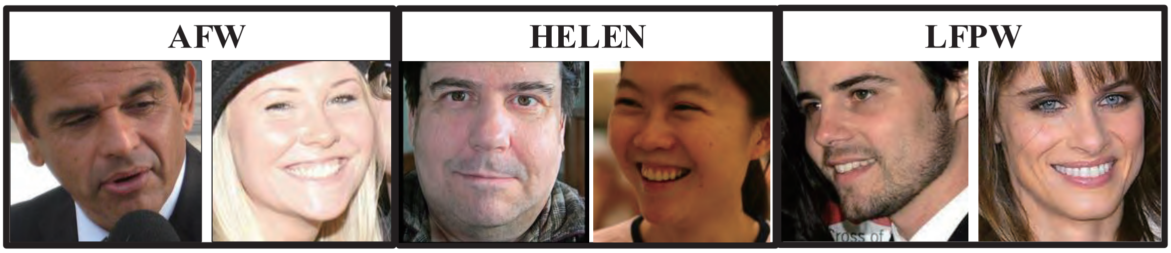


Figure 1.5: Samples of the public face datasets (AFW [3], HELEN [4], and LFPW [5]).

But face images are highly variable in the real world, especially when the faces captured by the cameras are in unconstrained situations. The variability of the face images includes head pose, low resolution, lighting conditions, occlusion, and facial expression. Due to the influence of those factors, traditional face recognition methods are limited in their recognition accuracy.

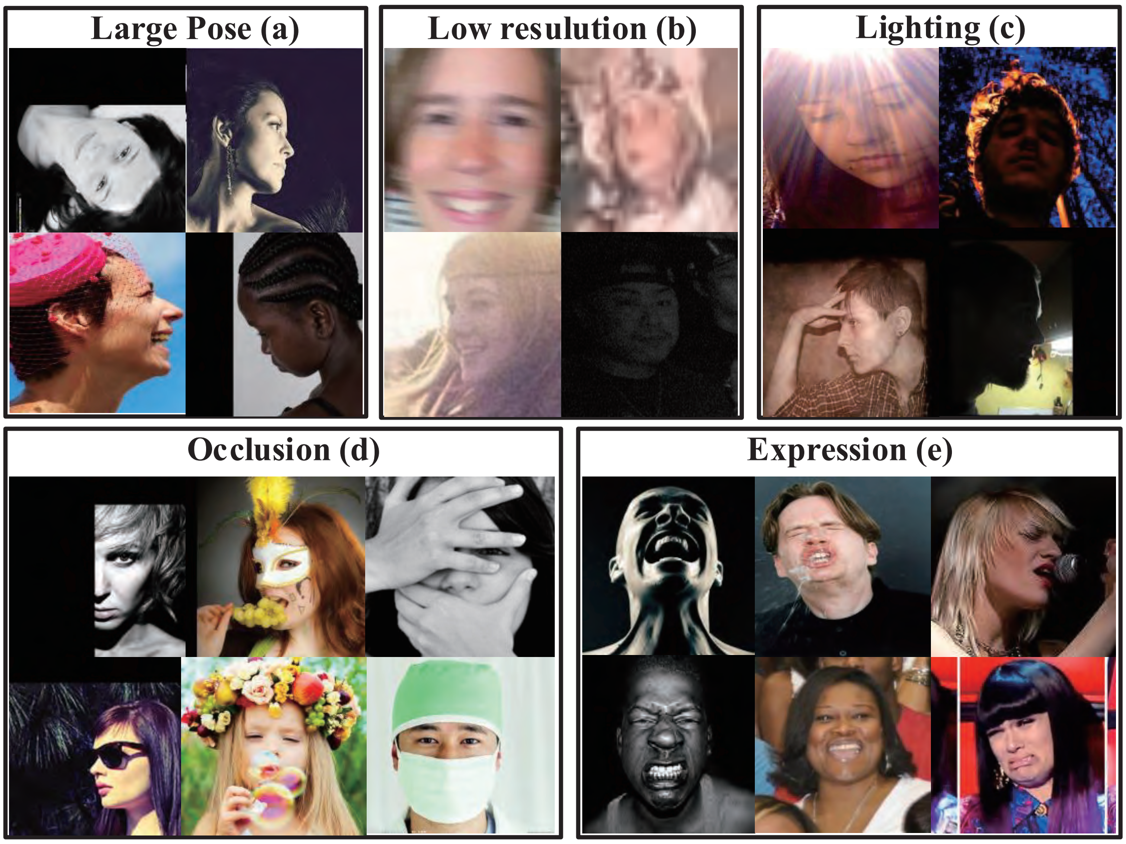


Figure 1.6: The challenging face dataset in the wild.(Note: images are from 300W [6], MENPO [1], COFW [7], and AFLW [2])

1) **Large pose**: The appearance of face differs greatly between different camera-object poses (Fig. 1.6 (a)) (e.g., frontal, profile, head up or down) [24]. For profile faces, one side of the

face contour is invisible. It caused face keypoint annotations to need to guess by the person.

2) **Low resolution**: Low-resolution facial images (Fig. 1.6 (b)) lack sufficient visual information. Even expert human operators can find it hard to locate landmark positions. The main reasons for low resolution can be different lighting conditions (e.g., fog, rain, snow, dust), motion blur or out-of-focus blur, or possible camera contamination.

3) **Lighting conditions**: Different lighting (Fig. 1.6 (c)) such as intensity and direction of illumination, overexposure, and underexposure of the camera sensor may significantly change the appearance of the face and lose the face detailed textures.

4) **Occlusion**: Facial occlusion (Fig. 1.6 (d)) can lead to the disappearance of some facial features, resulting in a decrease in the accuracy of the training model. There are three main categories of several types of occlusions: first, physical occlusions: objects covering the face (e.g., hats, glasses, scarves). Second, light occlusions: uneven or intense external lighting leads to occlusion. Third, self-obscuration, caused by the human posture, such as the profile face.

5) **Expression**: The corresponding changes (Fig. 1.6 (e)) in the texture and appearance of the face brought about by muscle movements when the facial makes an expression will have a global information effect on the face image. For example, laughing may cause large deformation of the shape of the mouth or eyes.

1.3 Synthetic datasets

Different applications of face-related research will have various requirements on the data, and the requirements for training images depend on the scene complexity, the number of item classes, and accuracy. Synthetic datasets will effectively reduce the development cost of computer vision applications while ensuring compliance with strict privacy and regulatory standards. For example, privacy regulations limit the availability of real-world data or dictate how to use datasets. Synthetic data is necessary when real-world data may not exist or does not meet specific conditions or needs. In addition, it is expensive to generate data in real life for algorithm training, which requires large amounts of data. Synthetic datasets make up for many of the shortcomings of manually collecting and labeling real-world data.

Synthetic data refers to computer-generated by generator models, which is anonymous and is created based on user-specified parameters. The advantages of synthetic data are as follows:

1) Synthetic data reduces the need to capture data from real-life events and generates and builds datasets much faster. For events that rarely occur, synthetic models can simulate more data from some real-world data samples.

2) Synthetic data can be automatically labeled at the time of generation, which provides controlled and reliable annotation, thus reducing the time and cost required to label the data.

3) Synthetic data sets can minimize privacy concerns because synthetic data is not based on real people or real-life events.

This thesis work opens research directions through synthetic datasets. It demonstrates the great potential of synthetic data in reducing the various biases in real-world datasets. The experiments suggest the complementarity and availability of the synthesized face images.

1.4 Outline

This thesis consists of the following seven chapters. Chapter 1 describes the background and purpose of the research, as well as the challenges. Chapter 2 overviews some previous works, which are most closely related to this paper. To address the challenge of insufficient datasets, Chapter 3 uses the 3D FaceGen model (http://www.facegen.com) to synthesize an unlimited number of training datasets with automatically created annotated labels, which reduces the burden of both the data collection and labeling process. The synthetic samples allow us to precisely control the rendering process of the images to make the datasets have different characteristics (e.g., race persons, various angles, and various facial expressions). By using real front datasets plus synthetic datasets to train the face landmark detection model for challenging faces. The results demonstrate the great potential of synthetic data in reducing the various biases impact in real-world datasets. Considering the ”domain gap” between synthetic datasets and real-world datasets, Chapter 4 devises a novel generative transfer model ($R \rightarrow S$) based on Cycle Generative Adversarial Network (CycleGAN) [84] and test the accuracy of the proposed method with a face alignment task. Chapter 5 proposes a face alignment model based on another transfer model ($S \rightarrow R$) that converts the synthetic face images generated by the FaceGen model into more realistic face images. The proposed algorithm not only protects the privacy of faces but also improves face detection accuracy. For the face 3D reconstruction task, computational complexity is another consideration besides the accuracy of the model. Other synthetic data set synthesized from 300-W dataset [6] through a morphable model-based 3D profiling algorithm proposed by [9]. It coverages across large pose ranges from $-90°$to $90°$. Chapter 6 designs an improved method that combines three major achievements: 3DMM [34, 85], ShuffleNet series of units [86, 87], and Squeeze-and-excitation (SE) attention mechanism [88] to do face alignment and 3D face reconstruction at the same time. The results show proposed network can improve the accuracy of face alignment without adding too many network parameters and GFLOPs, which shows the effectiveness of the proposed model. Finally, the results of this study are summarized, and future work is introduced.

Chapter 2

Related work

This chapter focuses on face studies that are closely related to the thesis. The work of face-related research mainly contains facial alignment and 3D face reconstruction algorithms. Topics for synthetic datasets include 3D models as well as generative networks.

2.1 Face alignment

Facial landmark detection is also known as face alignment [89, 90, 91], which aims to estimate the projection of facial key points (such as eye, nose tip, eyebrow, chin center, and mouth corners).

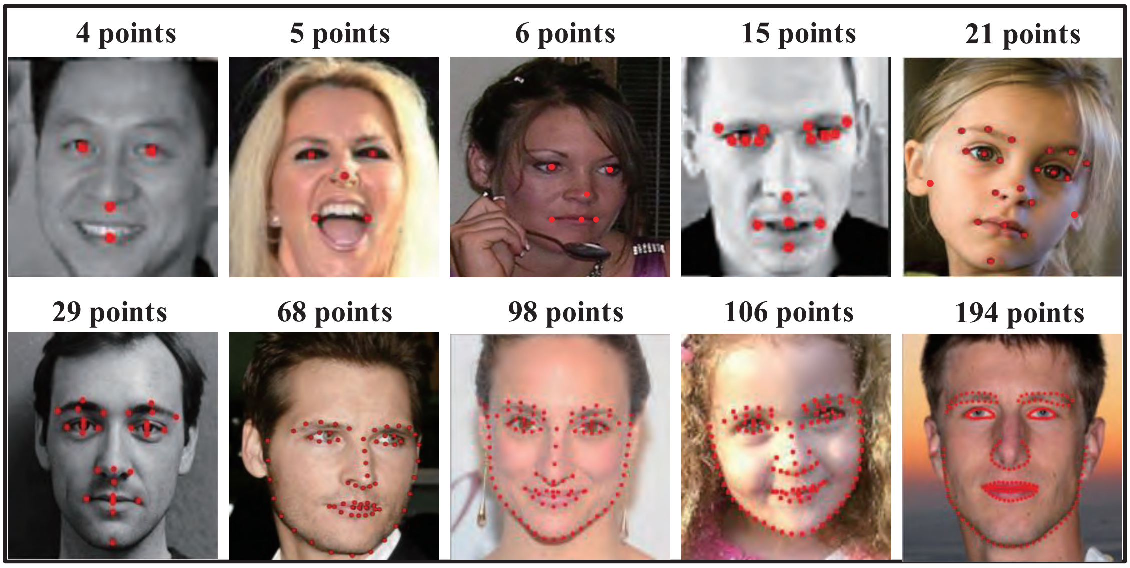


Figure 2.1: Facial landmarks for different tasks [8]. The 68-point and 5-point landmarks are commonly used for existing face alignment algorithms.

The majority of the research on [59, 92] implies that face alignment and facial landmark detection are used as interchangeable terms [11]. On the face analyses, face alignment can be considered an essential intermediate step. Many important tasks, such as face recognition, face tracking, facial expression recognition, and head pose estimation, can benefit from the accurate localization of facial points [24]. However, facial landmark detection remains challenging due to the ambiguity of landmarks under invisible viewpoints, and it is hard to train a unified model on it. For most existing methods of face alignment, the facial landmarks, or so-called facial keypoints (see Fig. 2.1), are indispensable. Fig. 2.1 shows samples on some face tasks where the number and type of facial points required may differ. Over the last decade, face alignment has substantial advances in progress. Many of the methods achieve high accuracy in detecting the landmark in both frontal and near-frontal face images. To better understand face alignment, this chapter reviews of some related methods based on deep learning models.

According to existing face alignment methods, it can categorize landmark-based alignment methods into three subcategories: 3D model fitting-based methods, coordinate regression-based methods, and heatmap regression-based methods [8].

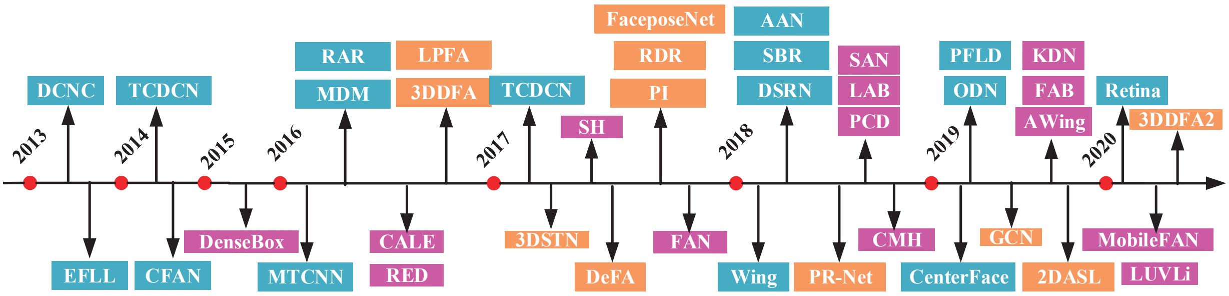


Figure 2.2: Development of face alignment methods. The blue, plum, orange parts represent the coordinate regression, heatmap regression, and 3D model-fitting methods, respectively.

2.1.1 Coordinate regression

Coordinate regression-based methods consider landmark coordinates as regression targets through neural networks that learn the map from face images to landmark coordinate vectors. Sun et al. [51] first proposed a Deep Convolutional Network Cascaded (DCNC) for facial points estimation. Zhou et al. [52] designed an Extensive Facial Landmark Localization (EFLL) network based on DCNC to localize extensive facial landmarks with a coarse-to-fine convolutional network cascade.

Zhang et al. [93] proposed a Coarse-to-Fine Auto-encoder Networks (CFAN), which cascades a few successive Stacked Auto-encoder Networks (SANs). Once a deep model performs multiple tasks learning, features learned from one may be used for others. In [53], they designed a Tasks-Constrained Deep Convolutional Network (TCDCN) to jointly optimize facial landmark detection with a set of related tasks such as expression, head pose. Zhang et al. [38] proposed a deep cascaded multiple tasks framework consisting of three stages (MTCNN) to exploit the inherent correlations between them to boost up their performance. The Mnemonic Descent Method (MDM) [94], and Recurrent Attentive-Refinement Network (RAR) [95] employed CNN and RNN together to extract global features and refine the prediction. Wu et al. [96] introduced a Tweaked CNN (TCNN) that improves CNN-based landmark detection by differently tweaking the processing of different intermediate features. For facial landmark localization, several methods developed new regression approaches, such as Miao et al. [97] proposed the first shape regression model (DRSN) for end-to-end face alignment without relying on cascaded models. Yue et al. [98] designed an attentional alignment Network (AAN) to do the face alignment. Dong et al. [99] presented a supervised approach (SBR) to encourage the optical flow coherency of detected landmarks, which can improve the accuracy of facial landmark detectors on both images and videos. However, the regular loss is L2 loss in existing deep neural network-based facial landmark detection systems. Feng et al. [55] proposed a new loss function, namely Wing loss, which amplifies the influence of samples with small or medium range errors, which further improves the accuracy of CNN-based facial landmark localization systems. For facial images collected in complex situations like unconstrained poses, expressions, and illumination, researchers also proposed numerous approaches, such as the work of [100] presented occlusion-adaptive deep networks (ODN) to overcome the occlusion problem for robust facial landmark detection. PFLD [101] employed a branch of the network to estimate the geometric information of faces and subsequently regularize the landmark localization. Xu et al. [102] proposed a Center face model based on the multitask learning strategy to predict the face boxes and landmark points at the same time. Retinaface model [50] forces the network to learn exclusive facial features by jointly learning facial bounding box locations, facial landmarks, and 3D vertices.

2.1.2 Heatmap regression

Compared to coordinate regression, the output of heatmap regression methods is likelihood response maps for each landmark [8]. The idea of jointly regressing part-detection score maps for localization explores in [103] in the context of human pose estimation. Early research such as DenseBox [104], and Convolutional Aggregation of Local Evidence (CALE) [105] studied how to aggregate the score maps along with early CNN features through joint regression to refine the face landmark prediction. Peng et al. [106] proposed a Recurrent Encoder-Decoder (RED) network for landmark localization in sequence images. Newell et al. [107] were pioneers in proposing a Stacked Hourglass (SH) network to generate heatmaps for human pose estimation. The crucial idea behind Hourglass models is repeating resolution-preserving bottom-up and top-down processing in conjunction with intermediate supervision. The current state-of-the-art performance in face alignment maintains for some time by using Hourglass models or other deformable Hourglass models [57, 58, 60, 108]. For example, Bulat et al. [58] proposed the Face Alignment Network (FAN), constructed by stacking four HourGlass models. Deng et al. [60] proposed a multi-view Hourglass model, which allows the network to capture the features from different scales, as well as the context information.

Besides the Hourglass structure, several effective heatmap regression architectures have been proposed, such as Kumar et al. [109] proposed a Pose Conditioned Dendritic (PCD) network which can capture shape constraints in a deep learning model. Considering the issue of the large variety of different image styles, Dong et al. [110] proposed a Style-Aggregated Network (SAN) for facial landmark detection. Furthermore, landmarks in occluded facial regions also cause incorrect annotations. Wu et al. [82] proposed a novel boundary-aware face alignment algorithm by utilizing boundary lines as the geometric structure of the human face to help facial landmark localization. Regarding motion-blurred images, Sun et al. [111] proposed a framework named FAB that takes advantage of structural consistency in the temporal dimension for facial landmark detection in motion-blurred videos. To address the imbalance between foreground and background pixels. Wang et al. [91] designed a novel loss function, named Adaptive Wing loss, that can adapt its shape to different types of ground truth heatmap pixels. Kernel Density Network (KDN) [112], and LUVLi [113] LUVLi [113] presented a novel framework for jointly predicting landmark locations, associated uncertainties of these predicted locations, and landmark visibilities. Recent research also focuses on whether it can reduce network parameters and high memory consumption. Zhao et al. [114]

proposed a lightweight model, namely Mobile Face Alignment Network (MobileFAN), using a simple backbone MobileNetV2 as the encoder and three deconvolutional layers as the decoder.

2.1.3 3D model fitting

There is a clear relationship between 2D facial landmarks and 3D facial shapes. Briefly speaking, a 3D model fitting-based approach reconstructs 3D face shapes from 2D images and then projects them onto the image plane to obtain 2D landmarks. The benefit of 3D model fitting-based methods can fit faces to 3D model vertices and align them with large poses. Early research of Large-pose Face Alignment (LPFA) [61] and 3DDFA [9] explored face alignment for large-pose face images, by combining cascaded CNN's regressor and 3DMM model [61]. Liu et al. [89] proposed a Dense Face Alignment (DeFA) model for 3D facial shape estimation, which aligns limited facial landmarks and fits facial contour points. Beyond regressing 3D facial shape parameters, in [11], FacePoseNet (FPN) estimates warping parameters by rendering a different view of a general 3D face model. Bhagavatula et al. [115] presented an approach to simultaneously extract the 3D shape of a face and the semantically consistent 2D alignment through a 3D Spatial Transformer Network (3DSTN) to model both the projection matrix of the camera and the warping parameters of the 3D model. Jourabloo et al. [116] proposed a Pose-invariant (PI) architecture for model fitting, which consists of several visualization blocks to adjust the 3D shape and projection matrix. Xiao et al. [117] proposed a novel Recurrent Dual Refinement (RDR) model that provides a closed-loop learning process for 2D landmark detection and its dual task of 3D face model refinement. Some papers have also proposed methods for simultaneously reconstructing the 3D facial structure and providing dense alignment. Position map Regression Network (PR-Net) [62] proposed a 2D representation called UV Positional Map, which records the 3D shape of faces in UV space, then trains a simple Convolutional Neural Network to regress it from images. Volumetric Regression Network (VRN) [118] performed a direct regression of the volumetric representation of 3D facial geometry from the 2D image. With the idea that 3D facial mesh can be represented by a graph, Wei et al. [119] proposed a graph convolution network to regress 3D face coordinates, which directly performs feature learning on the 3D facial mesh. Recent research for the challenge of wild face images include 2D-Assisted Self-supervised Learning (2DASL) [63], and 3D Dense Face Alignment (3DDFA) [120], which show state-of-the-arts for both 3D face reconstruction and dense face alignment by a large margin.

2.2 CNN-based Face reconstruction

Many works have been proposed for the 3D facial reconstruction task, which improves 3DMM based modeling performance. Most previous approaches regress the parameters of 3DMM by solving a non-linear optimization problem to establish a point correspondence between a 2D facial image and a canonical 3D model of the face. However, such methods are usually time-consuming and heavily rely on the accuracy of landmarks or other feature-points of the detector [121]. Since advances in deep learning, CNN-based approaches have achieved remarkable success in many areas of computer vision. As opposed to non-linear optimization, CNN can be employed as regressors to estimate the coefficients of the 3DMM directly, which can significantly improve the quality and effectiveness of reconstructions. This section presents and compares the most relevant works in 3D face reconstruction that use deep learning as the essential tool [14].

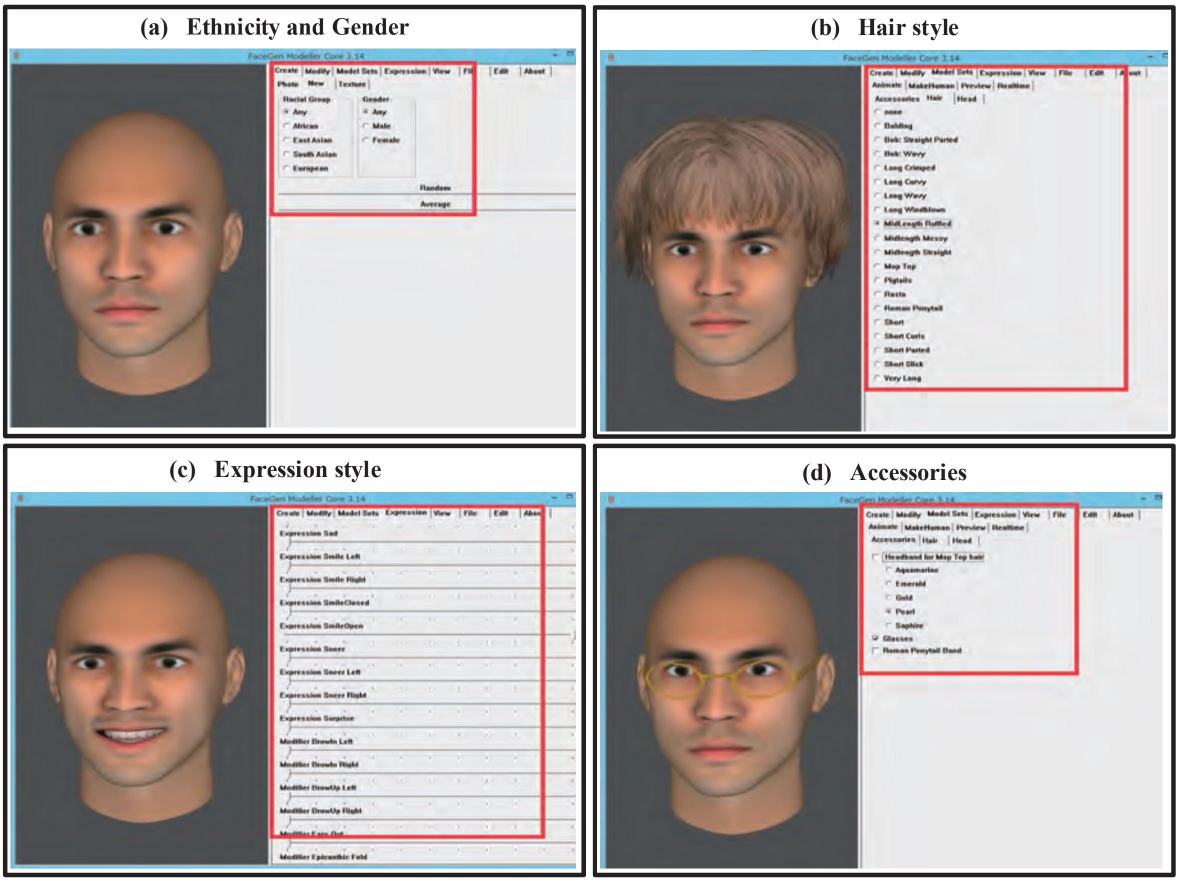
In general, current 3D face reconstruction methods are briefly summarized as follows: the parametric representation provided by 3DMMs [71, 74, 122, 123, 124, 125, 126], depth maps [127, 128, 129], UV space [62, 130], volumetric representations [118, 131], and image space [9, 10, 61, 116]. In the first category, parametric prediction of 3DMMs is the most commonly adopted method. Because 3DMMs model the shape variations with a Euclidean subspace and 3D face described by the model parameters [14]. For instance, Richardson et al. [122] proposed a ResNet-based architecture [132], which extracts the face geometry directly from its image. Dou et al. [71] designed an approach for End-to-End 3D Face Reconstruction (UH-E2FAR) from a single 2D image. Wu et al. [126] extracted features from three images of the same subject (frontal, left, and right views), and the model regressed the projection parameters, the joint 3DMM shape, and expression parameters separately. Genova et al. [74] utilized the FaceNet network and passed its output into a decoder that estimated the 3DMM parameters. For the second category, Sela et al. [127] used the network to estimate both a depth map and a correspondence map from each image pixel to a vertex of the reference mesh. Koizumi et al. [128] proposed to estimate a dense image-model correspondence map with an image-to-image CNN architecture, which is a completely unsupervised strategy for learning to fit a 3DMM to a single image. Shou et al. [129] proposed the main network to estimate a depth map and two supplementary networks to help regularize the output of the main network. However, some methods are highly dependent on the image and the template. In the UV Space, Feng et al. [62] designed the 3D face geometry into the UV location map and trained a CNN to directly regress the complete 3D facial structure together with the semantic information from a single image. In the work of [130], they proposed to add branches at the end of the ResNet-based network to estimate the U-and V-coordinates of a UV-map separately. Another technique is to construct a volumetric face representation, defined as a 3D binary discrete volume. For example, Jackson et al. [118] proposed to map image pixels to a volumetric representation of 3D facial geometry through CNN-based regression. Yi et al. [131] proposed to utilize a volumetric subnetwork to estimate an intermediate geometry representation and a parametric subnetwork to regress the 3DMM parameters. While the volumetric representation is no longer limited to a 3DMM space, it needs a complex network structure and much time to predict voxel information. Jourabloo et al. [61, 116] proposed to estimate a camera projection matrix and 3D shape parameters using a cascade of CNN-based regressors. Zhu et al. [9] trained a 6-layer CNN iteratively to output updates of the pose, shape, and expression parameters. Tewari et al. [123, 124, 125] proposed a convolutional encoder network that can be trained end-to-end in an unsupervised manner. This unsupervised method makes it feasible to train on absolutely large (unlabeled) real-world data. However, unsupervised methods perform poorly on large poses or heavily occluded faces.

2.3 Face Model

2.3.1 FaceGen

FaceGen is a 3D face generating middleware 3D modeling produced by Singular Inversions [133], which employs a ”parameterized” approach to define the properties that make up a face. FaceGen uses multiple controls for editing faces: like age, race, and gender. It can create three-dimensional human faces randomly or from photos. Faces are represented by using 100 dimensions (50 shape dimensions and 50 reflectance dimensions; for more details, see Procedure).

By using a fixed set of parameters, FaceGen can morph and modify the face model independently. Fig. 2.3 shows the set of parameters for FaceGen to generate faces (the parts marked with red lines). FaceGen Modeller allows the user to randomize, tween, normalize faces, and also includes algorithms for adjusting apparent age, ethnicity and gender (see Fig. 2.3 (a)), hairstyle (see Fig. 2.3 (b)), and other accessories parts like glasses (see Fig. 2.3 (d)). FaceGen Modeller also allows limited parametric control of facial expressions style (see Fig. 2.3 (c)). All faces (male or female) are generated randomly.


(a) Ethnicity and Gender
FaceGen Modeller Core 3.14
Create | Modify | Model Sets | Expression | View | Fil | Edit | About
Photo | New | Texture
Random
Average
(b) Hair style
FaceGen Modeller Core 3.14
Create | Modify | Model Sets | Expression | View | File | Edit | About
Animate | MakeHuman | Preview | Realtime
Accessories | Hair | Head
(c) Expression style
FaceGen Modeller Core 3.14
Create | Modify | Model Sets | Expression | View | File | Edit | Abou
(d) Accessories
FaceGen Modeller Core 3.14
Create | Modify | Model Sets | Expression | View | File | Edit | About
Animate | MakeHuman | Preview | Realtime
Accessories | Hair | Head

Racial Group
● Any
○ African
○ East Asian
○ South Asian
○ European

Gender
● Any
○ Male
○ Female

○ none
○ Balding
○ Bob: Straight Parted
○ Bob: Wavy
○ Long Crimped
○ Long Curvy
○ Long Wavy
○ Long Windblown
● MidLength Ruffled
○ Midlength Messy
○ Midlength Straight
○ Mop Top
○ Pigtails
○ Rasta
○ Roman Ponytail
○ Short
○ Short Curls
○ Short Parted
○ Short Slick
○ Very Long

Expression Sad
Expression Smile Left
Expression Smile Right
Expression SmileClosed
Expression SmileOpen
Expression Sneer
Expression Sneer Left
Expression Sneer Right
Expression Surprise
Modifier BrowIn Left
Modifier BrowIn Right
Modifier BrowUp Left
Modifier BrowUp Right
Modifier Ears Out
Modifier Epicanthic Fold

□ Headband for Mop Top hair
○ Aquamarine
○ Emerald
○ Gold
● Pearl
○ Saphire
☑ Glasses
□ Roman Ponytail Band

Figure 2.3: FaceGen Modeller (Singular Inversions, Toronto, Canada).

FaceGen can generate near-photorealistic faces having a unique identity. For example, the generated faces change from young to old, from male to female, from one portrait to another, from one race to another. Fig. 2.4 gives some examples of different faces with different settings (including ethnicity, age, gender, or expression). It can obtain various male and female faces by setting FaceGen's gender control from (extremely masculine) to (extremely feminine), get ethnicity faces by setting FaceGen's ethnicity control, and have different age faces by setting FaceGen's age control.

The expressions of sadness, disgust, surprise, and smile with the expression manipulation tools available in FaceGen (see red lines part in Fig. 2.4). FaceGen also supports adding several emotional expressions to any face, and the intensity can be set anywhere between 0 and 100. Here, it can also generate gender (see blue lines part in Fig. 2.4), ethnicity (see green lines part in Fig. 2.4), and age (see magenta lines part in Fig. 2.4). All the generated faces using the face space model implemented in FaceGen.

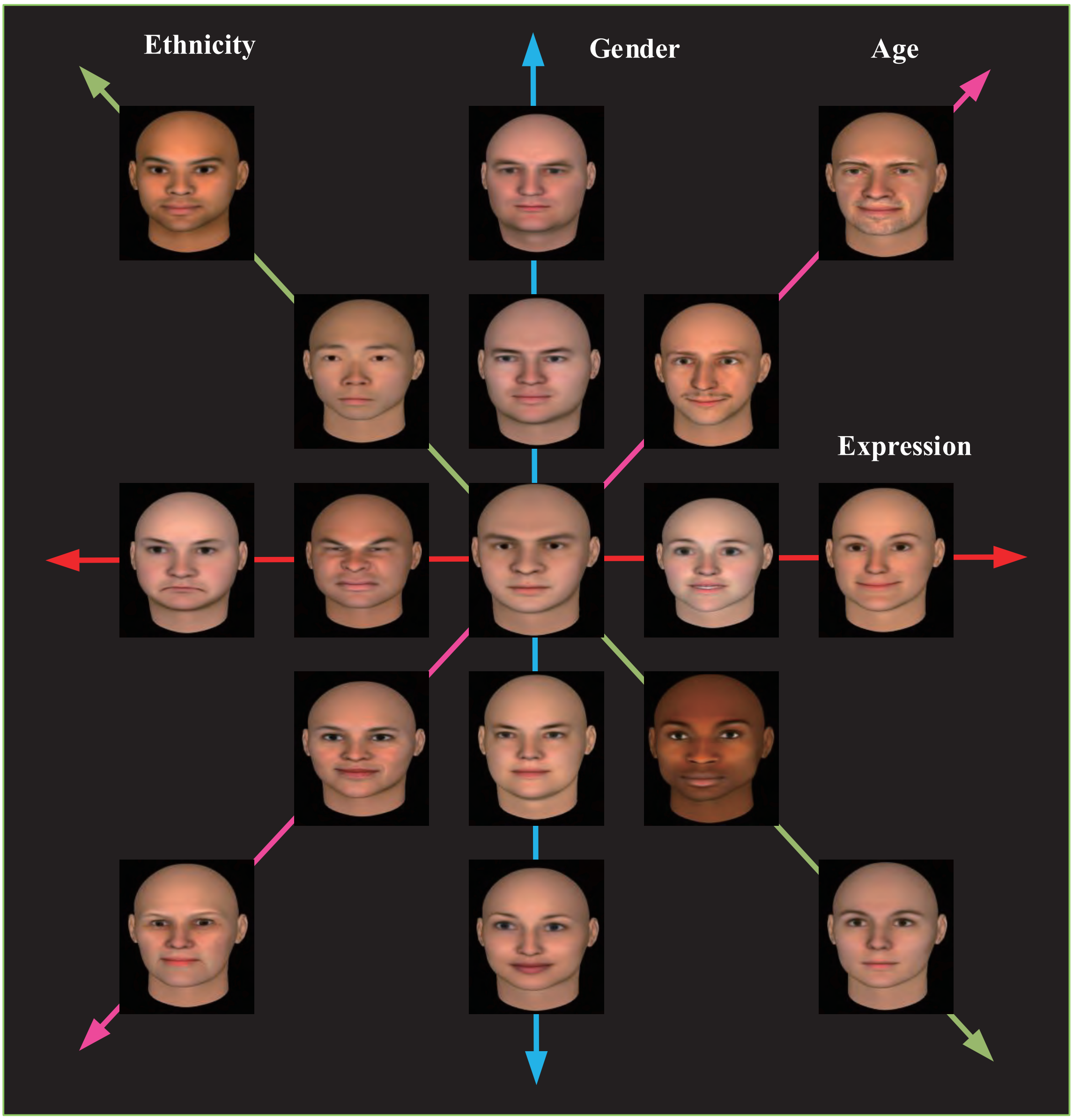


Figure 2.4: Examples of faces created by FaceGen Modeller.

In the FaceGen Modeller, individual faces are represented by a combination of facial shape and face reflectance components [134]. Facial shape corresponds to the combination of positions and shapes of facial features (e.g. eyes and chin) described by the vertex positions of a polygonal model of fixed mesh topology [134, 135]. Face reflectance includes facial properties such as brightness, color, and texture variations on the surface map of the faces [136, 137]. Computer-generated faces allow for greater control of facial features. Due to the simplicity and relatively low cost of FaceGen, many works [134, 137, 138, 139] generate facial images and 3D face models by using it. Nakamura et al. [134] designed a face model of the attractiveness of East-Asian faces by using FaceGen to generate the attractive face database. Jeni et al. [138] proposed the multi-class SVM method for classification by using the FaceGen Modeller to acquire a 3D emotionally modulated database with different poses. Todorov et al. [137] built a social perception model for faces by applying a face space model as implemented in FaceGen. Aksasse et al. [139] introduced a face alignment approach that employs a single-3D face model as a reference produced by FaceGen Modeller.

2.3.2 3D Morphable Models

3D Morphable Models (3DMMs) are powerful 3D statistical models of human facial shape and texture, proposed by Blanz and Vetter [34, 85]. Compared with 2D models, 3DMMs separate rigid (pose) and non-rigid (shape and expression) transformations, enabling it to cover diverse shape variations while preserving shape priors. It proved that 3DMM is capable of inferring a 3D facial surface from a single image of a person [140]. Several 3DMMs have been built and made public over the last decade. Table 2.1 lists the characteristics of the most popular available 3DMMs.

Table 2.1: Characteristics of the most popular available 3DMMs [14].

3DMM	Released Year	Subjects	Age	Ethnicity	Expression
Blanz[34]	1999	200	young	-	-
BFM [141]	2009	200	8-62	most Europens	-
FaceWarehouse [142]	2013	150	7-80	Various	Yes
Multilinear Wavelet[143]	2014	99	-	-	Yes
SFM [144]	2016	169	young	most Caucasian	-
LSFM[145]	2017	9663	about 1-80	most White	-
BFM2017 [146]	2017	200	8-62	-	Yes
LYHM [147, 148]	2017	1212	about 5-90	-	-

Paysan et al. [141] constructed the well-known Basel Face Model (BFM) by applying the Non-rigid Iterative Closest Point (NICP) algorithm [149] to compute these dense correspondences directly between 3D faces. Cao et al. [142] constructed a FaceWarehouse model from depth maps of 150 individuals aged between 7 and 80 years old from various ethnic backgrounds. Each individual involves a neutral expression and 19 other expressions such as mouth-opening, smile, kiss. The Blanz and Vetter face model [34] was fitted to the depth maps to obtain meshes in dense correspondence. A bilinear face model was built by applying Higher-Order Singular Value Decomposition

(HOSVD) to the 3-rank data tensor (vertices, identities, expressions) constructed from the vectorized meshes [14]. Brunton et al. [143] built a Multilinear Wavelet facial model by also applying HOSVD to separate identity from variations expression facial, and using a training set of facial scans from 99 subjects with 25 expressions each. Huber et al. [144] presented Surrey Face Model (SFM), which are constructed from 169 objects very diverse in both age and ethnicity. Booth et al. [145] suggested the largest 3DMM until now, the Large Scale Facial Model (LSFM), which used the detected 2D landmarks to map to a 3D face by inverting the rendering, after then, deforming a predefined template mesh to fit the face shape. BFM was extended by Gerig et al. [146] who established a dense correspondence with a Gaussian process deformation model instead of using the NICP algorithm. Dai et al. [147, 148] built a Liverpool-York Head Model (LYHM) of the whole head, whose dense correspondences were also established by deforming a 3D facial template to each facial shape. However, their method is based on the coherent point drift algorithm [150] and refined the correspondences using optical flow for the texture channel.

3DMM is applied in face analysis widely, including model fitting, image synthesis, and face alignment. There are existed several surveys on 3D face analysis based on 3DMM [14]. 3DMM fitting representative approach based on the minimization of the difference between the input and the rendered image is followed by many others [14, 151, 152, 153, 154, 155]. For instance, Zhu et al. [9, 10, 151] jointly estimated the 3DMM parameters, the projection parameters, and the position of the 3D contour landmarks on the given 2D image. Blanz et al. [152] reconstructed global 3D faces from multiple images of a person and then fused the reconstructions by a criterion based on the accuracy of each of the subregions. Booth et al. [153] constructed a texture model using dense feature-based representation, then by combining a robust statistical model of facial shape to propose an ”in-the-wild” 3DMM. Liu et al. [155] updated the 2D contour landmarks iteratively while estimating the shape and projection parameters.

2.4 Generative Adversarial Network (GAN)

The Generative Adversarial Network (GAN) was proposed in 2014 by Goodfellow [84]. It comprises two different models called Generator and Discriminator to fight each other. The Generator network takes random samples from the latent space as input, and its output needs to mimic the real-world data in the training set as closely as possible. The Discriminator network feeds with either the real one or the Generator network output. It aims to distinguish the Generator network output from the real as far as possible. The Generator network has to deceive the discriminator network as much as possible. The two networks work against each other, constantly adjusting their parameters, with the ultimate goal of making the Discriminator network unable to determine whether the output of the generative network is correct or not until the Nash equilibrium is reached. GANs model pipeline is illustrated in Fig. 2.5.

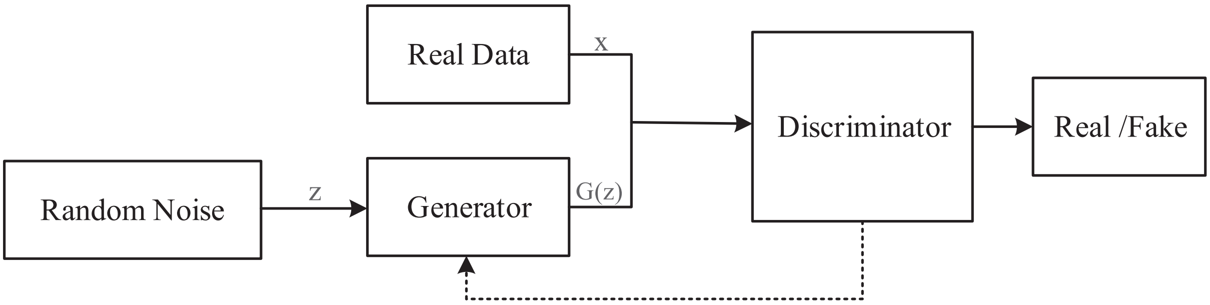


Figure 2.5: The pipeline of GANs model.

As shown in Fig. 2.5, the Generator network is mapping the random noise z to the synthesized sample G(z) that should 'fool' the Discriminator model. Then the synthesized sample G(z) and the real-world data x are added to the Discriminator model. After then the Discriminator model judges whether the input is real or generated data by the generator. Through continuous learning, the Discriminator model can not recognize the data generated by the generator to falsify data.

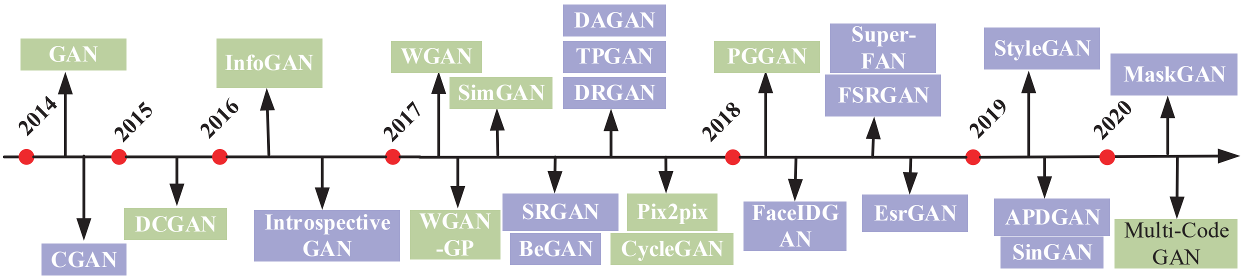


Figure 2.6: The development of GANs methods. The purple, green represent classical GANs and GANs used for face analysis, respectively.

The initial GANs have some issues, for example, the difficulty of training, the difficulty of convergence, and the lack of diversity in the generated samples. Since then, many researchers [156, 157, 158, 159, 160, 161, 162, 163] have tried to solve those problems and proposed improvements. Radford et al. [156] proposed a deep convolutional generative adversarial network (DC-

GAN). Arjovsky et al. [158] proposed WassersteinGANs (WGANs) that use Earth-Mover's rather than Jensen-Shannon to measure the distance between the real sample and generated sample distributions. Gulrajani et al. [159] proposed a gradient penalty (WGAN-GP) which can improve WGAN. Gauthier [160] proposed Conditional GAN (CGAN) by adding extra information, which can be the label or additional auxiliary information. Shrivastava et al. [161] suggested SimGAN, which supplements the adversarial loss with a self-regularization L1 loss that penalizes large changes between the synthetic and refined images to make the synthetic datasets more realistic and can be used to enrich unlabeled real datasets. The pix2pix framework [162] proposed a generator to learn the mapping function between two paired images. CycleGAN [163] introduced a cycle generator confrontation network and puts forward a transfer between different image domains without the need for specific image pairs. The previous method is successful for style transfer tasks mapping local texture but typically unsuccessful for image translation tasks with larger shape changes in the wild. More recent studies focus on flexibly controlling the amount of change of shape and texture. Such as Karras et al. [164] proposed the Progressive GAN (PGGAN), which uses a novel methodology for training GAN based on progressive neural networks. Kim et al. [165] proposed a novel method (UGATIT) for unsupervised image-to-image translation that incorporates a new attention module and a new learnable normalization function. Gu et al. [166] proposed a novel approach, called mGANprior, to incorporate the well-trained GANs as effective prior.

Generative adversarial networks (GANs) are a hot research topic in facial analysis, such as face super-resolution, face synthesis and manipulation, and video processing. Many papers [167, 168, 169] summarized a large number of studies on GANs. Brock Andrew et al. [170] presented an Introspective Adversarial Network, which is a novel hybridization of VAE and GAN to do face editing. Berthelot et al. [171] proposed a new equilibrium enforcing method paired with a loss derived from the Wasserstein distance for training auto-encoder based on GANs called Boundary Equilibrium GAN (BEGAN). Tran et al. [172] proposed Disentangled Representation learning-Generative Adversarial Network (DR-GAN) for pose-invariant face recognition. The work by SRGAN [173], and ESRGAN [174] introduced GANs methods for face image super-resolution. Huang et al. [175] propose a Two-Pathway Generative Adversarial Network (TP-GAN) for photorealistic frontal view synthesis by simultaneously perceiving global structures and local details. Zhao et al. [176] proposed a novel Dual-Agent Generative Adversarial Network (DA-GAN) for profile view synthesis. Shen et al. [177] generate identity preserving faces by proposing FaceID-GAN. FSRGAN [178], and

Super-FAN [179] are face Super-Resolution GAN networks that rely on prior knowledge, like facial landmark heatmaps and parsing maps, which can improve the quality of low-resolution facial images. Karras et al. [180] proposed the StyleGAN architecture that takes per-block incorporation of style vectors (defined by a mapping network) and stochastic variation as inputs, instead of samples from the latent space, to generate a synthetic image. Shaham et al. [181] proposed SinGAN that fine-tunes a randomly initialized GAN on patches of a single image in different scales, achieving various image manipulation or restoration effects. APDrawingGAN [182] is proposed to generate artistic portrait sketches from face photos with hierarchical GANs. A novel framework termed MaskGAN [183], enabling diverse and interactive face manipulation.

2.5 Summary

Although there are many related types of face research, this paper is primarily for face alignment and 3D face reconstruction. First, this chapter introduces the definition of face alignment and each of the three methods (3d model fitting-based method, coordinate regression-based method, and heatmap regression-based method) that exist for face alignment in detail. Then gives the introduction of 3D face reconstruction and its associated research methods. Finally, this chapter summarizes the 3D face generation model (FaceGen) and the 3D Morphable Model (3DMM) for the synthetic datasets involved in this thesis work. For the study of synthetic datasets transformation, this chapter describes the principles of the Generative Adversarial Network model (GAN) and its related research methods. In summary, Chapter 2 concentrates on analyzing previous studies that are related to this work.

Chapter 3

Synthetic data for face alignment

Synthetic data refers to computer-generated data by a generator model, rather than data measured and collected from real-world environments. The data is anonymous and is created based on user-specified parameters. This chapter aims to investigate for producing synthetic for training face alignment algorithms. And how it compares to using real-world images.

As noted in section 1.3 in Chapter 1, the benefit of using synthetic data is that it minimizes time, cost, and risk. Synthetic datasets do not require manually annotated data, which reduces the burden of both the data collection and labeling process. In particular, when real-world data sets are insufficient to train models effectively, large amounts of synthetic data sets can ensure comprehensive training. For face-related tasks, the training set for faces in real-world scenarios is insufficient, and most of the publicly available datasets are front faces. In other cases, real-world data can not be used for testing, training, or quality assurance due to face privacy concerns and business value. This data is sensitive or is only applicable to highly regulated industries. These reasons have led researchers to focus on synthetic datasets.

3.1 Dataset

The majority of publicly available face datasets have only small or medium angles used for training. As summarized in section 2.1 in Chapter 2, the facial keypoint annotations range from 4 points to 194 points. In this work, public datasets with 68 point annotations are used as training datasets. The 300W [184] training set consists of 3148 face images from AFW [3], HELEN [4] and LFPW [5], each of them with 68 landmarks. The test set consists of faces images from MENPO [1].

As shown in Fig. 3.1, the different rows represent real-world face images from AFW, HELEN, and LFPW, respectively. It can be observed clearly that publicly available datasets are mostly frontal face images, meaning that most faces have an angle of less than 45°.

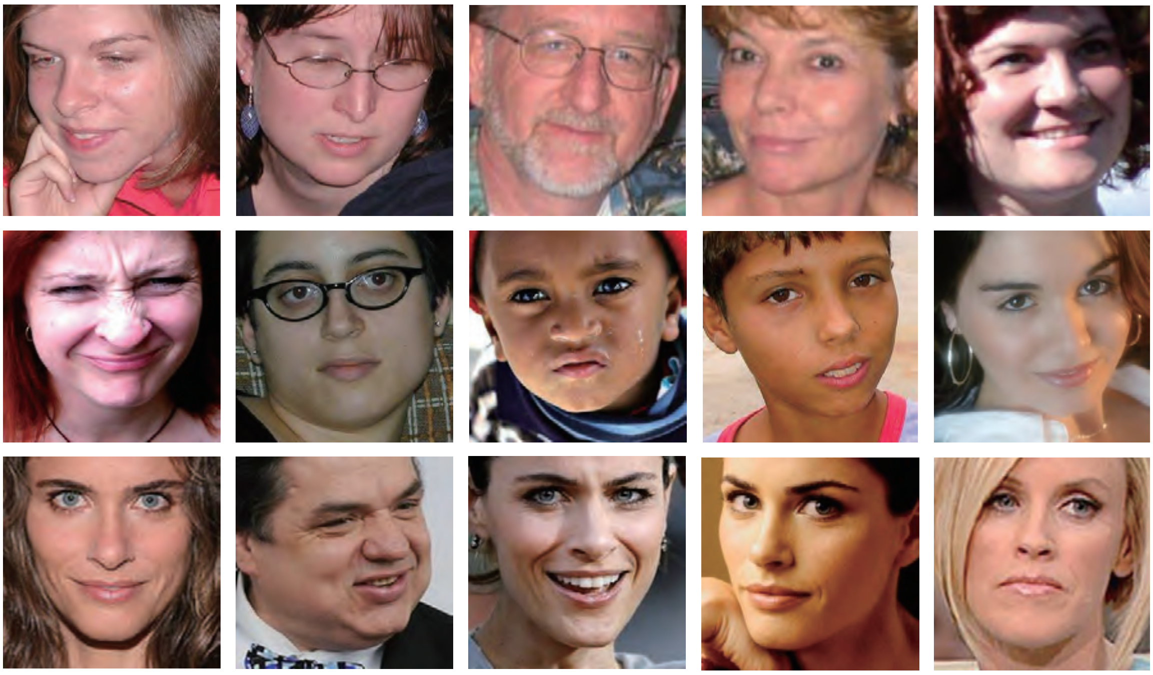


Figure 3.1: Examples of 300W public datasets.

Manually annotating a face database with 68 key points is a highly time-consuming procedure that requires an enormous workload and a trained expert. Factors like fatigue and lack of concentration are reasons why the annotations are inaccurate in some cases. It highlights the need to create synthetic datasets with automatic annotation.

As discussed in section 2.3 in Chapter 2, Fig. 3.2 shows synthetic training examples. All synthetic face images are generated using FaceGen model. The advantages of using synthetic data are that it circumvents the face privacy risk and automatically generates any number of face images and 68 key points annotations of faces, avoiding issues caused by manual annotation. It can generate large pose images, which means that it can obtain training sets with face angles greater than 45°.

Fig. 3.2 shows randomly generated different faces by adjusting FaceGen's gender control (female and male), ethnicity control (African, East Asian, European, and South African), expression control (near, surprise, smile, anger, disgust).

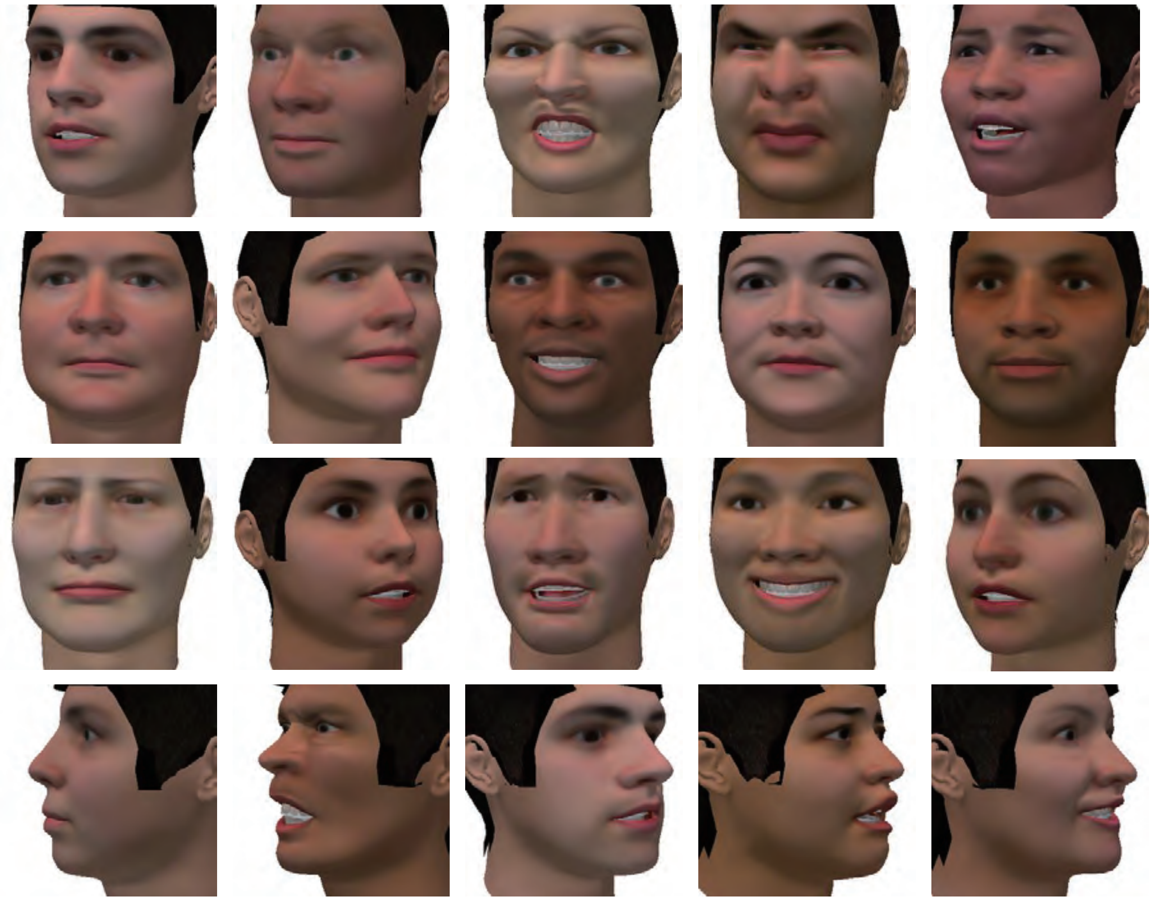


Figure 3.2: Examples of generated synthetic datasets.

3.1.1 Data Augmentation

Data augmentation makes some transformations on the original data to create more data. The advantage is increasing the amount of data, enriching the data diversity, and improving the generalization ability of the training model. Augmentation is based on a priori knowledge, and the strategy will be different for different tasks and scenarios. It aims to achieve a more comprehensive data representation, thus reducing the gap between the validation and training sets and the final test datasets, which allows the network to better learn the data distribution on the dataset.

We use some geometric transformations (e.g., flip, crop, rotation) in this experiment. Especially for the rotation of synthetic dataset, each rotation axis of the 3D rotation has a rotation matrix form. We can get different combined rotation matrices depending on the rotation angle. Then we can acquire 2D images at different angles (e.g., looking up or down faces) through rotation matrices.

Fig. 3.3 shows the training datasets (Synthetic and Real) examples after the rotation method.

You can define your rotation angle and augmentation number according to your task requirements. Here, we obtain 2D images by 3D projection every 15 degrees, and the number of augmentations for each image is 11.

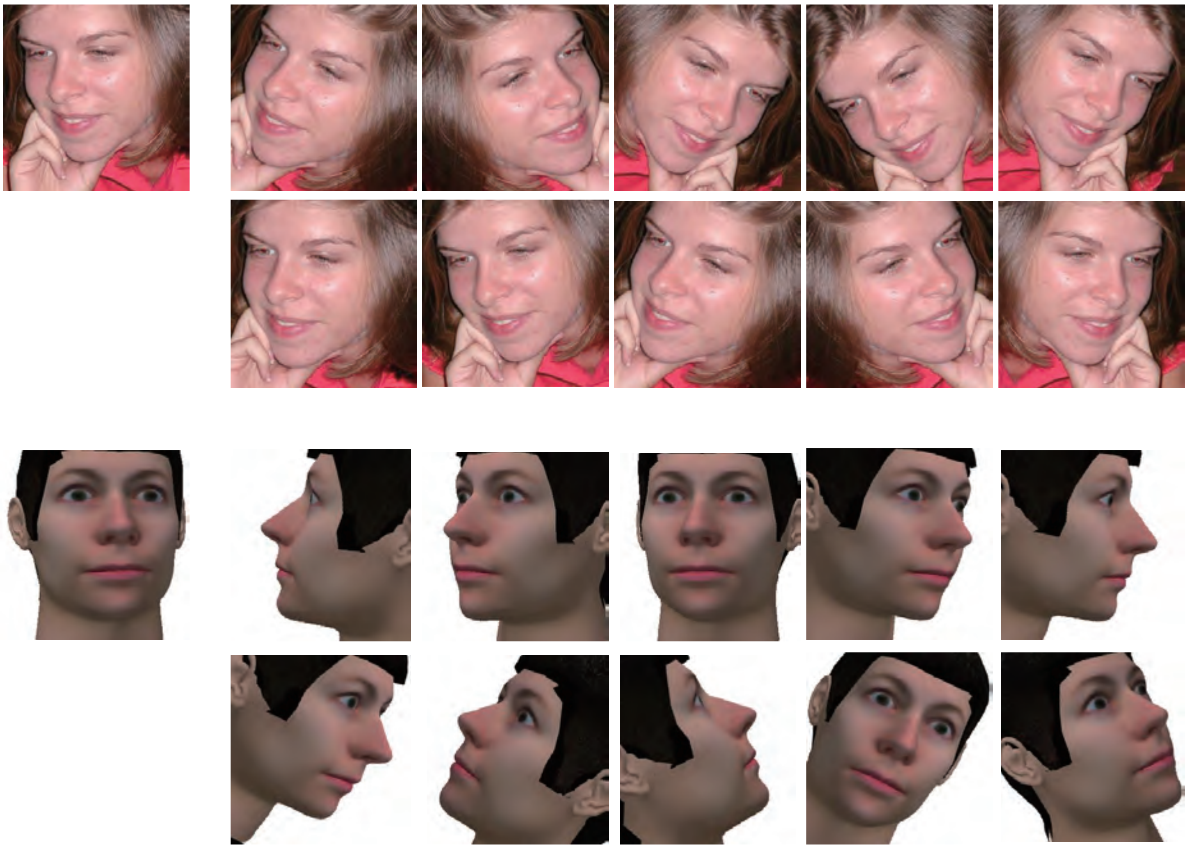


Figure 3.3: Examples of augmented synthetic and real-world datasets.

In summary, Table 3.1 lists five datasets to train the face alignment model. (1). Real 300W datasets (R) (2). Synthetic datasets generated by the FaceGen model (S) (3). Real 300W dataset plus synthetic datasets (R+S) (4). Augmentation Real 300W dataset (RA) (5). Augmentation Real 300W dataset plus synthetic datasets (RA+S).

Table 3.1: Dataset prepared for face alignment experiments.

	Nums	Augumentation	Dataset Source	Face Posture
R	3,148	No	300W	front face
S	86,412	No	FaceGen	front/side face
R+S	89,560	No	300W/FaceGen	front/side face
RA	34,628	Yes	300W	front face
RA+S	121,040	Yes	300W /FaceGen	front/side face

3.2 Method

3.2.1 Network architecture

Fig. 3.4 shows the proposed face alignment pipeline of Chapter 3 network. As the figure shown, we used the Facegen generator model to create the randomly synthetic datasets. In the preprocessing part, the synthetic dataset is combined with real-world images, and some augmentation algorithms are done. Then we used these datasets to train our landmark model. Finally, we use the real challenging data to evaluate our trained model.

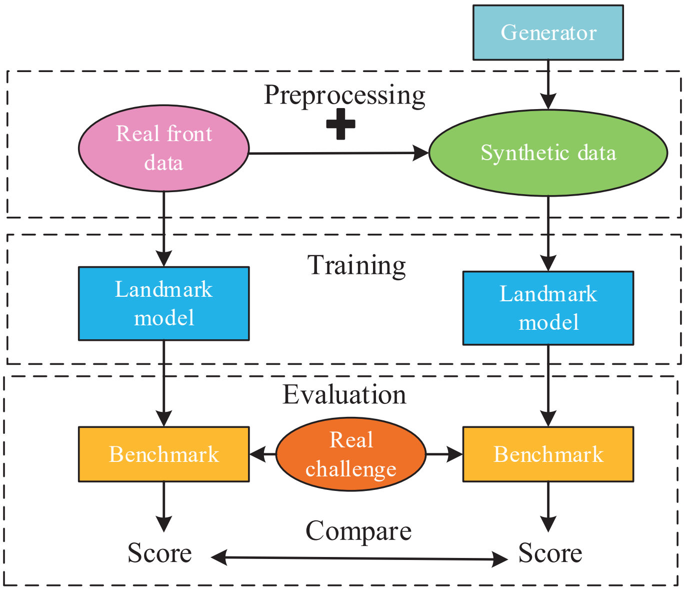


Figure 3.4: The proposed face alignment pipeline of Chapter 3.

About landmark model, heatmap regression, which regresses a heatmap generated from landmark coordinates is used widely for face alignment [82, 91, 185, 186]. The essence of heatmap regression is to output a Gaussian distribution centered at each ground-truth landmark. Inspired by the recent successes of heatmap regression based on the structure of HourGlass (HG) [107, 187], which is an asymmetric top-down and bottom-up fully convolutional network.

Fig. 3.5 describes the overall of our architecture for landmarks model. For a huge of training data, if the storage matrix of the picture is directly used as the image feature to perform various operations, there is a certain amount of redundant information. The network first uses several Residual blocks [132] to reduce the dimensionality of the image data, or called feature extraction, to store important information about the image.

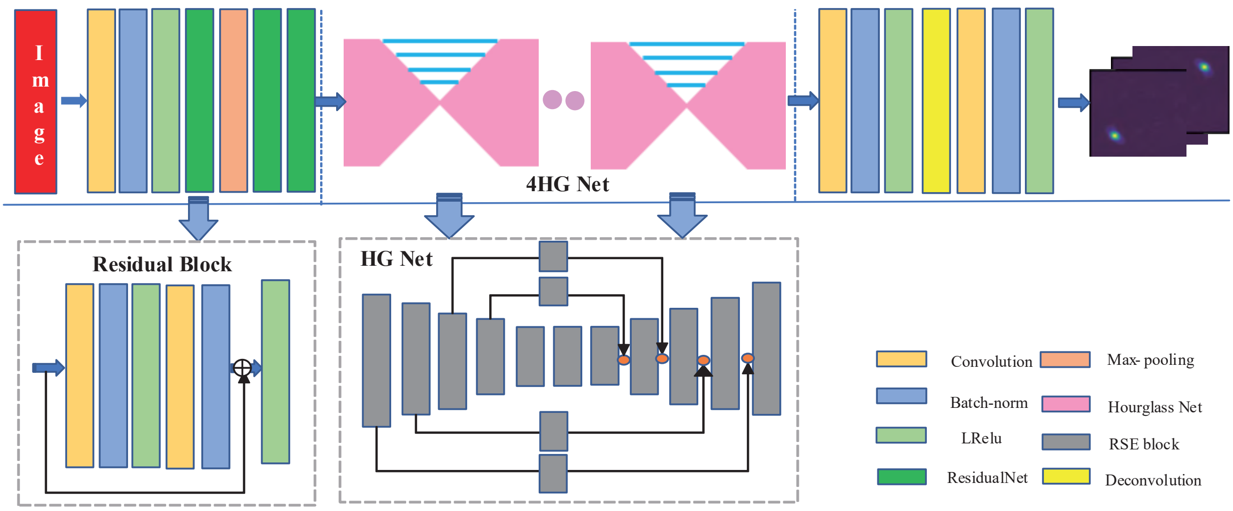


Figure 3.5: The network architecture for landmark model.

For each reconstruct RSE block, we applied the Squeeze-and-Excitation Network (SE) module [88] to the residual branch. RSE block replace each traditional residual block of HG network. By using the attention mechanism, important information filters out from a large number of pieces of information. Fig. 3.6 shows the differences between residual block and the reconstruct RSE block.

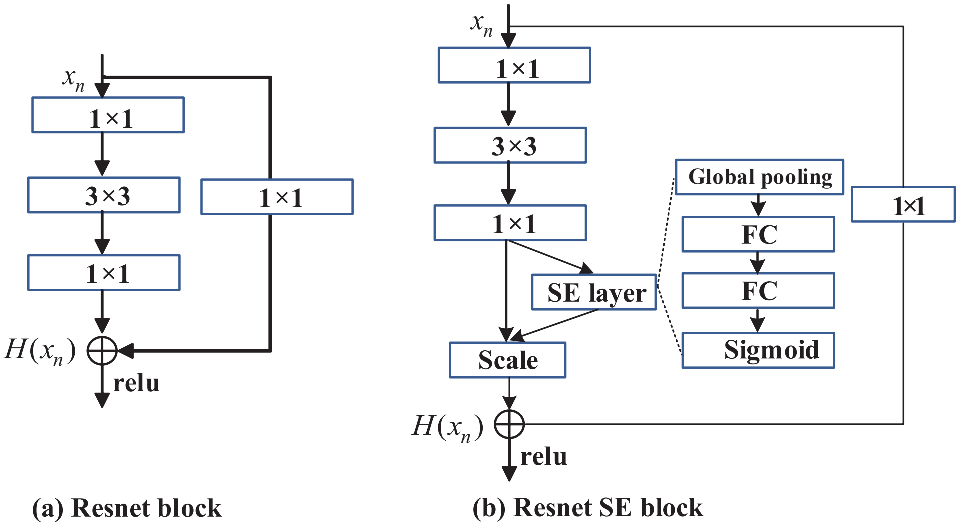


Figure 3.6: The different blocks to reconstruct the Hourglass model.

Residual blocks use a shortcut connection that connects the input of the layer directly to the output. The input and output of the unit are x_n and $H(x_n)$. A given RSE unit obtains $F(x)$ by processing x with the weight-sharing layers and SE layer.

3.2.2 Loss functions

During the training, the loss function can be calculated as follow:

$$L = \frac{1}{N} \sum_{n=1}^{N} \left(\sum_{i,j} ||p(i,j) - g(i,j)||_2^2 \right) \tag{3.1}$$

where $p(i,j)$ and $g(i,j)$ are the predicted heatmap and ground truth heatmap at pixel location (i,j) respectively. N is the number of landmarks.

3.3 Experiments

3.3.1 Experiments metric

Performance of algorithms for face alignment can be evaluated by the Normalized Mean Error (NME), the average Euclidean distance between the ground-truth and the predicted landmark normalized with the distance d_i.

$$NME = \frac{1}{M} \sum_{i=1}^{M} \frac{\frac{1}{N} \sum_{j=1}^{N} ||p_{(i,j)} - g_{(i,j)}||_2}{d_i} \tag{3.2}$$

where M and N are the number of images and the number of landmarks respectively. d_i is the normalize distance (such as the distance between the eye centers as shown in Fig. 3.7 (a) [59, 91], the distance between the outer eye corners as shown in Fig. 3.7 (b) [11] and diagonal distance of ground truth bounding box [188]).

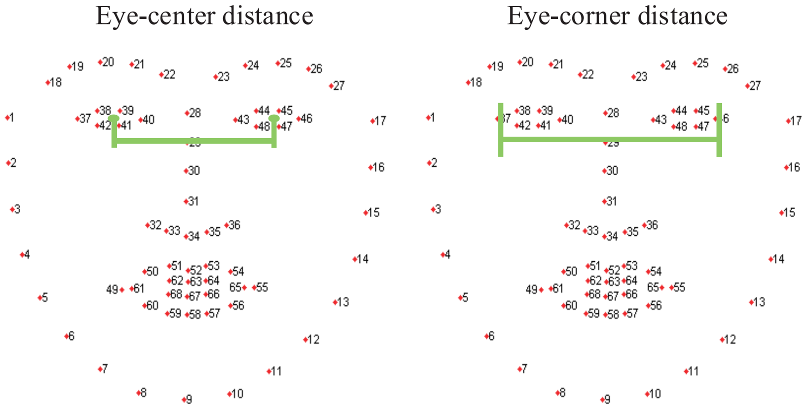


Figure 3.7: Two normalization methods for the face.

Menpo [1] is a large-scale dataset, where the images contain both front and side faces varying in illuminations, poses, and occlusions. This dataset uses as a validation set to verify the generalization ability of the algorithm.

The face alignment algorithm evaluation was performed using two different metrics: Accuracy (ACC) and Area Under Curve (AUC), which all based on NME calculations.

Accuracy (ACC) is calculated based on the NME value, which is the percentage of NME below the threshold value.

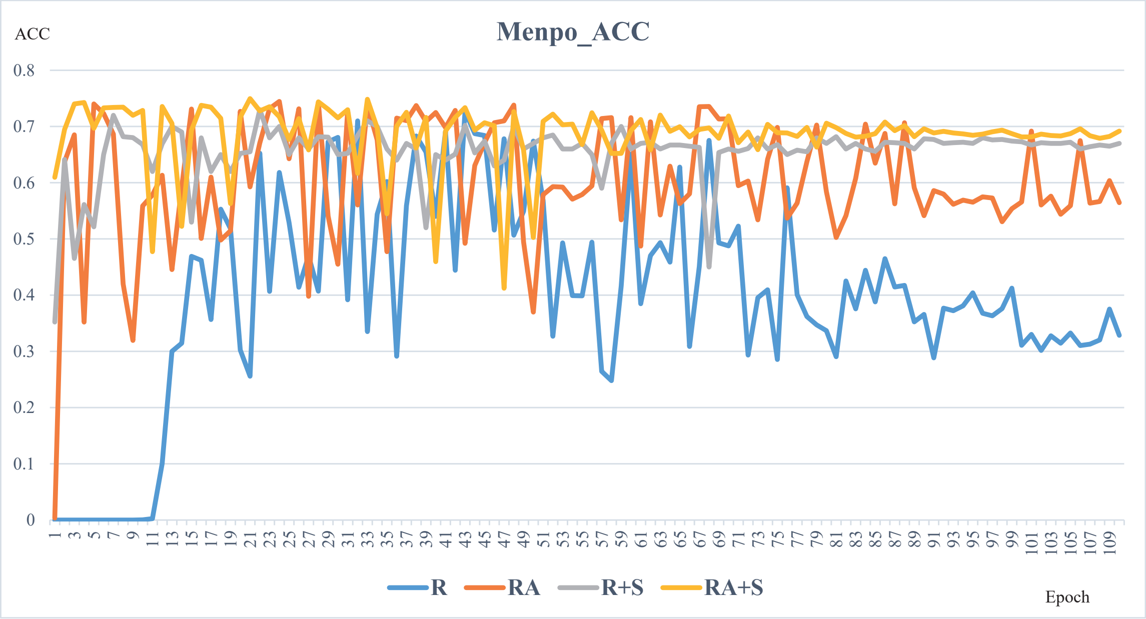


Figure 3.8: The ACC value of different training datasets.

Fig. 3.8 shows ACC values for the validation dataset (Menpo) during the model training process. In this experiments, ACC value is the percentage of NME below the threshold (0.07) value. If only using the synthetic dataset (S) to train the face alignment network. The value of NME in the validation real-world challenging set is about 26% to 30% during the whole training process, so the value of ACC is always equal to 0. Therefore, the model with dataset S is not shown in the above figure. It can easily see that the RA+S model (orange line) performs well than others, and R model is the worst. Results show that augmentation datasets can improve the ACC value (RA (blue line) > R(green line)).

Experimental results suggest that adding synthetic datasets to real datasets can improve detection accuracy, especially for challenging datasets (Menpo). The interpretation of this observation is that the FaceGen generators model can simulate pose changes and changes in facial expressions. However, the dataset R hardly covers the full range of these changes in real-world datasets.

Area Under Curve (AUC) is calculated based on the CED curve [189]. AUC_ϵ as the area under the CED curve for images with normalized errors smaller than ϵ (e.g., 10%, 8%, 7%). The range of the AUC_ϵ values is always 0 to 1.

$$AUC_\epsilon = \int_0^\epsilon f(e)de \tag{3.3}$$

where e is the normalized error, $f(e)$ is the cumulative error distribution (CED) function, and ϵ is the upper bound used to calculate the definite integration. The value of AUC_ϵ will not be influenced by points with an error bigger than ϵ [189].

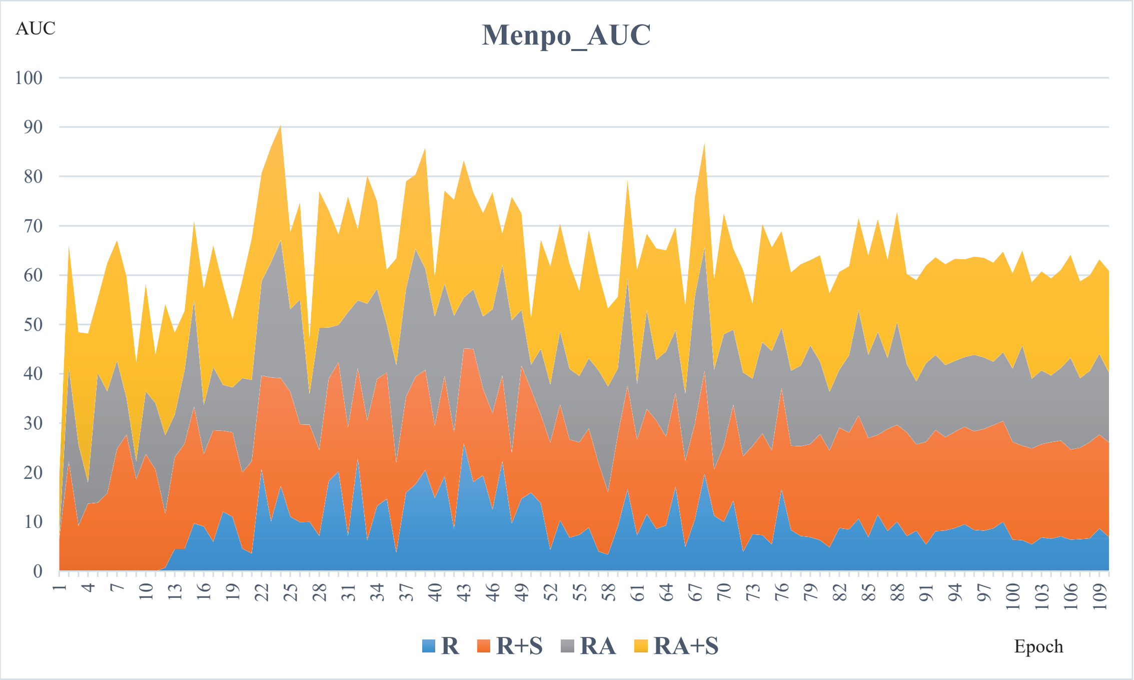


Figure 3.9: The AUC value of different training datasets.

A higher AUC value reflects that a large portion of the dataset predicts well. Fig. 3.9 draws AUC values for the validation dataset (Menpo) during the model training process. If only using the synthetic dataset (S) to train the face alignment network. The value of AUC in the validation real-world challenging set is about 4% to 7%, the value is too small to be negligible. So the model with dataset S is not shown in the above figure. AUC score also showed that adding synthetic datasets performs better at dealing with the same Menpo dataset.

3.3.2 Experinment results

As for the 300W test datasets, the datasets are divided into four parts based on the other prior work [190]. The first subset, called the Common subset, consists of the LFPW test dataset and Helen (554 images). The second subset, named the Challenging subset, includes the i-bug dataset (135 images). The third subset contains all 300W public test-set. The fourth subset consists of a 300W private test-set (600 images).

Fig. 3.10 shows some face alignment results from 300W testing datasets compared to the different model's (S/R/RA/R+S/RA+S). And the rectangle and arrow on the picture show comparisons between each model.

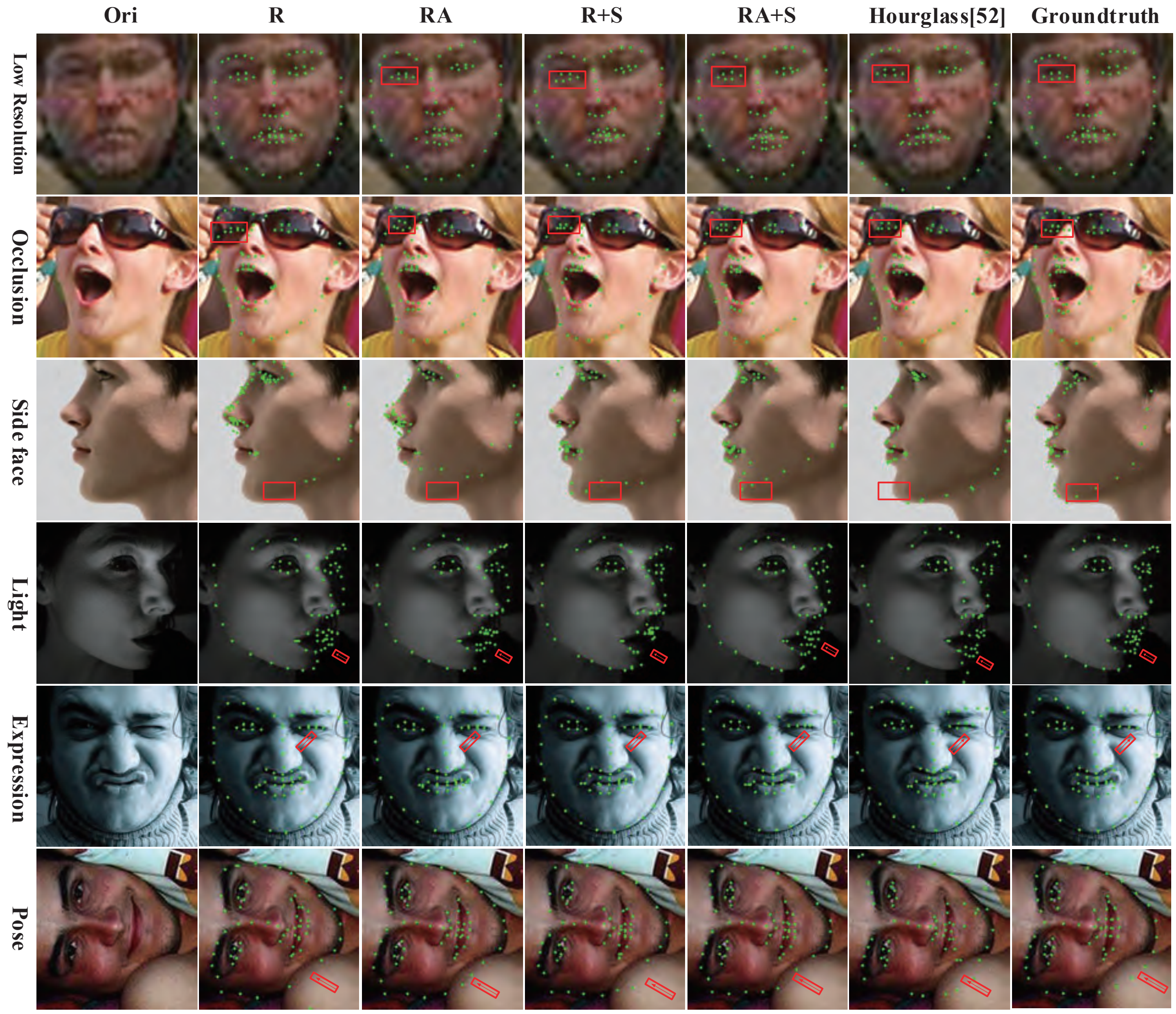


Figure 3.10: The experiment results comparing various train models.

Compare to the different model's results, the RA+S model performs well than others, even in the unconstrained environment (e.g., blur, occlusion, light, and pose). The experiment also compares the results of RA+S datasets train other SOTA methods like PFLD [101] and Hourglass model [58]. Visualization results further demonstrate that adding synthetic datasets to real datasets can improve detection.

Based on the other prior work [190], Face alignmnt method is evaluated on the 300W Public test datasets. It has been used extensively for face alignment that consists of the HELEN, LFPW, AFW, and IBUG datasets, with 68 landmark annotations for each face image.

300W Public: includes the Common subset, which is composed of the LFPW test dataset as well as Helen (554 images), and Challenging subset, which contains the i-bug dataset (135 images). The Full set consists of a common dataset and a Challenge set containing 689 images.

For comparison, Table 3.2 reports the NME results of 300W public datasets.

Table 3.2: NME errors for face alignment methods on the 300W public dataset.

Method	300W public datasets		
	Common	Challenging	Full
Inter-pupil Normalization (%)			
RCPR [7]	6.18	17.26	8.35
CFAN [93]	5.50	16.78	7.69
ESR [23]	5.28	17.00	7.58
SDM [191]	5.57	15.40	7.50
LBF [59]	**4.95**	11.98	6.32
3DDFA [9]	6.15	10.59	7.01
ECSAN [192]	5.42	11.80	6.69
S	26.73	43.01	29.92
R	6.74	10.57	7.49
RA	6.18	9.64	6.86
R+S	5.47	9.86	6.33
RA+S	4.98	**9.28**	**5.82**

Some methods (e.g., LBF [59]) get the best-reported result on the common test datasets that are almost front face. The proposed method (RA+S) improves upon previous approaches when applied to challenging datasets.

3.4 Summary

For the challenges of real-world face datasets mentioned in Chapter 1, this chapter focuses on using synthetic datasets in the face alignment algorithm. The experimental results demonstrate the great potential of synthetic data to analyze and reduce the effects of dataset bias on face alignment. To discuss whether the synthetic dataset generated by the generative model software FaceGen can be used directly as a dataset for alignments, this chapter designs five different datasets (S/R/RA/R+S/RA+S) and conducts comparative experiments. The overall results indicate that adding synthetic datasets to train the model can improve facial landmark detection accuracy for challenging datasets (e.g., low resolution, occlusion, and large pose) to some extent, particularly for side face landmark detection.

Chapter 4

Synthesizing data-(Real to Synthetic) for face alignment

Even adding synthetic datasets can improve the accuracy of side face landmark detection, unfortunately, side face detection ability is still limited under complex situations because the ”domain gap” among synthetic datasets and real-world datasets is still a challenge (see in Fig. 4.1). Thus, this chapter aims to improve the sense realism of synthetic data and narrow the ”domain gap” between synthetic datasets and real datasets. A face generation model based on CycleGAN that can preserve characteristics of synthetic and real-world datasets is designed. After that, the transformed datasets are used to evaluate the face alignment.

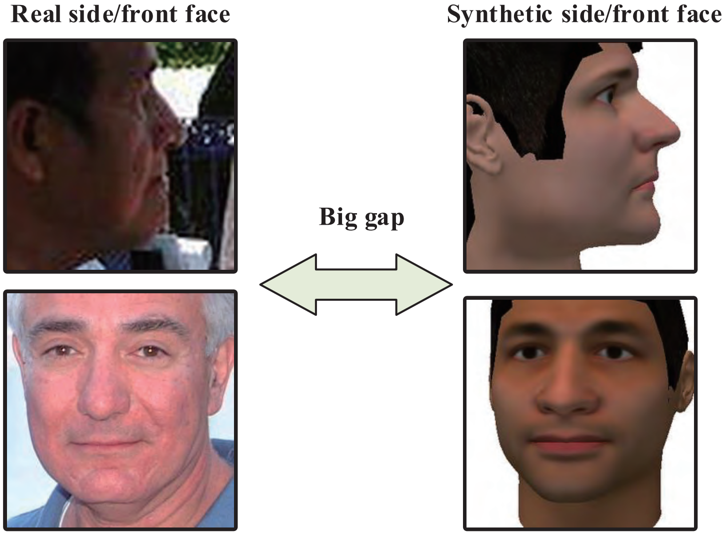


Figure 4.1: The ”domain gap” between synthetic and real-world datasets.

It is seen from Fig. 4.1 that texture and facial parts (mouth, eye, hair, and eyebrow) have a big gap between real-world and synthetic datasets. Second, the synthetic dataset has no background, but the real dataset has complex interference scenes in the real world. For these issues, this chapter proposes a translation model $R \rightarrow S$ to narrow the domain gap.

4.1 Method

4.1.1 Network architecture

Fig. 4.2 shows the whole pipeline of face alignment compared to Chapter 3 network. This design can solve the face alignment of the side face even if training datasets only have real front faces and synthetic datasets. The idea is to use translation model $R \rightarrow S$ to convert the side face first, then test the RA+S model.

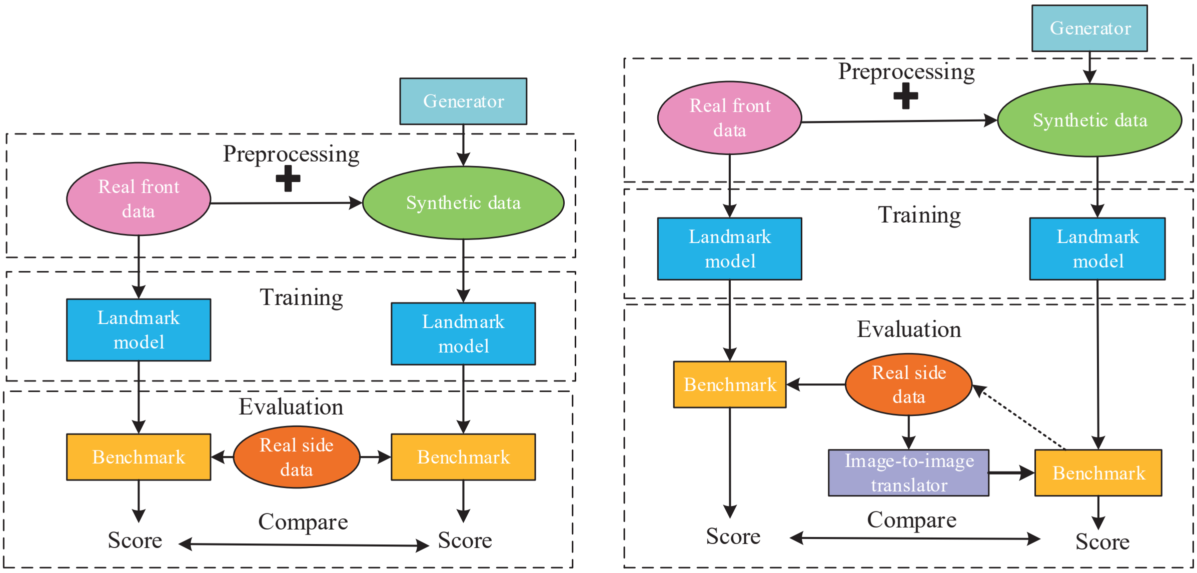


Figure 4.2: The proposed face alignment architecture compared to Chapter 3.

The landmark model has already been described in section 3.2.1, so the following part only introduces the Image-to-Image translator network in detail.

In recent years, Generative Adversarial Networks (GANs) have been the focus of much attention and have been used to produce astonishing results in synthetic image generation. The key idea is that both generator and discriminator are trained with an adversarial relationship. The generator and discriminator play a two-player zero-sum game where the Nash equilibrium corresponds to the generator producing samples indistinguishable from real-world data [84]. CycleGANs are a novel approach for translating an image from a source domain A to a target domain B. One of the cool features of CycleGANs is that it does not require paired training data to produce stunning style transfer results. The Image-to-Image translator network adopts CycleGAN [163] network architecture with Wasserstein adversarial loss [159] to learn the mapping for the generator network.

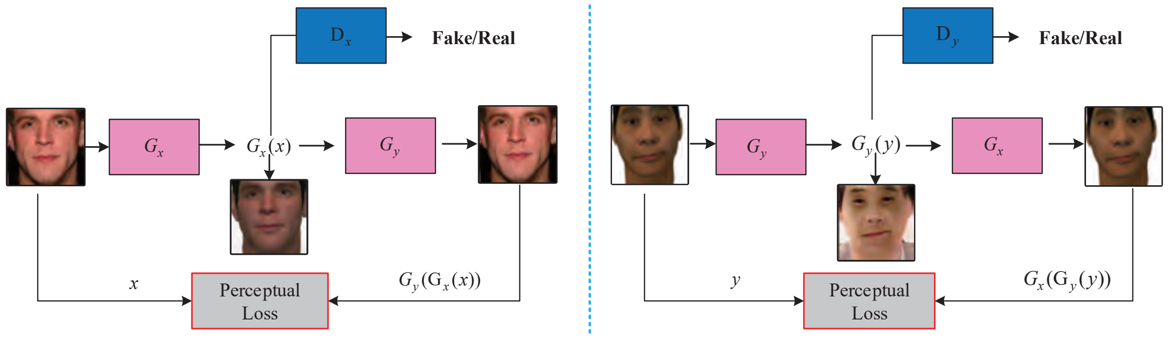


Figure 4.3: The overall architecture of the Image-to-Image translator network.

Fig. 4.3 shows the overall architecture of the proposed network with both synthetic images and real images as input. Datasets: synthetic images of $x \in X$ and real images of $y \in Y$ as well. Two generators: G_x, G_y and two discriminators: D_x, D_y.

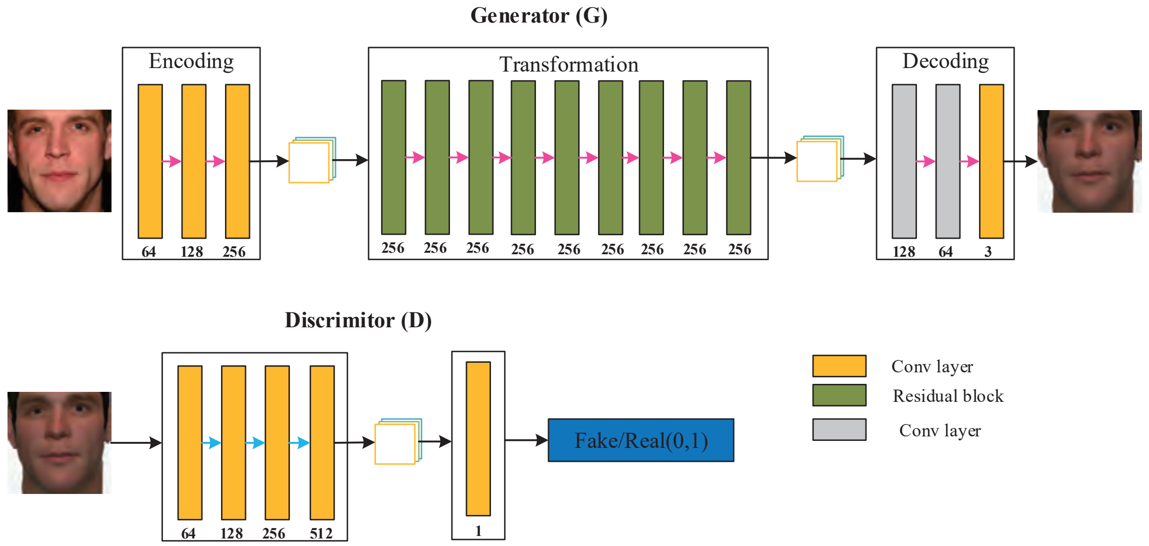


Figure 4.4: Generator and Discriminator network structures.

As shown in Fig. 4.4, the generator's job is taking an input image and performing the transformation to produce the target image. First, the encoding process consists of three convolution layers, and an activation function ReLU is used for each layer. Second, in the transformation process, nine residual blocks are constructed. Third, the decoding process consists of two deconvolutions and one convolution layer [193].

For discriminator, networks can use different patch sizes N: from 1 × 1 PixelGAN to 256 × 256 ImageGAN for the whole image. PixelGAN does not help with spatial clarity but can relatively improve the effect of color. Using 16 × 16 PathGAN improves the sharpness of the output further, but some unnatural textures appear. If increasing the size of N to 256 × 256 ImageGAN does not improve the effect. Appendix A.1 gives the detailed structure of the specific different Discriminators. This experiment uses PatchGANs [162, 163, 173, 194], which aim to classify whether 70 × 70 overlapping image patches are real or fake. It consists of four convolution layers and leaky ReLU as an activation function for each layer. The first step extracts the features from the image, and finally, the last convolution layer produces a one-dimensional output to decide these features belong to which specific category. Besides that, the instance normalization scheme [163, 195] is used for all layers of the generator and discriminator networks.

4.1.2 Loss functions

GANs try to use loss functions that reflect the distance between the distribution of the data generated by the GAN and the distribution of the real data. Let P_r be the data distribution, P_g be the model distribution defined by $\tilde{x}_g = G(z)$, where z is the input to the generator, the discriminator network D adjusts its weights to reliably distinguish real data samples $x_r \sim P_r$ from fake data samples $\tilde{x} \sim P_g$, via the generator network. The generator network G adjusts its weights to fool D. The two networks are trained iteratively using a loss function given by:

$$L_{GAN} = \min_{G} \max_{D} E_{x_r \sim\ P_r}[\log(D(x_r))] + E_{\tilde{x}_g \sim\ P_g}[\log(1 - D(\tilde{x}_g))] \tag{4.1}$$

To address GAN's problems, Mao et al. [196] proposed Least Squares Generative Adversarial Network (LSGAN), which adopt the least squares loss function for the discriminator.

$$L_{LSGAN} = \min_{G} \max_{D} E_{x_r \sim\ P_r}[(D(x_r))^2] + E_{\tilde{x}_g \sim\ P_g}[(1 - D(\tilde{x}_g))^2] \tag{4.2}$$

The adversarial loss (LSGAN) instead of the sigmoid cross-entropy loss is used in the Cycle-GAN model to learn a mapping function that the distribution of images from G_x is indistinguishable from Y. This loss can guarantee the samples generated by the generator have the same distribution as the real-world samples.

Two generators mapping G_x: $X \rightarrow Y$ and G_y: $Y \rightarrow X$ and two adversarial discriminators D_x and D_y are used to distinguish whether images translated from another domain. The cycle consistency loss defined by CycleGAN enforces these mappings should be reverses of each other, and both mappings should be bijections.

$$L_{CYC}(G_x, G_y) = E_{y\sim\ P_y}[||G_x(G_y(y)) - y||_1] + E_{x\sim\ P_x}[||G_y(G_x(x)) - x||_1] \tag{4.3}$$

CycleGAN paper also provided the identity loss L_{Iden} to ensure that the color distributions of the input and output image are similar.

$$L_{Iden} = E_{y\sim\ P_y}[||y - (G_x(y))||_1] + E_{x\sim\ P_x}[||x - (G_y(x))||_1] \tag{4.4}$$

Johnson et al. [197] have shown impressive results for neural style transfer and super-resolution. The loss function relies on the fixed style representation captured by the features of a ResNet network [132] pre-trained on ImageNet. The perceptual loss is to ensure that the high-dimensional representation of the transformed image and the ground truth image are as identical as possible.

$$\begin{aligned} L_{perceptual} = \sum_j \frac{1}{C_j H_j W_j} (||\phi_j(x) - \phi_j(G_x(x))||_2^2 + ||\phi_j(y) - \phi_j(G_y(y))||_2^2 + \\ ||\phi_j(G_x(x)) - \phi_j(G_y(G_x(x)))||_2^2 + ||\phi_j(G_y(y)) - \phi_j(G_x(G_y(y)))||_2^2 + \\ ||\phi_j(y) - \phi_j(G_x(G_y(y)))||_2^2 + ||\phi_j(x) - \phi_j(G_y(G_x(x)))||_2^2) \end{aligned} \tag{4.5}$$

Compared to the traditional CycleGAN, we explored the mapping from the input image to its ground truth through combining the LSGAN adversarial loss and the perceptual loss and presented the translation $R \rightarrow S$ model for solving image-to-image transformation tasks. The total objective function is minimized during the training process as shown:

$$L_{total} = L_{LSGAN} + \beta L_{CYC} + L_{Iden} + \lambda L_{perceptual} \tag{4.6}$$

About the selection of parameters, the first three parameters are set based on the original CycleGAN authors define in the experimental details in the article that the best results are obtained when β is taken as 10. The only parameter that needs to be adjusted is λ, which adjusted by the quality of the image generated by $R \rightarrow S$ model. $\lambda = 0.1$ are the best weight for our experiment.

4.2 Experiments

4.2.1 Experiments results

Fig. 4.5 shows front/side faces results of translation models $R \rightarrow S$ (S: synthetic R: real).

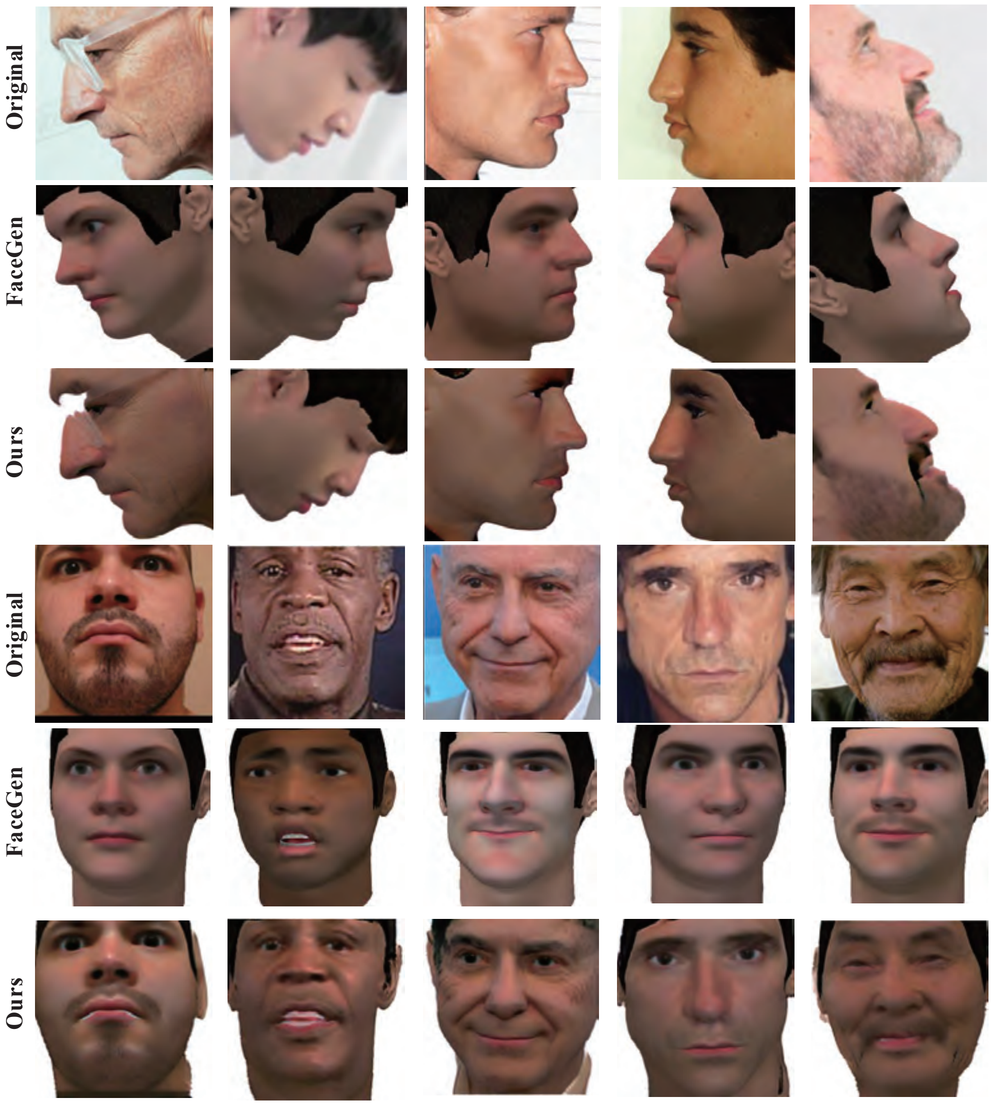


Figure 4.5: Front face results of the translation model $R \rightarrow S$.

As the Fig. 4.5 shown, the $R \rightarrow S$ model can convert the real-world face into more synthetic face images. It can be seen that the generated faces maintain the characteristics of synthetic and real-world images at the same time. Compared with the synthetic images generated directly by FaceGen, the proposed algorithm further reduces the difference between the real-world and synthetic datasets.

In our experiment, the network use Adam solver with a batch size of 1. All networks were trained with a learning rate of 0.0002. Same learning rate for the first 100 epochs and linearly decay the rate to zero over the next 100 epochs. In Fig. 4.6, each experiment uses the same network architecture, same evaluation datasets, and same test epoch to compare the proposed method against several loss functions. For example, original CycleGAN ($L_{GAN} + L_{CYC} + L_{Iden}$) [163], DCGAN ($L_{DCGAN}$) [156], LSGAN ($L_{LSGAN}$) [196] and Ours ($L_{LSGAN} + L_{CYC} + L_{Iden} + L_{perceptual}$).

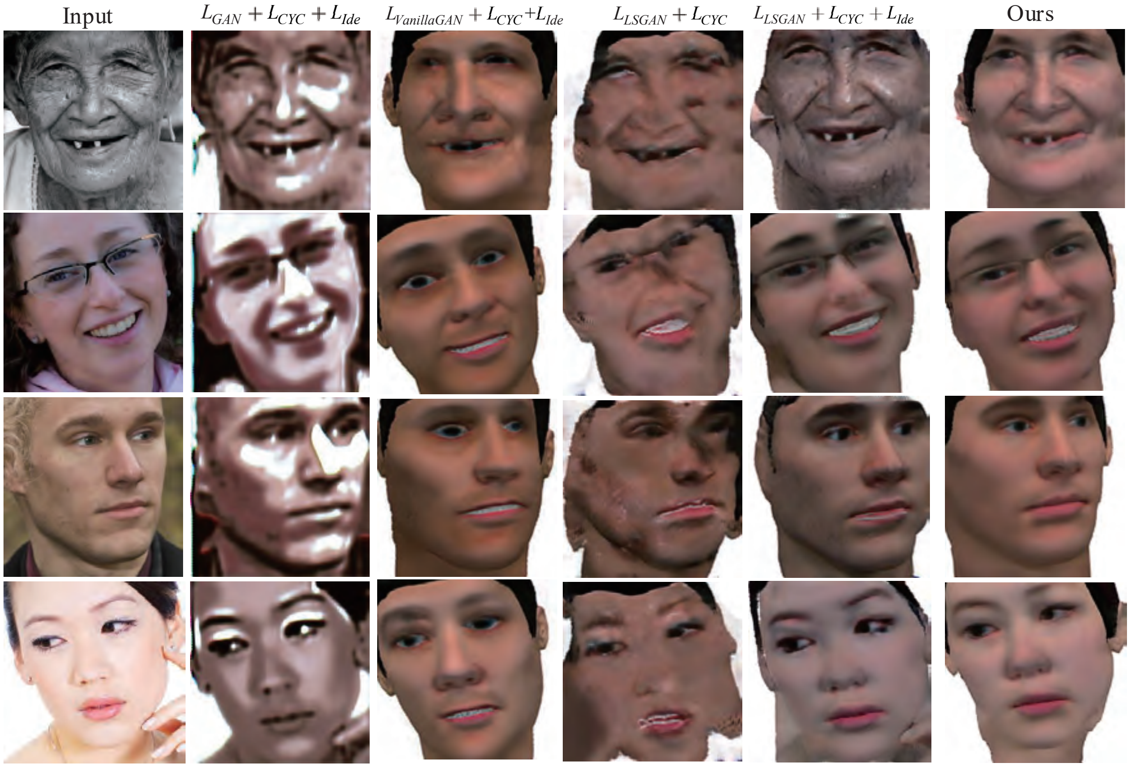

Input
$L_{GAN} + L_{CYC} + L_{Ide}$
$L_{VanillaGAN} + L_{CYC} + L_{Ide}$
$L_{LSGAN} + L_{CYC}$
$L_{LSGAN} + L_{CYC} + L_{Ide}$
Ours

Figure 4.6: Comparative results of several transformed models ($R \rightarrow S$).

As the Fig. 4.6 shown, for the test images, which vary in color, expression gesture, and brightness, our method works well for almost all of them. Comparing the results of $L_{GAN} + L_{CYC} + L_{Iden}$, $L_{DCGAN} + L_{CYC} + L_{Iden}$ and $L_{LSGAN} + L_{CYC} + L_{Iden}$ results, L_{LSGAN} shows better performance than the other generator in terms of both convergences of the generator and sample quality. We show the benefits of $L_{LSGAN} + L_{CYC} + L_{Iden}$ over $L_{LSGAN} + L_{CYC}$ results. It can generate higher quality images than$L_{LSGAN} + L_{CYC}$. Compared with the $L_{LSGAN} + L_{CYC} + L_{Iden}$ and $L_{LSGAN} + L_{CYC}$ results, our method behaves more stable during the learning process and can maintain both the content of the real-world image and the style of the synthetic.

4.2.2 Experiments evaluation

In Fig. 4.7, the first row represents the results of feeding the side face image directly into the (RA+S) evaluation model. In the second row, the side face image into the translation model $R \to S$, and then input this transformed face into the (RA+S) evaluation model, after the corresponding keypoint location information matches the original image.

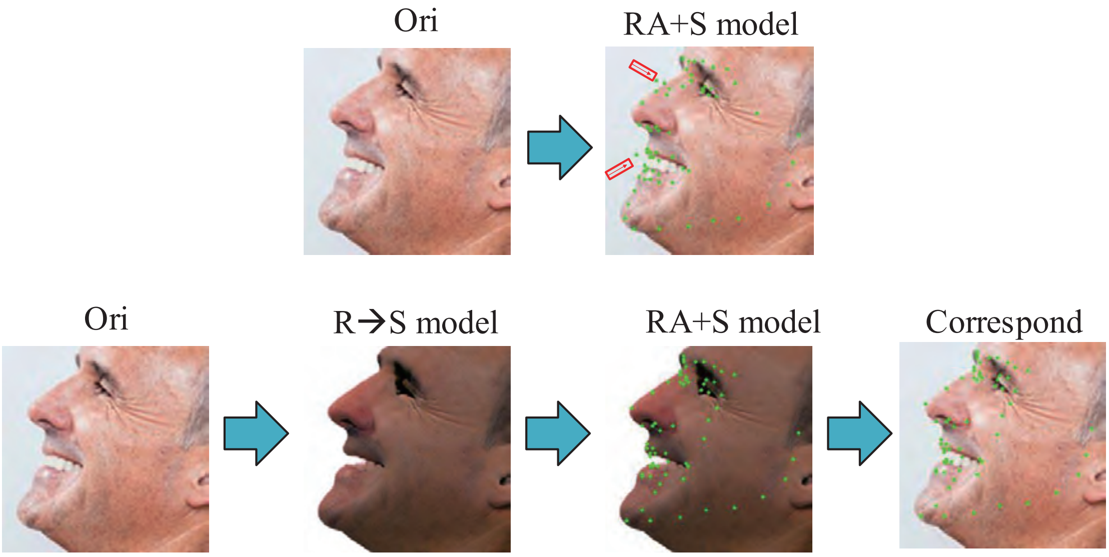


Figure 4.7: Real faces alignment comparison results with faces obtained by translation model.

300W private: was introduced after 300W datasets and was used for the 300W Challenge benchmark. It consists of Indoor (300 images) and Outdoor (300 images) with 68 landmarks using the same annotation scheme as 300W. All the images downloaded from the web, thus captured under totally unconstrained.

COFW: shows large variations in shape and occlusions due to differences in pose, expression, use of accessories such as sunglasses and hats and interactions with objects (e.g. food, hands, microphones, etc.). The training set consists of 1345 images, and the testing set consists of 507 faces with a wide range of occlusion patterns with 29 landmarks. This experiments use the reannotated version of 68 landmarks for comparison to other approaches. According to the previous works introduced by [198, 199], it has on average 28% of the landmarks occluded.

Fig. 4.8 visualizes more face alignment model comparison results.

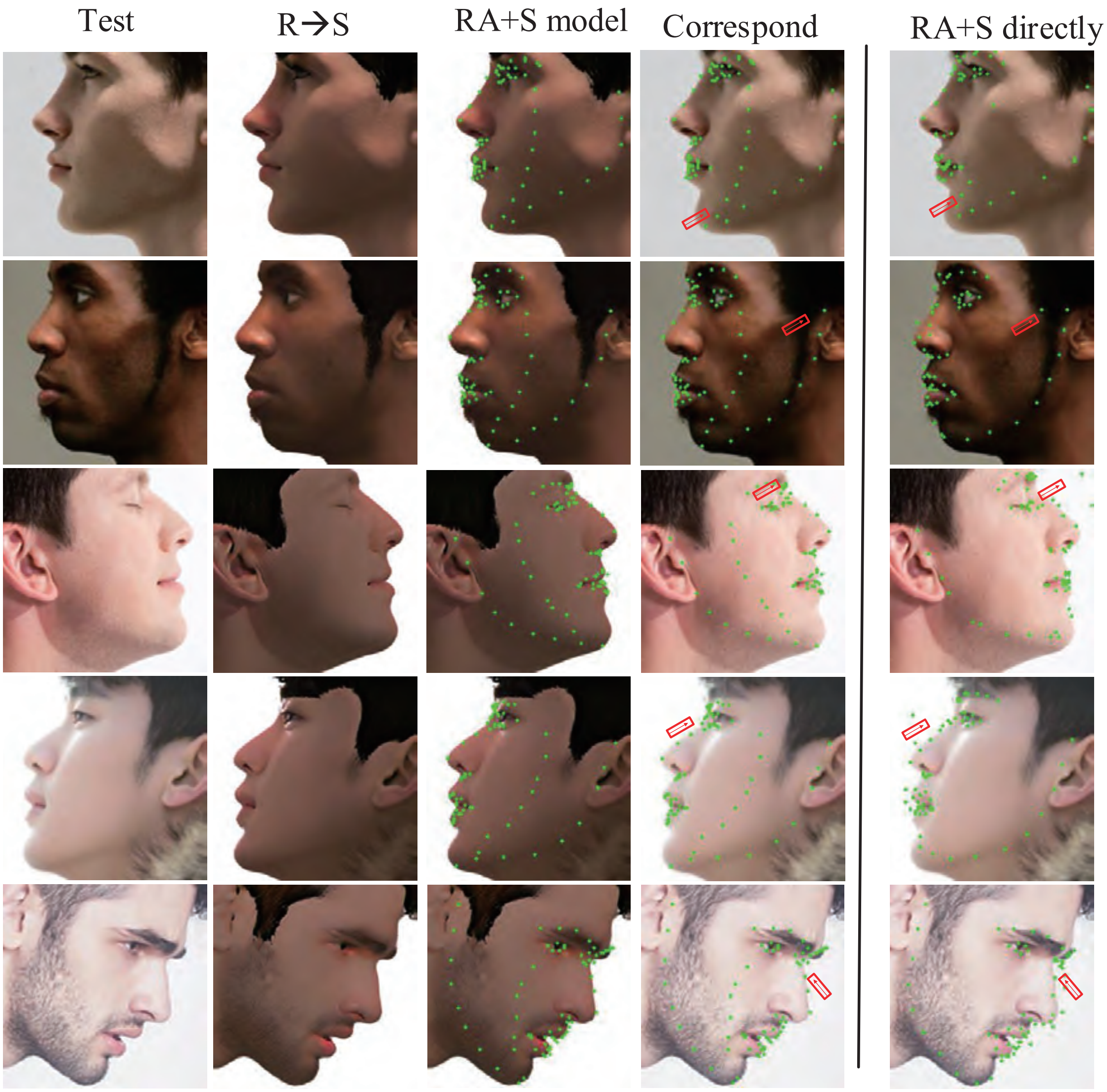


Figure 4.8: More alignment comparison results.

As the figure shows, the area indicated by the arrow in the figure highlights the two comparison effects. The results suggest that this transformation model $R \rightarrow S$ is more effective than with real-world side faces for face alignment directly.

Table 4.1 shows the Normalized Mean Error (NME) and Area Under Curve (AUC_7) value. The parameter sets $\epsilon = 7\%$ of different approaches on the whole 300W private test set and each testing subset (Indoor and Outdoor) and COFW challenging datasets. Experimental results show that the proposed method improves the correctness of face detection compared with the RA model.

Table 4.1: NME Error of face alignment methods on the 300W private and COFW dataset.

Method	300W private datasets							
	Indoor sets		Outdoor sets		Full sets		COFW	
	NME	AUC_8	NME	AUC_7	NME	AUC_8	NME	AUC_7
RCPR [7]	-	-	-	-	-	-	8.50	-
TCDCN [53]	-	-	-	-	-	-	8.05	-
LMM [200]	5.86	-	5.94	-	-	-	-	-
RA	5.64	22.71	5.97	21.23	5.80	21.97	6.03	19.63
Ours	**5.25**	**28.89**	**5.32**	**29.24**	**5.29**	**29.06**	**5.79**	**23.73**

However for side faces in complex backgrounds or large-scale expressions, the improvement is limited, and Fig. 4.9 also shows some of the bad results.

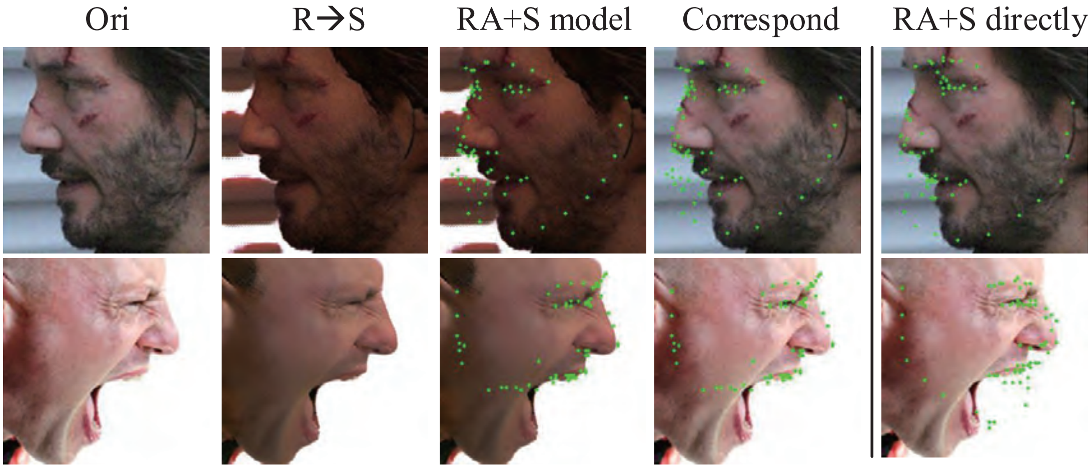


Figure 4.9: Bad results under challenging conditions.

4.3 Summary

It is easily observed that the synthetic dataset generated by the FaceGen generator has a significant gap from the real-world dataset. One question is whether synthetic datasets can be transformed with GANs models to reduce the gap with real datasets. An improved CycleGAN model for transforming real-world to synthetic images is designed to narrow the domain gap. The experimental results demonstrate that the landmark detection accuracy can be improved after this transformation model even if the data distributions of the training dataset (synthetic dataset) and the evaluation dataset (real dataset) are different.

Chapter 5

Synthesizing data-(Synthetic to Real) for face alignment

This chapter focuses on the following question: ”Can image-to-image translation GANs be used for data augmentation methods to introduce variations in the data? ”. This work aims to convert the synthetic face images generated by the FaceGen 3D model into more realistic face images for training face alignment algorithms.

5.1 Dataset

CycleGAN based methods showed to perform well for style transfer tasks mapping local texture but usually fail for image-translation with various shape changes in the wild images. Therefore, some processing steps such as image segmentation and alignment are often required to avoid these problems by limiting the complexity of the data distribution.

5.1.1 Preprocessing-remove background

In previous semantic segmentation networks, the segmentation results are often coarse for two main reasons, one is the loss of information due to pooling, and the other is that the failure to exploit the probabilistic relationships between labels. For these two points, the DeepLab network suggests targeted improvements. Firstly, DeepLab uses Atrous Convolutions to avoid information loss caused by the pooling layer, and then uses CRF (Conditional Random Field) to further optimize the segmentation accuracy.

This experiment used the pre-trained DeepLab model [201, 202] for the dataset preprocessing to remove face image background. Fig. 5.1 draws the image segmentation process and intermediate quantities used.

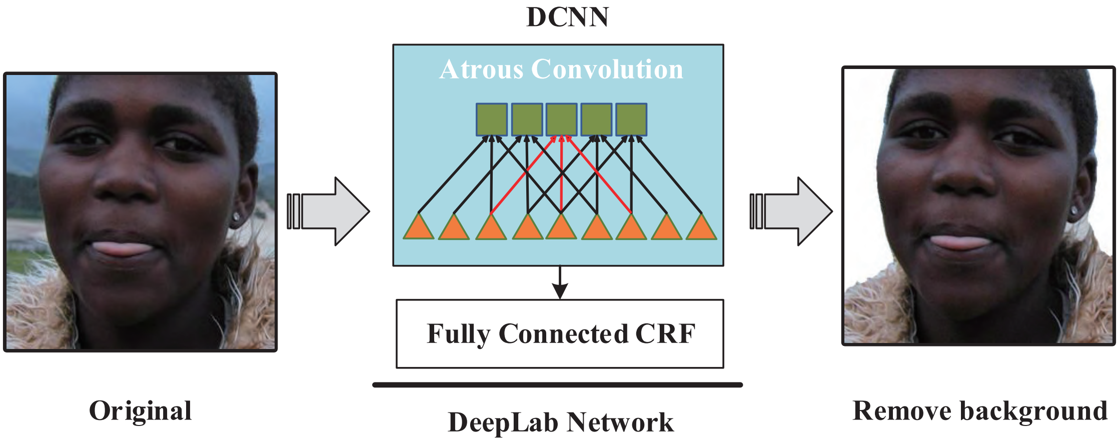


Figure 5.1: Process of removing the background algorithm.

The main function of the Atrous (Dilated) convolution is to increase the perceptual field without increasing the number of parameters and maintain resolution. A Conditional Random Field (CRF), where each pixel point is a node, and pixel-to-pixel relationships use as edges, then constitutes a conditional random field. The relationship between pixels is described by a binary potential function, which encourages similar pixels to be assigned with the same label. While pixels with large differences are assigned with different labels, and the definition of this ”distance” is related to the color value and the actual relative distance.

Fig. 5.2 shows the training datasets sample of translation model $S \rightarrow R$.

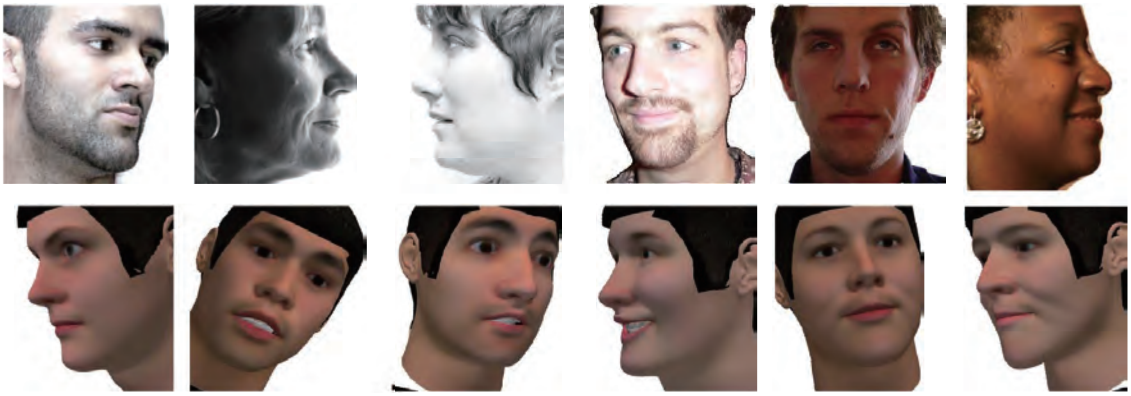


Figure 5.2: The training datasets sample of translation model $S \rightarrow R$.

5.2 Method

5.2.1 Network architecture

Fig. 5.3 shows the whole pipeline of face alignment compared to Chapter 3 and Chapter 4 network. Compared to Chapter 3, the difference is added $S \rightarrow R$ translation model in our training part. Regarding the model of Image-to-image translator, Chapter 4 utilizes $R \rightarrow S$ translation model in the evaluation part. Our design in Chapter 5 is to use the $S \rightarrow R$ translation model to transfer synthetic datasets, then these transferred realistic images for training.

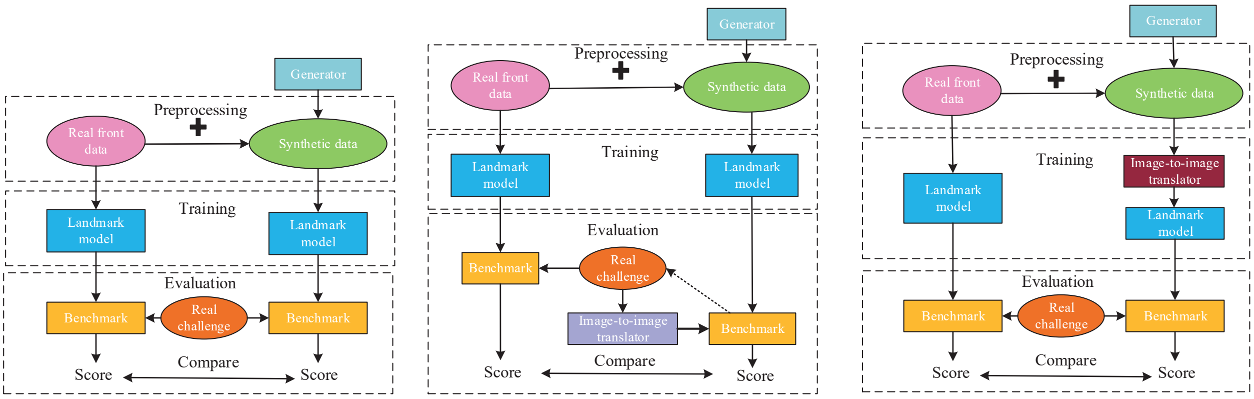


Figure 5.3: The proposed pipeline compared to Chapter 3 and Chapter 4.

Fig. 5.4 shows the detailed structure of the face alignment model. Synthetic data from FaceGen generated Real_CG data by a $S \rightarrow R$ model based on UGATIT GAN. Then those data plus real-world data as training datasets to train the face alignment model. In Chapter 3, section 3.2.1 and section 3.3.1 describe the details of the face alignment algorithm and evaluation metrics parts. Next, we focus on introducing the algorithm of pre-trained UGATIT GAN.

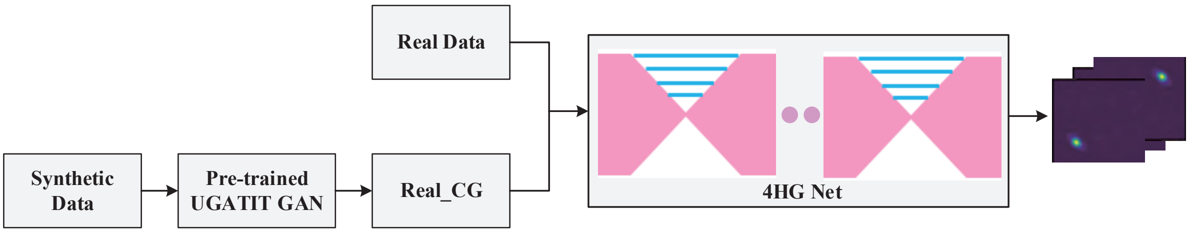


Figure 5.4: Face alignment model overall structure.

5.2.2 Pre-trained UGATIT GAN

For the $S \rightarrow R$ translation model based on the method of UGATIT [165], which is also an improved variant of Cyclegan, Minivision's project (A.3 [n.]) suggested employing two hourglass modules before encoder and after decoder, which can improve the performance in a progressively way. The attention mechanism are added in ResNet block (see Fig. 3.6), which can emphasize valid information and suppress invalid information by weighting the channels. Fig. 5.5 illustrates the detail of the Generator structure.

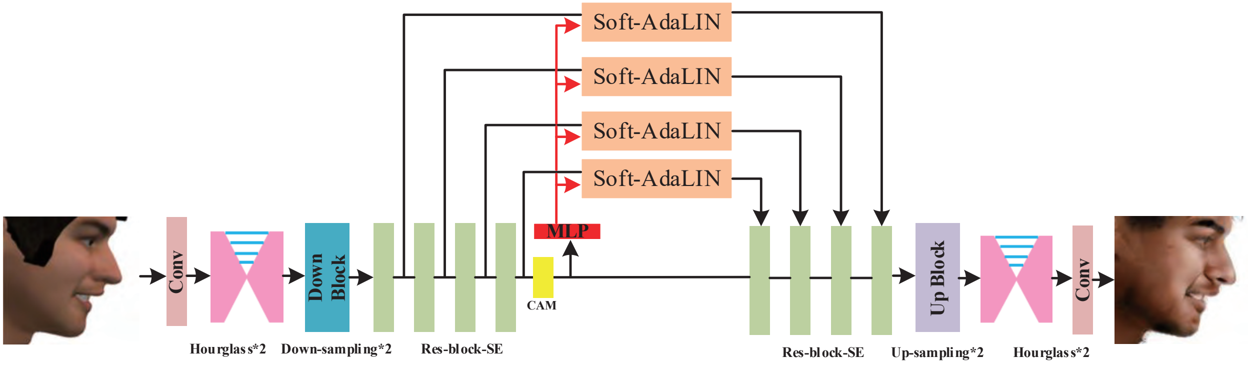


Figure 5.5: The Detail of Generator model.

In contrast to the original Resnet block, the SE module applies to the Residual branch. First, we reduce the feature dimension to 1/r of the input and then raise it back to the original through a Fully Connected layer after ReLu activation. The benefits of having more nonlinearity to better fit complex correlations between channels and reducing the number of parameters and computational effort over a direct FC layer. Through the Sigmoid function to obtain the normalized weights, and finally, the normalized weights are weighted to the features of each channel by a Scale operation. Feature rescaling performs on the branch before the addition. If rescaled after the addition process, the scale operation of 0 and 1 on the main stem, it will be easy to have gradient dissipation near the input layer when the network is into BP optimization, making the model difficult to optimize.

The Attention method of the model is inspired by Class Activation Map (CAM) [203]. The global average pooling in the last layer of the image classification network to obtain the mean value of each feature map as the input of the classification softmax, and the weights of the softmax are trained and used to weight and sum all the feature maps to obtain a visualization result. Fig. 5.6 shows the detail of CAM structure.

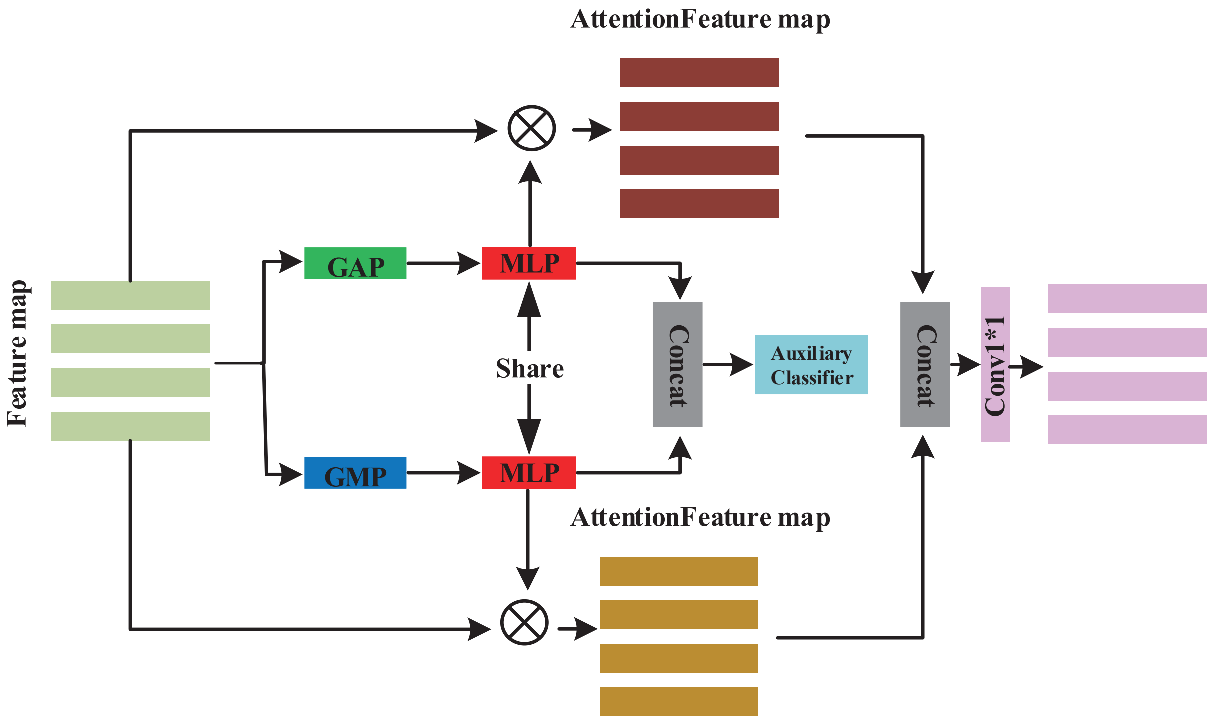


Figure 5.6: The detail of CAM structure.

The Encoder Feature map is obtained by downsampling and residual SE block. Then after global average pooling and global max pooling, a feature vector depending on the number of channels get and learn parameters weight create. Then the learnable parameter weights and Encoder Feature map do multiply method. For each channel of the Encoder Feature map, by assigning a weight that determines the importance of the corresponding feature of that channel to get the attention mechanism under the Feature map. Finally, it concatenates the attention map derived by average and max. And it returns to the number of input channels after one convolutional layer to perform adaptive normalization.

Recent studies on neural style transfer have indicated that CNN feature statistics (mean and variance) can be used as direct descriptors of image styles. In particular, Instance Normalization (IN) has the effect of eliminating style variation by directly normalizing the image feature statistics and is more commonly used in style transfer than Batch Normalization (BN) or Layer Normalization (LN). UGATIT [165] propose an Adaptive Layer-Instance Normalization (AdaLIN) function to choose IN and LN adaptively. With AdaLIN, the model can flexibly control the amount of shape and texture variations. In the open-source Minivision project, they proposed a Soft Adaptive Layer-Instance Normalization (Soft-AdaLIN) method. It incorporates statistics on encoded and decoded features in denormalization.

In Fig. 5.7, we show the detail for Soft-AdaLIN Normalization Method the Soft-AdaLIN normalization. It is a combination of Instance Normalization (IN) and Layer Normalization (LN) that can perform style transfer under the premise of saving image content.

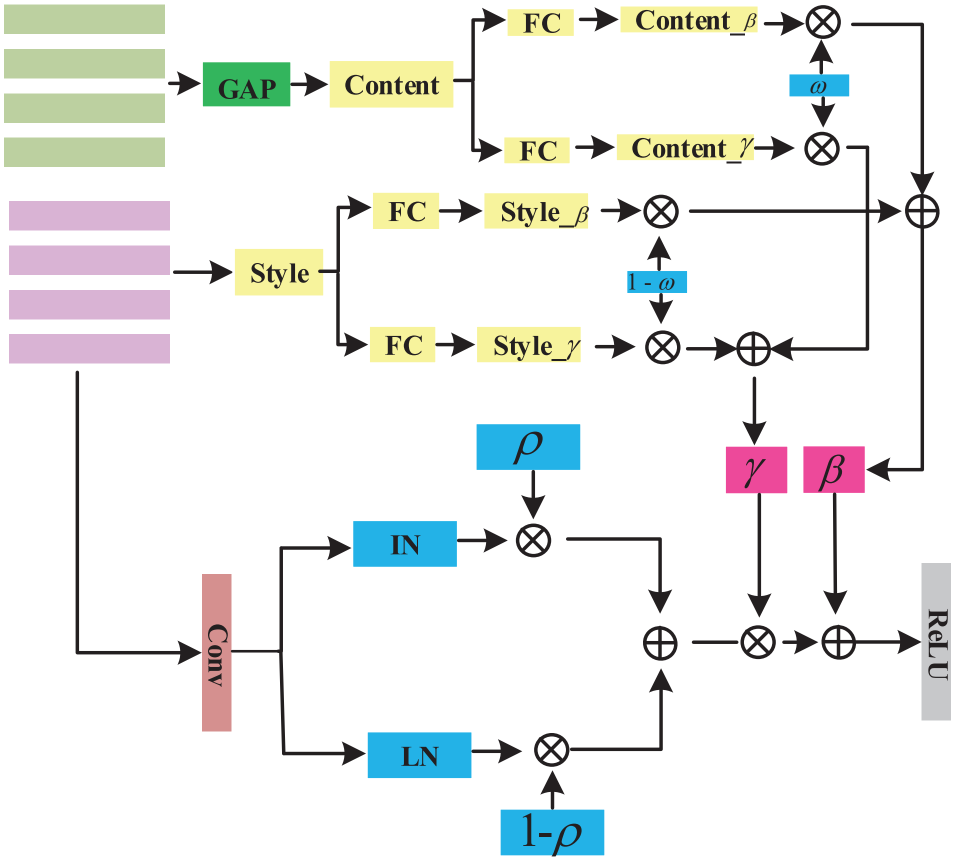


Figure 5.7: The detail of Soft-AdaLIN Normalization method.

As shown in Fig. 5.7, the content feature generated by the encoder feature map and the weights ω of the content feature and style feature are used to get the soft γ and soft β for the Soft-AdaLIN. It helps the attention-directed models to control the amount of variation in shape and texture flexibly, especially for faces in challenging conditions. The next section 5.2.4 describes different normalization methods in detail.

The discriminator is designed by combining a global discriminator and a local discriminator. The difference between the global discriminator and the local discriminator is that the global discriminator compresses the input image at a deeper level, and the upper layer of the final output, the feature map size reaches $h/32, w/32$. The local discriminator, the size of the final output feature map reaches $h/8, w/8$. The specific sizes of the generative and discriminative networks are shown in Table A.1, Table A.2 and Table A.3 respectively in the Appendix A.2.

5.2.3 Loss functions

Our network also includes two generators mapping G_x, G_y and two adversarial discriminators D_x, D_y which are the same as CycleGAN. We only show the loss function of one translation model $X \rightarrow Y$. The same loss function is used for another translation model $Y \rightarrow X$. The full objective of one translation model $X \rightarrow Y$ model comprises six loss functions. The details of each of the loss functions are as follows:

Adversarial loss: Least Squares Generative Adversarial Networks (LSGANs) [196] which adopt the least squares loss function for the discriminator. This loss is employed to match the distribution of the translated images to the target image distribution [165].

$$L_{lsgan} = E_{x\sim\ P_x}[(D_x(x))^2] + E_{y\sim\ P_y}[(1 - D_x(G_x(x)))^2] \tag{5.1}$$

CycleGAN loss: The cycle consistency constraint to the generator, which is the same as Eq. 4.3. This loss makes sure the image translates back to the original domain successfully.

$$L_{cycle} = E_{x\sim\ P_x}[||x - G_y(G_x(x))||_1] \tag{5.2}$$

Identity loss: Same as in the original CycleGAN paper, this loss is to ensure that the color distributions of the input image and output image are similar.

$$L_{identity} = E_{x\sim\ P_x}[||x - (G_y(x))||_1] \tag{5.3}$$

CAM loss: CAM loss is exploiting the information from the generator auxiliary classifiers η_x and discriminator auxiliary classifiers η_{D_x}. This loss can better focus on the region where the source domain differs from the target.

$$\begin{aligned} L_{CAM_G} &= E_{x\sim\ P_x}[log(\eta_x(x))] + E_{y\sim\ P_y}[log(1 - \eta_x(x))] \\ L_{CAM_D} &= E_{y\sim\ P_y}[(\eta_{D_x}(x))^2] + E_{x\sim\ P_x}[(1 - \eta_{D_x}(G_x(x)))^2] \end{aligned} \tag{5.4}$$

FaceId loss: FaceId loss uses the pre-trained face recognition model-MobileFaceNets (M) [204] to extract the id features of the input and the generated output. This loss uses the cosine distance to constrain the id information of the generated output to be as similar as possible to the input.

$$L_{faceid} = E_{x\sim\ P_x}[||1 - cos(M(x), M(G_x(x)))||_1] \tag{5.5}$$

Pixel loss: Pixel loss uses to maintain the characteristics of the lower level after style conversion. (C, H, W) means the shape of the image.

$$L_{pixel} = E_{x\sim\ P_x}[||x - (G_x(x))||_1] \tag{5.6}$$

The total loss function of one translation model $X \to Y$, and the same as another $Y \to X$.

$$L_{X\to Y} = L_{lsgan} + \lambda_1 L_{cycle} + \lambda_2 L_{Identity} + \lambda_3 L_{CAM} + L_{faceid} + L_{pixel} \tag{5.7}$$

Finally, the full objective function of our network is defined as L_{total} (the best results set the parameter $\lambda_1 = 10$, $\lambda_2 = 10$, $\lambda_3 = 1000$).

$$L_{total} = L_{X\to Y} + L_{Y\to X} \tag{5.8}$$

5.2.4 Normalization methods overview

Normalization enables us to use much higher learning rates and be less careful about initialization, and in some cases, eliminates the need for Dropout [205]. Here briefly introduce several normalization methods.

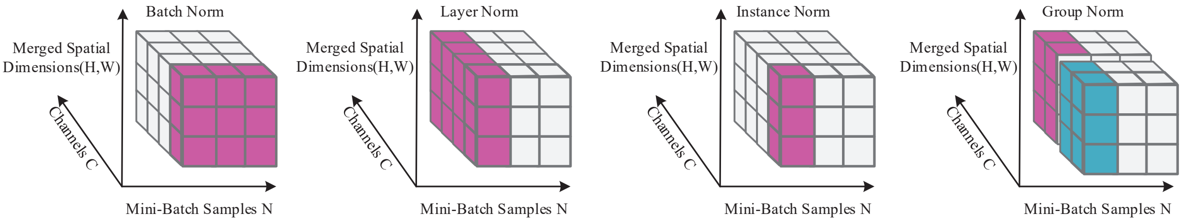


Figure 5.8: Difference Normalization methods such as BN, LN, IN, and GN.

Batch Normalization (BN): Ioffe and Szegedy [205] introduced BN layer to ease the training of feed-forward networks by normalizing feature statistics. BN layers are originally designed to speed up the discriminative networks but have also been found effective at generative modeling of images [206].

Given an input batch $x \in \mathbb{R}^{(N \times C \times H \times M)}$, BN normalizes the mean and standard deviation for each feature channel.

$$BN(x) = \gamma(\frac{x - \mu(x)}{\sigma(x)}) + \beta \tag{5.9}$$

where $\gamma, \beta \in \mathbb{R}^C$ are affine parameters learned from data. $\mu(x), \sigma(x) \in \mathbb{R}^C$ are the mean and standard deviation computed across batch size and spatial dimensions independently for each feature channel.

$$\begin{aligned} \mu_c(x) &= \frac{1}{NHW}\sum_{n=1}^{N}\sum_{h=1}^{H}\sum_{w=1}^{W}(x_{nchw}) \\ \sigma_c(x) &= \sqrt{\frac{1}{NHW}\sum_{n=1}^{N}\sum_{h=1}^{H}\sum_{w=1}^{W}(x_{nchw} - \mu_c(x))^2 + \epsilon} \end{aligned} \tag{5.10}$$

Layer Normalization (LN): LN considers more correlations between input feature channels [207], LN is more thorough than IN in style transformation, but semantic information is not preserved enough. IN takes more into account the content of individual feature channels, and IN preserves the semantic information of the original image better than LN, but style transformation is not complete.

$$\begin{aligned} \mu_n(x) &= \frac{1}{CHW}\sum_{c=1}^{C}\sum_{h=1}^{H}\sum_{w=1}^{W}(x_{nchw}) \\ \sigma_n(x) &= \sqrt{\frac{1}{CHW}\sum_{c=1}^{C}\sum_{h=1}^{H}\sum_{w=1}^{W}(x_{nchw} - \mu_n(x))^2 + \epsilon} \end{aligned} \tag{5.11}$$

Instance Normalization (IN): Ulyanov et al. [208] found that significant improvement can be achieved simply by replacing BN layers with IN layers: different from BN layers, here $\mu_{nc}(x), \sigma_{nc}(x)$

are computed across spatial dimensions independently for each channel and each sample:

$$\begin{aligned} \mu_{nc}(x) &= \frac{1}{HW}\sum_{h=1}^{H}\sum_{w=1}^{W}(x_{nchw}) \\ \sigma_{nc}(x) &= \sqrt{\frac{1}{HW}\sum_{h=1}^{H}\sum_{w=1}^{W}(x_{nchw}-\mu_{nc}(x))^2+\epsilon} \end{aligned} \tag{5.12}$$

Group Normalization (GN): It is suitable for tasks that consume a large amount of video memory, such as image segmentation. GN is also batch-independent and is a compromise between LN and IN. They compute GN by dividing each sample feature map channel into G groups, each group will have C/G channels, and then the elements in these channels are normalized to the mean and standard deviation. Each group of channels is normalized independently with its corresponding normalization parameter.

$$\begin{aligned} \mu_{ng}(x) &= \frac{1}{(C/G)HW}\sum_{c=gC/G}^{(g+1)C/G}\sum_{h=1}^{H}\sum_{w=1}^{W}(x_{nchw}) \\ \sigma_{ng}(x) &= \sqrt{\frac{1}{(C/G)HW}\sum_{c=gC/G}^{(g+1)C/G}\sum_{h=1}^{H}\sum_{w=1}^{W}(x_{nchw}-\mu_{nc}(x))^2+\epsilon} \end{aligned} \tag{5.13}$$

Recent research shows that a weighted combination of such basic normalization methods can be adaptively normalized. For example, some typical improvement methods propose Adaptive LayerInstance Normalization and Soft Adaptive LayerInstance Normalization, which combines IN and LN normalization methods.

Table 5.1 lists the common normalization method's advantages and disadvantages. For example, each texture can be manipulated to change the style of the image by using IN. LN normalizes all channels and reassigns the mean-variance to them. So it is utilized to change the combination of shape features, but not the texture features.

Adaptive LayerInstance Normalization (AdaLIN): AdaLIN whose parameters are learned from datasets during training time by adaptively selecting a proper ratio between IN and LN [165].

$$\begin{aligned} AdaLIN(x) &= \gamma(\rho\cdot IN(x_{nchw})+(1-\rho)\cdot LN(x_{nchw}))+\beta \\ \rho &\leftarrow clip_{[0,1]}(\rho-\tau\Delta\rho) \end{aligned} \tag{5.14}$$

Table 5.1: Examples of several normalization methods.

Method	Advantage and Disadvantage
BatchNorm	The batch direction to do normalization, calculate the mean of NHW, does not work well for small batch size; the main disadvantage of BN is that it is sensitive to the size of the batch size because each calculation of the mean and variance is on a batch, so if the batch size is too small, the calculated mean and variance is not enough to represent the whole data distribution.
LayerNorm	The channel direction is normalized to calculate the mean value of CHW, which mainly use for RNN.
InstanceNorm	Normalization is done within a channel to calculate the mean value of H*W, which uses in stylized migration; since in image stylization, the generated result depends mainly on one image instance, normalization of the whole batch is not suitable for image stylization, so the normalization of HW is done. It can accelerate the model convergence and keep the independence between each image instance.
GroupNorm	Divide the channel directions into groups, then do normalization within each group and calculate the mean value of (C//G) HW; this is independent of the batch size and not constrained by it.
SwitchableNorm	It consists of BatchNorm, LayerNorm, InstanceNorm, giving weights and letting the network learn the normalization layer on its own.

Here, γ, β are parameters generated by the fully connected layer, τ is the learning rate, and $\Delta\rho$ indicates the parameter update vector (e.g., the gradient) determined by the optimizer. The values of ρ are constrained to the range of [0, 1] simply by imposing bounds at the parameter update step.

Soft Adaptive LayerInstance Normalization (Soft-AdaLIN): Unlike the original AdaLIN, the ”soft” feature is not to use the cartoon feature statistics directly to denormalize the decoded features, but to weight the average cartoon and encoded feature statistics by the learnable weights w_μ, w_σ, and to denormalize the normalized decoded features. Encoding feature statistics μ_{en}, σ_{en} is obtained by residual SE block. The output feature statistics μ_c, σ_c of each Resblock-SE in the feature extraction section are extracted from the feature maps output by the CAM module through the fully connected layer. The weighted statistics are expressed as:

$$\begin{aligned} Soft - AdaLIN(x) = \gamma(\rho \cdot IN(x_{nchw}) + (1-\rho) \cdot LN(x_{nchw})) + \beta \\ \gamma = w_\sigma \cdot \sigma_{en} + (1 - w_\sigma) \cdot \sigma_c \\ \beta = w_\mu \cdot \mu_{en} + (1 - w_\mu) \cdot \mu_c \end{aligned} \tag{5.15}$$

5.3 Experiments

5.3.1 Experiments results

In this experiment, 2500 synthetic images are generated from the FaceGen model as trainS, and 2500 real images from 300W [6] front face, and Menpo [1] side face datasets as trainR. Weights are initialized from a zero-centered normal distribution with a standard deviation of 0.02. For data enhancement, the images are flipped with a probability level of 0.2. All models were trained with a fixed learning rate of 0.0001 until 500000 iterations and then trained for 5000000 iterations using linear decay learning rate.

More experimental results of same person through the translation model $S \rightarrow R$ in Fig. 5.9 and Fig. 5.10 which include front and side face dataset. The input images are synthetic images generated from the FaceGen 3D model, and the output images are realistic images through our $S \rightarrow R$ model.

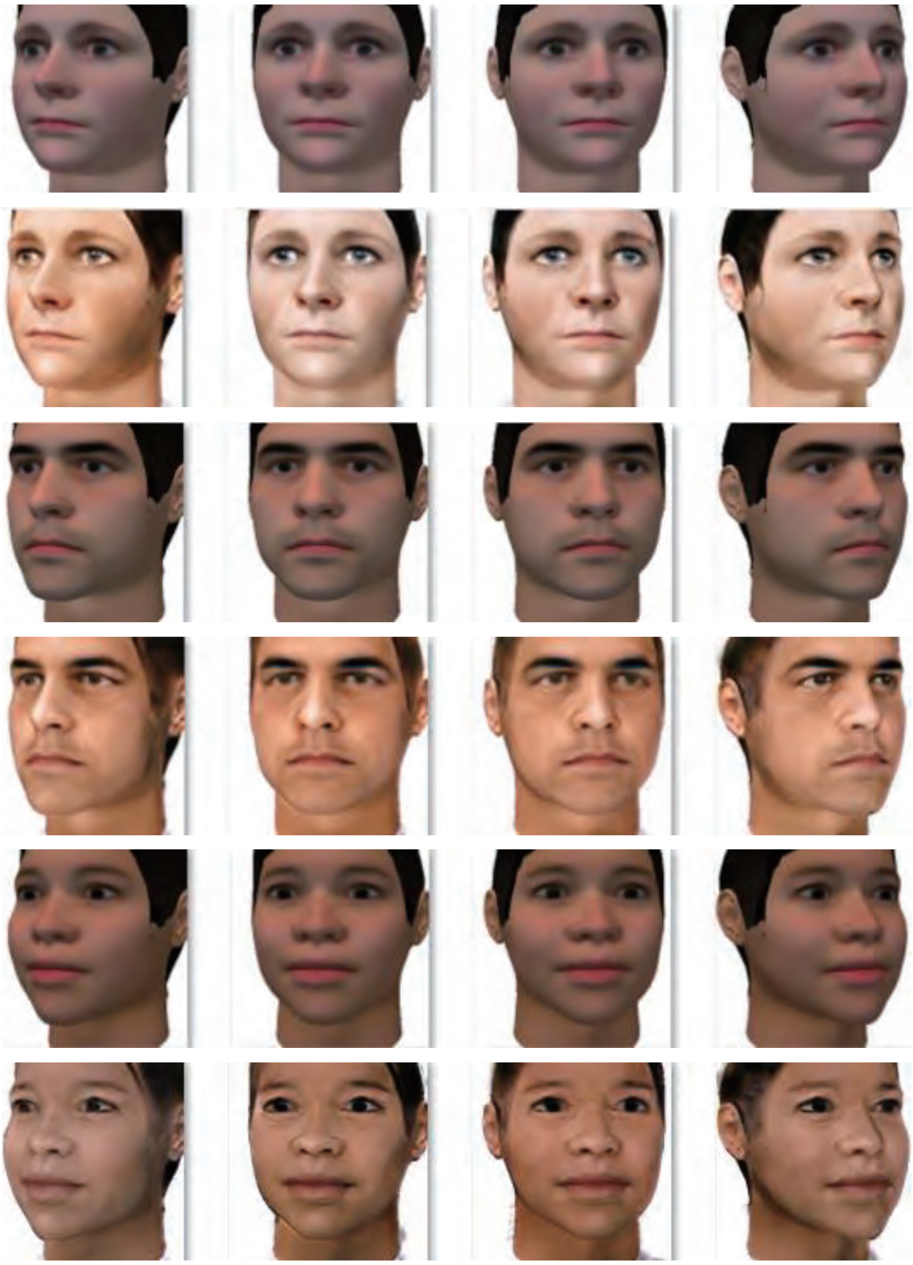


Figure 5.9: The front face results through translation model $S \rightarrow R$, odd-numbered rows represent input, even-numbered rows represent the output.

In the real-life, faces have various head poses, including extreme profile views. Fig. 5.10 also uses $S \rightarrow R$ model to test the large-pose data.

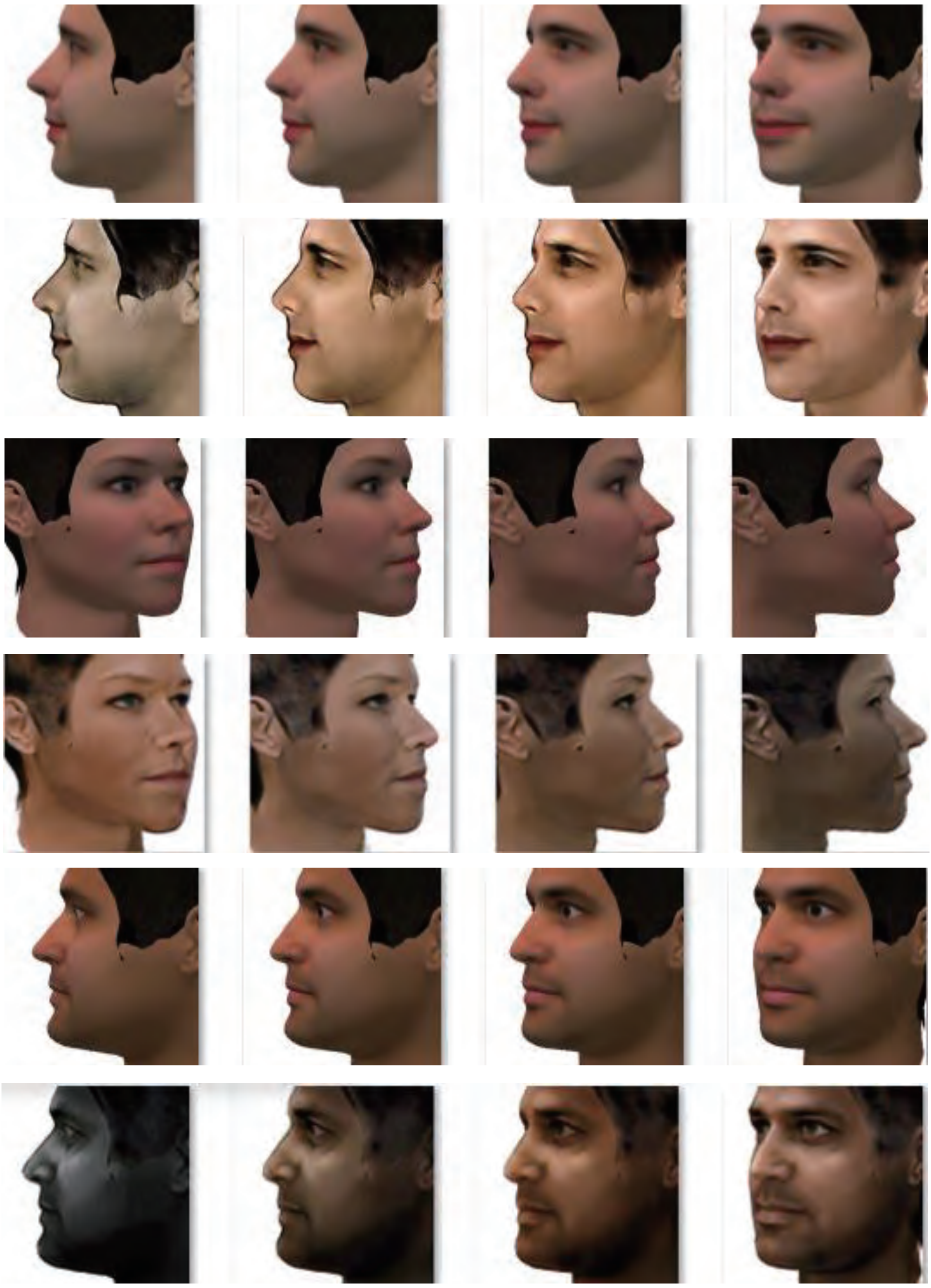


Figure 5.10: The side face results through translation model $S \rightarrow R$, odd-numbered rows represent input, even-numbered rows represent output.

It can easily convert the synthetic face images generated from the FaceGen 3D model into more realistic face images by the proposed translation model. The advantage is that the generated face image has a high perceptual quality and can be used to improve the performance of the face. Compared to the previous synthetic data, the images transformed by the $S \rightarrow R$ model maintain the content of the synthetic datasets while incorporating the style of the real-world images.

5.3.2 Experiments evaluation

This method is evaluated on the following challenging datasets: COFW, Menpo and WFLW. The augmentation Real 300W plus synthetic face images with realistic appearances (RA+G), which are transferred by the translation model $S \rightarrow R$. For comparison, figures visualize the detection results trained on various datasets (RA, RA+S, and RA+G) separately.

In Fig. 5.11 and Fig. 5.12, the areas that show the superiority of the proposed method in detecting key points are highlighted with red boxes and arrows.

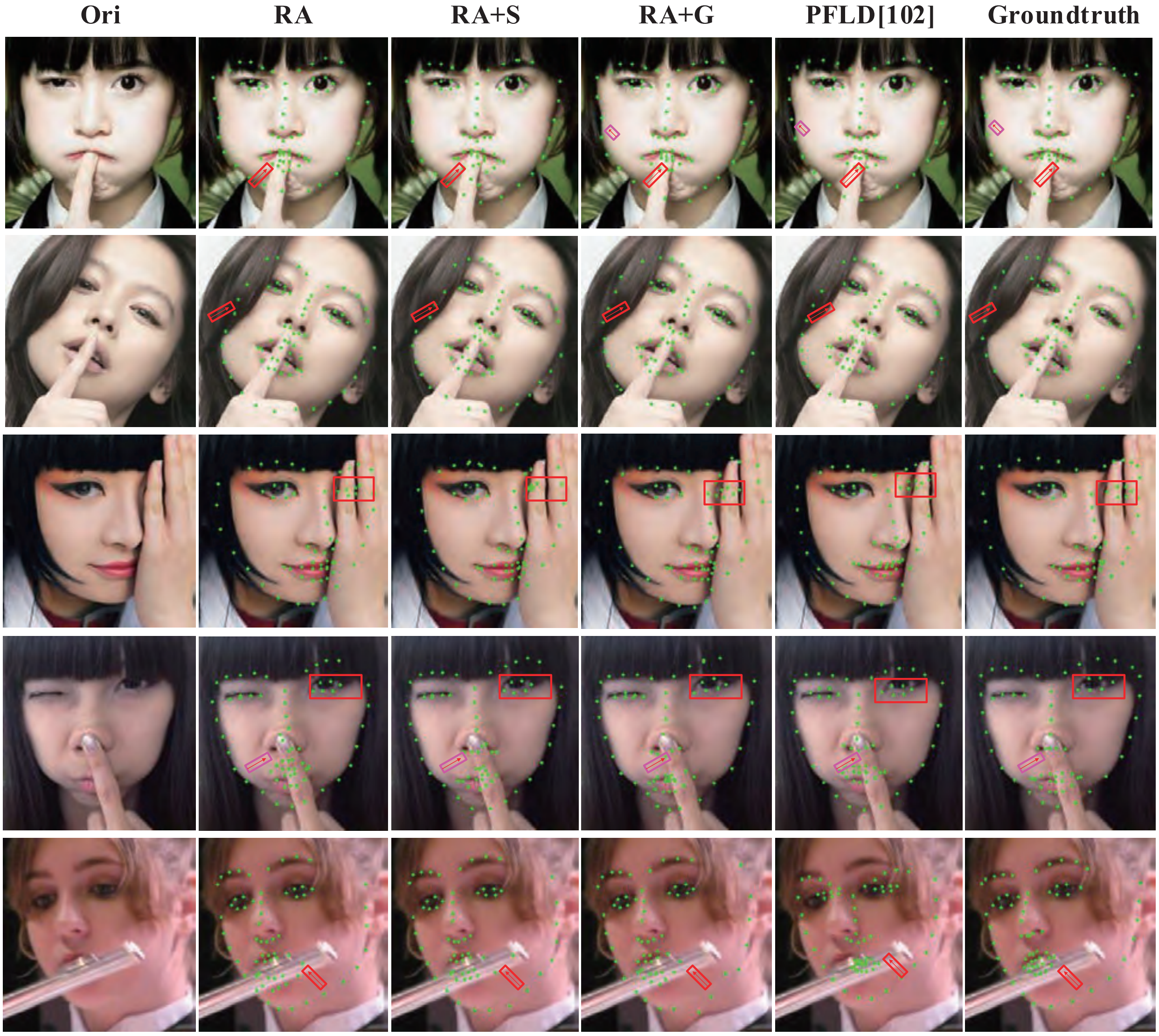


Figure 5.11: Face alignment results in COFW occlusion datasets.

Fig. 5.11 shows results on COFW datasets. It can be seen the proposed RA+G method can predict landmark localization well in occluded regions. It further illustrates that the accuracy of model detection can improve by synthetic datasets transferred by a GAN-based approach.

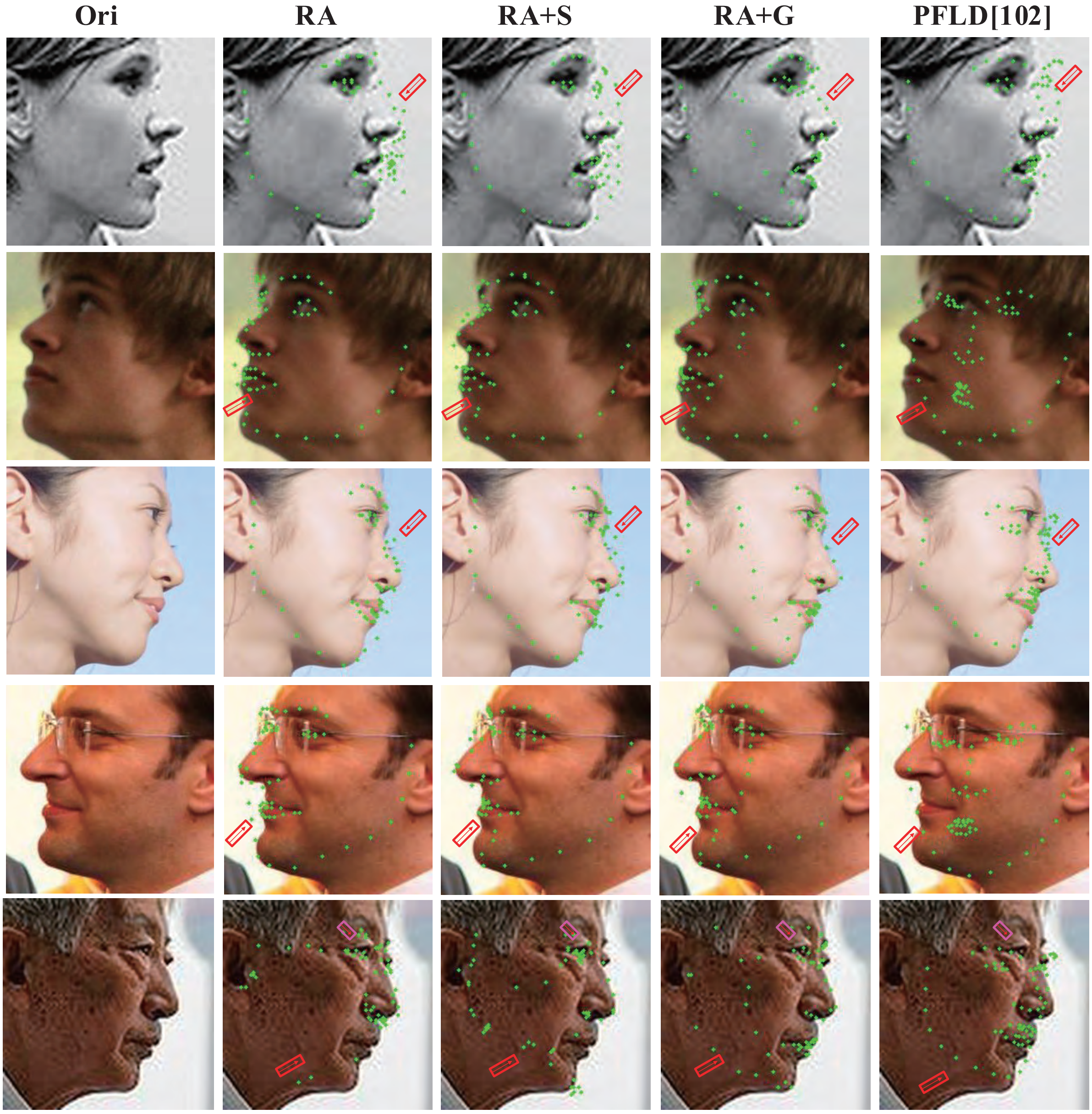


Figure 5.12: Face alignment results in Menpo-side datasets.

Fig. 5.12 also shows some examples on Menpo-side datasets, if the model uses only the real frontal face data (RA) for training, the detection rate is very low, and the contours of the face are barely detected when testing on faces with large-scale poses. Detection accuracy is further improved by adding synthetic datasets (RA+S) to the face alignment models, but the prediction for invisible points is still poor. When applying the synthetic dataset (RA+G) after GAN translation model $S \rightarrow R$, the invisible coordinates detection is improved.

WFLW dataset [82] is regarded as the most challenging dataset based on WIDER FACE. It contains 7500 faces with 98 annotated landmarks and corresponding face bounding boxes for training and 2500 for testing. Faces in WFLW are collected under unconstrained conditions, which involve rich attributes such as several subsets of large variations in poses (326 images), expression (314 images), illumination (698 images), make-up (206 images), occlusion (736 images), and blur (773 images).

WFLW dataset allows the analysis of results stratified by different types of difficulties (e.g., facial expressions, large poses, illumination changes). To comprehensively evaluate the robustness of the proposed method, NME on the test set and six typical subsets from WFLW are reported. The definitions of the six test set subsets come from the official attribute annotations.

RA means using 300W front face data as training datasets, RA+G means using 300W and the transformed synthetic datasets as training datasets, WFLW means WFLW as training datasets, and similarly, WFLW+G means adding WFLW to G as training datasets.

Table 5.2: NME Error of face alignment methods on the WFLW test dataset.

Metric	Method	Test	Pose	Exp.	Illu.	Mak.	Occ.	Blur
NME(%)	ESR [23]	11.13	25.88	11.47	10.49	11.05	13.75	12.20
	SDM [191]	10.29	24.10	11.45	9.32	9.38	13.03	11.28
	CFSS [209]	9.07	21.36	10.09	8.30	8.74	11.76	9.96
	DVLN [184]	6.08	11.54	6.78	5.73	5.98	7.33	6.88
	LAB [82]	5.27	10.24	5.51	5.23	5.15	6.79	6.32
	Wing [55]	5.11	8.75	5.36	4.93	5.41	6.37	5.81
	RA	19.95	25.75	13.98	14.34	15.64	16.58	18.33
	RA+G	14.04	18.75	9.71	9.78	9.73	12.61	11.77
	WFLW	5.53	8.76	5.97	5.39	5.34	6.35	6.09
	WFLW+G	**4.87**	**7.96**	**5.22**	**4.74**	**4.74**	**5.89**	**5.49**

In Table 5.2, the proposed method achieves better results on the WFLW test datasets that are more difficult than COFW and 300W (see Fig. 5.13 and Fig. 5.14 for visualizations). On each subset, WFLW+G models outperform previous approaches. Overall, the results show WFLW+G achieves the minimum NME values for challenging test datasets, and error is reduced by 0.66% compared to WFLW in the test images, which is an improvement compared to previous results and indicates the proposed approach is more robust to challenging scenarios. Compared to RA+G and RA, training datasets are mostly front faces and have fewer challenging datasets. So it has little effect on the test set of WFLW. However, compared to WFLW, the synthetic dataset transformed by the proposed algorithm has improved on each subset, especially for large pose subsets.

The proposed algorithm for six different subsets are visualized in Fig. 5.13 and Fig. 5.14. Fig. 5.13 shows faces alignment results under uncontrollable conditions such as blurred faces, different expressions and lighting.

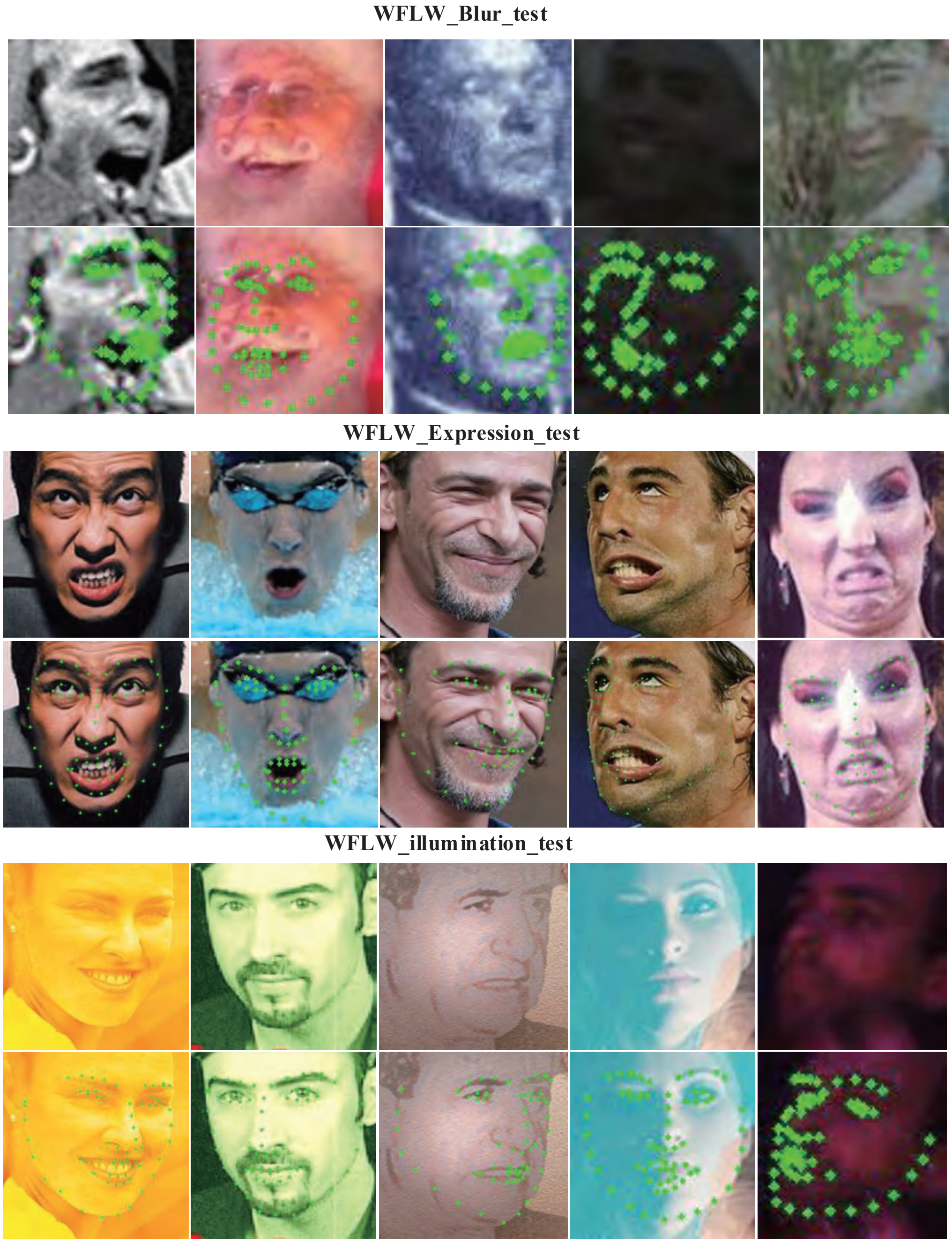


Figure 5.13: Face alignment results in WFLW (Blur, Expression, and Illumination)subsets.

Fig. 5.14 shows faces alignment results under different poses, occlusions, and make-up styles. (Note: the test image set includes images of different sizes and resolutions, and we zoomed the small images and unified them to the same size in the visualization. So the faces predicted points (green) in the figure are thick and thin).

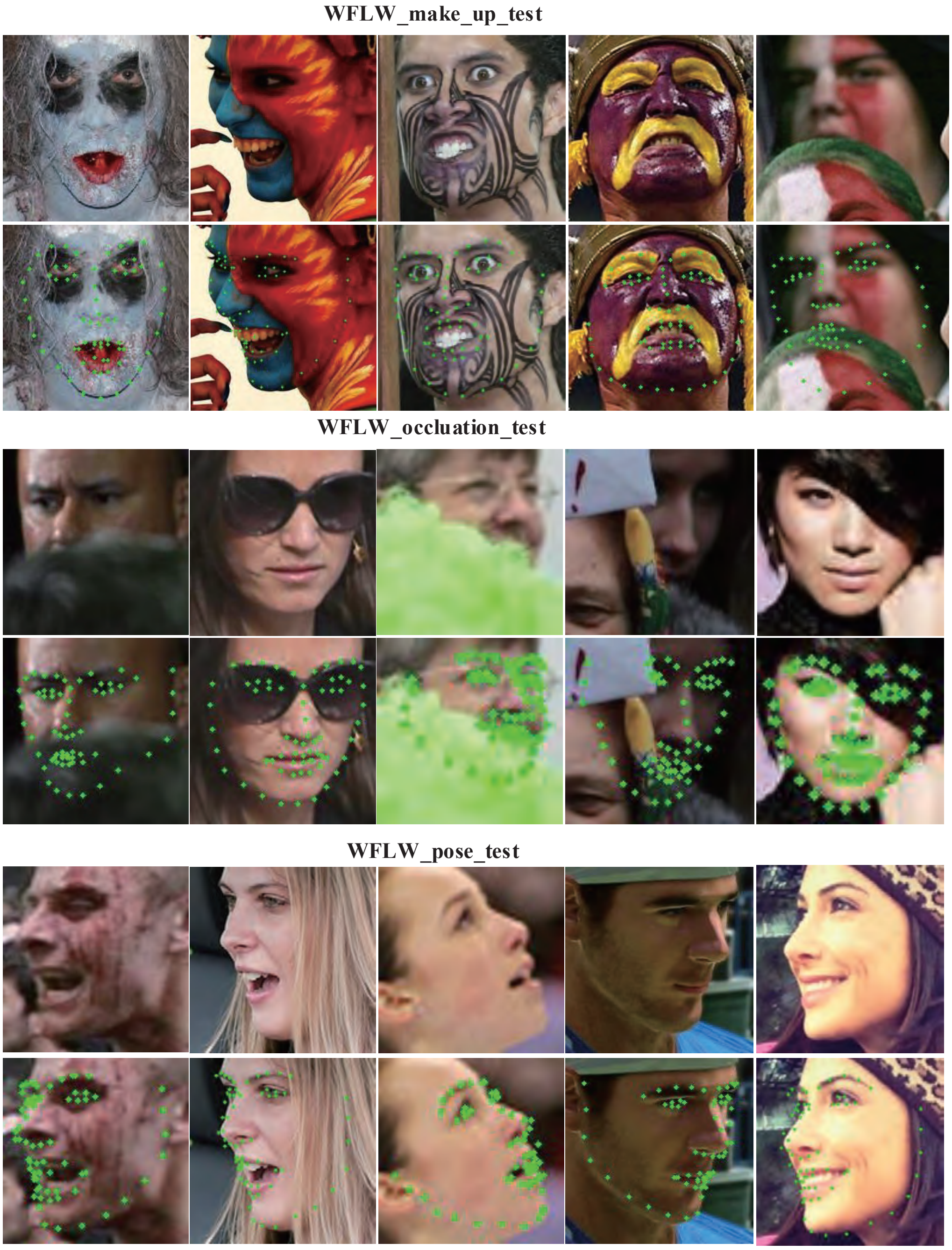


Figure 5.14: Face alignment results in WFLW (Make-up, Occlusion, and Pose) subsets.

It can be visually seen the advantages of the proposed model from the figures. The results indicate that the proposed model is robust to extreme conditions. It also shows that synthetic face images complement biased real-world datasets, thus improving the performance across face alignment benchmarks.

5.4 Summary

The models that transform from real-world to synthetic data cannot handle challenging datasets, for example, large-scale expression data. So another question is whether GANs models can be used in a data augmentation approach to introduce novel variations in the data. In this chapter, an improved model for transforming synthetic to real datasets is proposed. Firstly it needs to remove complex background from the real-world image set through pre-processing methods. Secondly, by adding an attention mechanism based on the UGATIT model from Minivision's project to improve the pre-trained UGATIT network . Finally, the transformed datasets as an augmentation of real datasets feed into the face alignment algorithm. The face alignment results show that synthetic data combined with real-world data can achieve large performance gain with small realworld data.

Chapter 6

Synthetic data applied to lightweight networks for face reconstruction

FaceGen model is limited in simulating synthetic data. Fig. 6.1 shows that even when the parameter controlling facial expressions are maximized, there is still a big difference compared to the real large facial expressions.

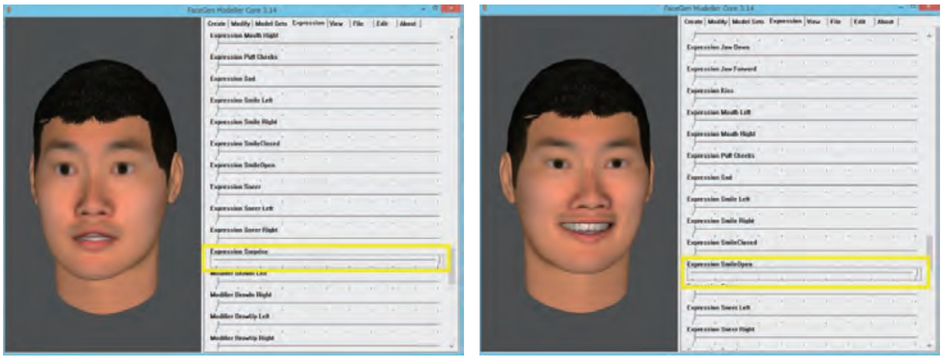
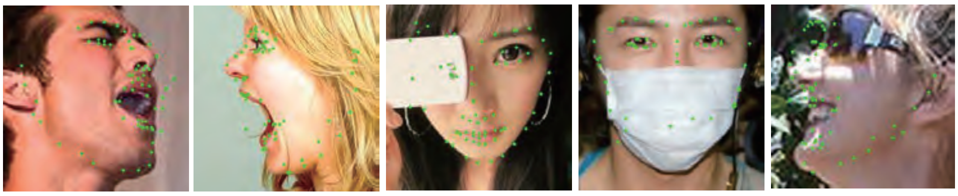


Figure 6.1: Limiting factors for FaceGen's expression parameters (e.g., surprise and smile).

Fig. 6.2 shows some examples performed poorly, especially for multiple challenging conditions.



Large-scale expression

Large-scale occlusion

Figure 6.2: Some examples performed poorly with the above proposed face alignment network.

6.1 Dataset

Previous methods faced a challenge that the training samples unbalanced distribution, especially for faces with multiple challenging conditions. The synthetic dataset 300W-LP [10] is applied, which are constructed by fitting the 3DMM with the Multi-Features Framework [210].

The generated process show in Fig. 6.3, which can produces multiple rendered views by 3D Face rendering.

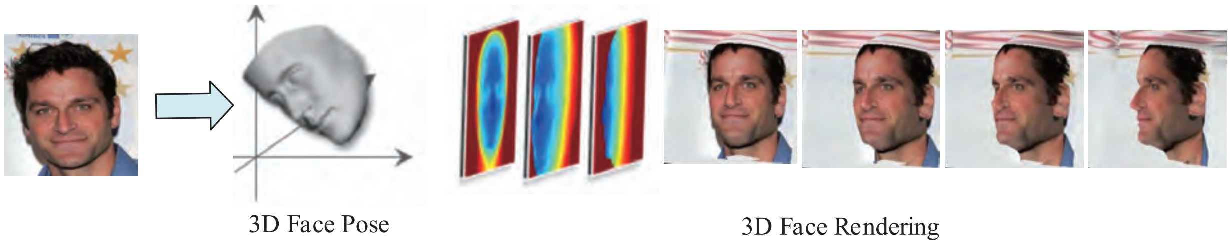


Figure 6.3: The generated datasets process by 3D Faces Rendering [9, 10, 11, 12].

Fig. 6.4 shows generated examples, which rely on the 3DMM to do the rotations and flipping.

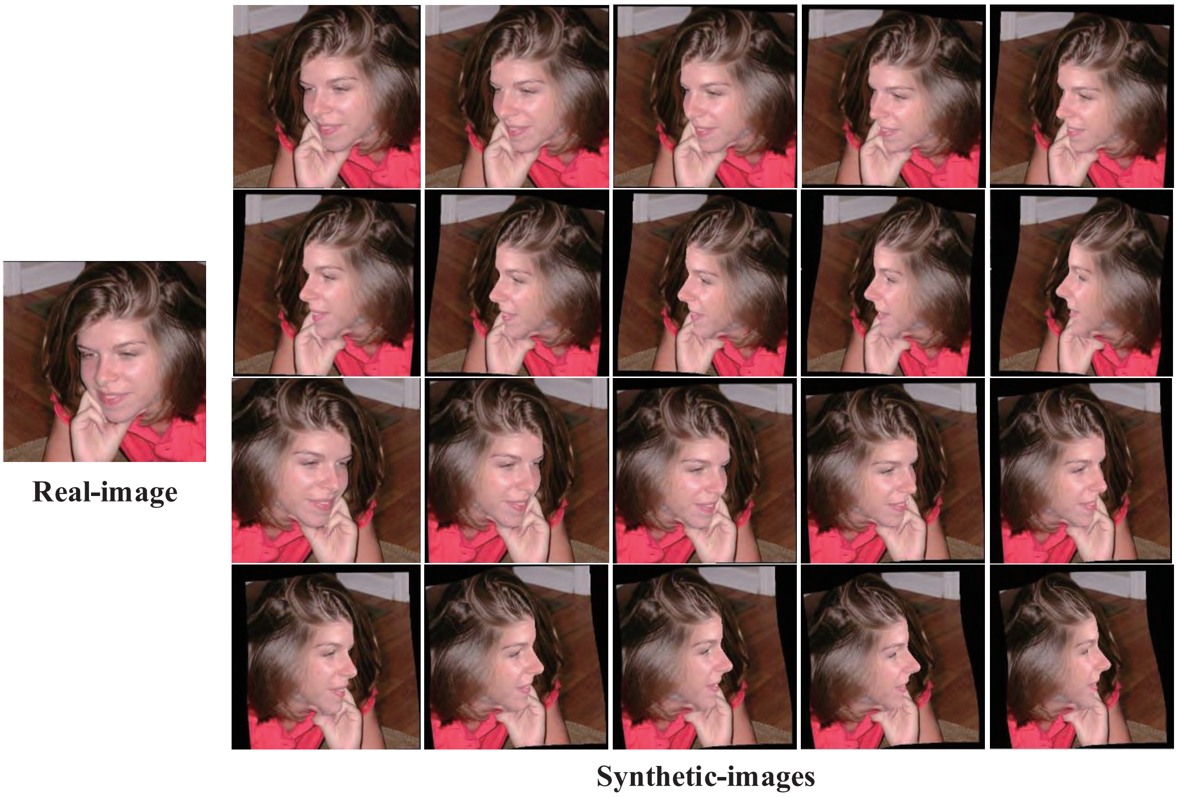


Figure 6.4: Real-world images come from 300W, and synthetic images generate with the generative algorithm described above.

Different from the original algorithm, the 3D landmarks are adjusted by landmark marching [151] and the 68-landmarks constraint is adopted throughout the fitting process[13]. Fig. 6.5 show the real-world and synthetic images and annotation of 300W-LP.

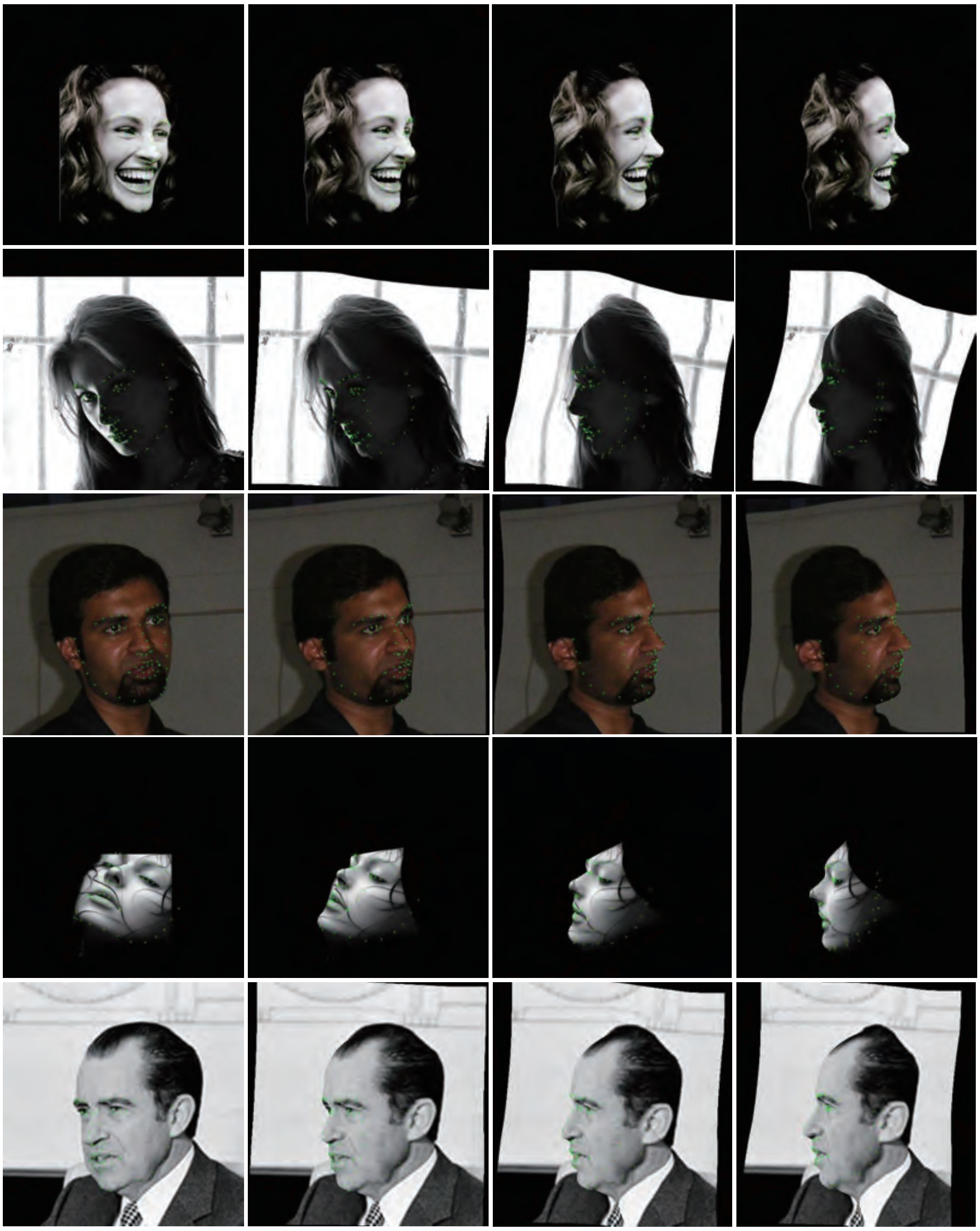


Figure 6.5: Examples of the 300W-LP, the first is original images, followed by the synthesized images with annotation.

300W-LP: The 300W across Large Poses (300W-LP) dataset [10] contains 61,225 synthetic face images across large poses (1,786 from IBUG [6], 5,207 from AFW [3], 16.556 from LFPW [5], and 37,676 from HELEN [52]) along with their corresponding 3DMM annotation coefficient values. These images are synthesized from 300W [6] through a morphable model-based 3D profiling algorithm proposed in [10] and are of coverage across large pose ranges from -90 to 90 degrees.

The dataset contains the images and their ground truth 3D faces. Fig. 6.6 gives some examples from 300W-LP and AFLW2000-3D datasets.

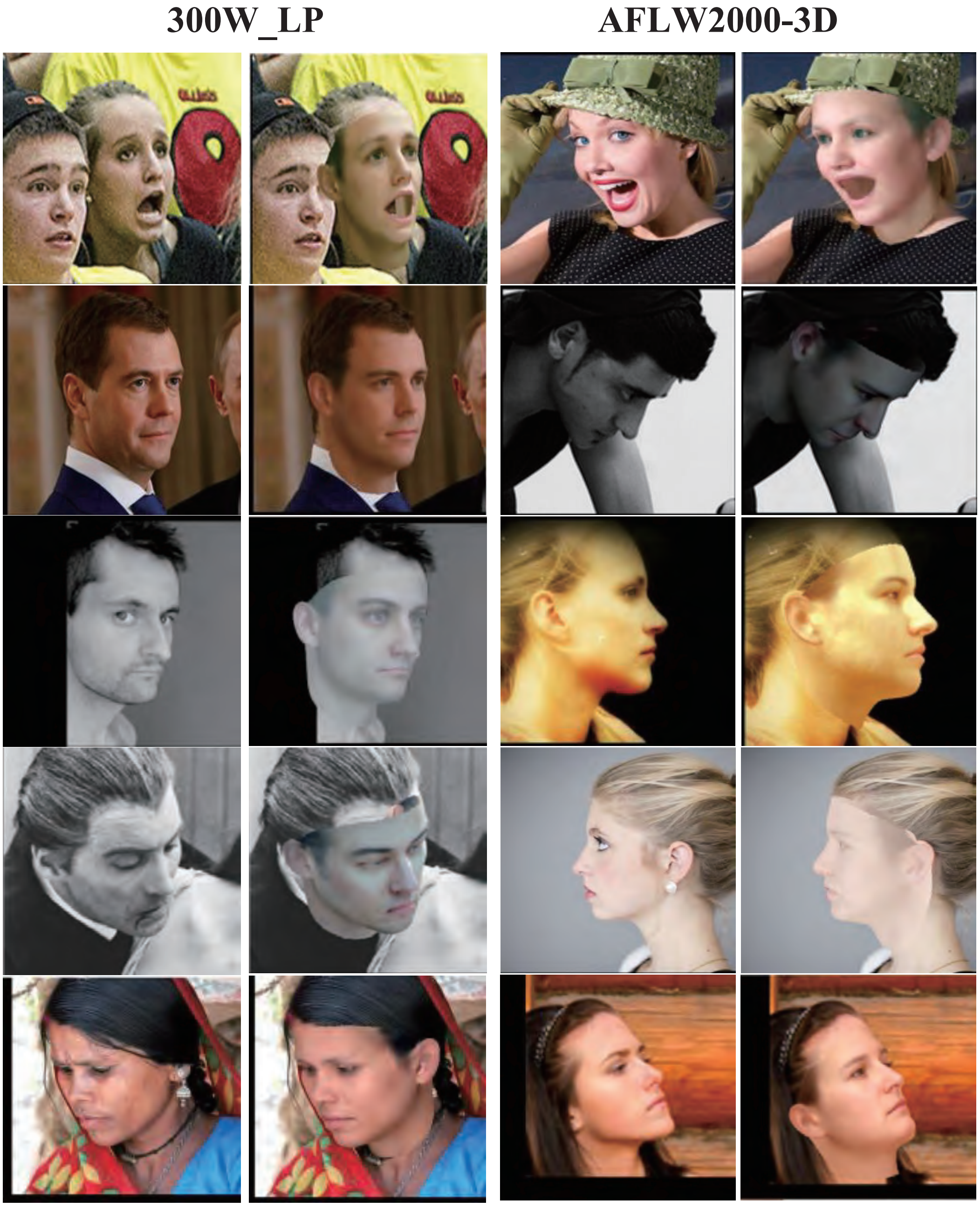


Figure 6.6: Examples of 300W-LP and AFLW2000-3D datasets [13], the left is original images, the right is the fitted 3DMM.

AFLW2000-3D: The AFLW2000-3D dataset is a sample of 2,000 faces selected from the AFLW dataset [2]. Zhu et al. [10] introduced this dataset and annotated its corresponding 3DMM coefficients and the corresponding 68 3D facial landmarks. For evaluation, the test datasets are split into 3 subsets according to their absolute yaw angles: $[0°, 30°]$, $[30°, 60°]$ and $[60°, 90°]$ with 1,312, 383 and 365 samples, respectively.

AFLW: The AFLW dataset [2] contains 21,080 faces in the wild, which is a large-scale face database including multi-poses and multi-views, and each face is annotated with 21 feature points. At test time, the test datasets are split into 3 subsets based on absolute yaw angle: $[0°, 30°]$, $[30°, 60°]$ and $[60°, 90°]$ with 11,596, 5457, and 4027 samples, respectively.

Menpo-3D: The Menpo-3D [1] dataset contains 8,955 challenging frames varying in illuminations, poses, and occlusions.

6.2 Method

6.2.1 Network architecture

The goal is to regress the face's three-dimensional geometric shape and its corresponding landmark information simultaneously. Briefly, given a 2D image, the algorithm can locate its key points and provide 3D face structure and depth information. It bases on lightweight convolutional neural networks. Moreover, the proposed network incorporates a channel-wise attention mechanism that can improve the ability of the network. The proposed method could substantially improve the quality of the 3D face reconstruction, even though lacking 3D samples and 3D annotations for the 2D images. It takes both the face image and 3DMM coefficients as inputs and learns the model to evaluate the intrinsic consistency between the predicted 3DMM coefficients and the corresponding face image, offering supervision for 3D face model learning.

A 3D face model (3DMM) is fitted to tackle the face alignment problem in large-scale expression and occlusion rather than heatmap or coordinate regression. By incorporating 3D information [10], it can inherently address variations in appearance and occlusion. The model should be able to recover 2D landmarks from predicted 3D ones via direct 3D-to-2D projection. The structure of most previous 3DMM-based networks is complex, and the model parameter space is massive, making it difficult to achieve convergence in network training.

Fig. 6.7 shows the pipeline of the proposed method, which describes network structure. Model's architecture contains convolutional layers with H-swish activation [211], a stack of reconstructed

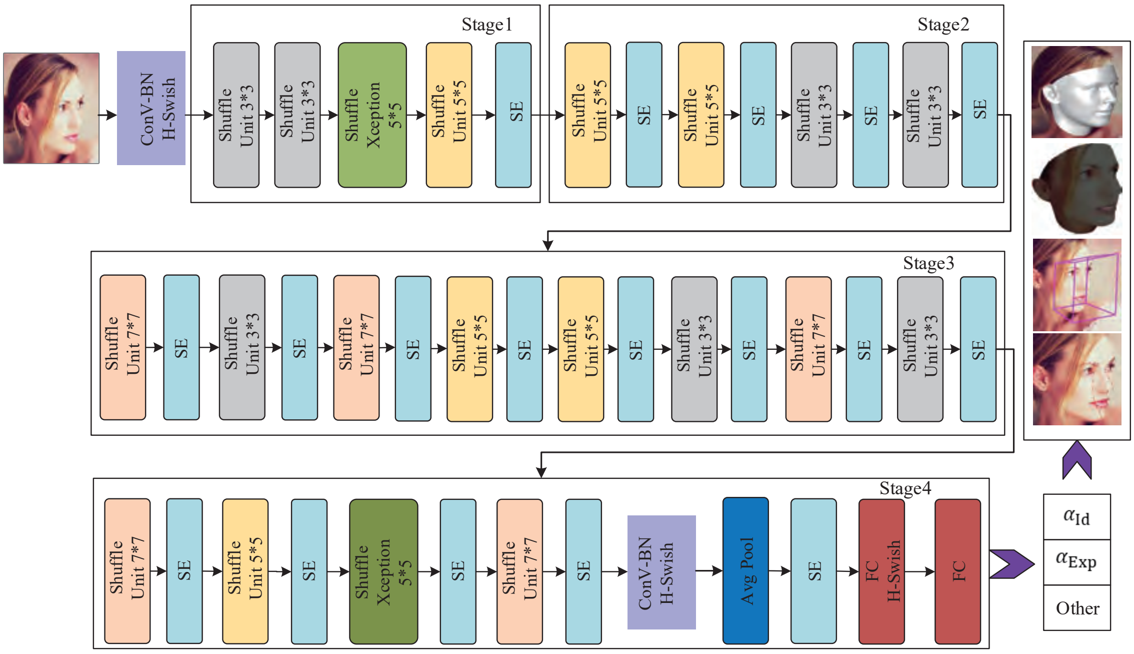
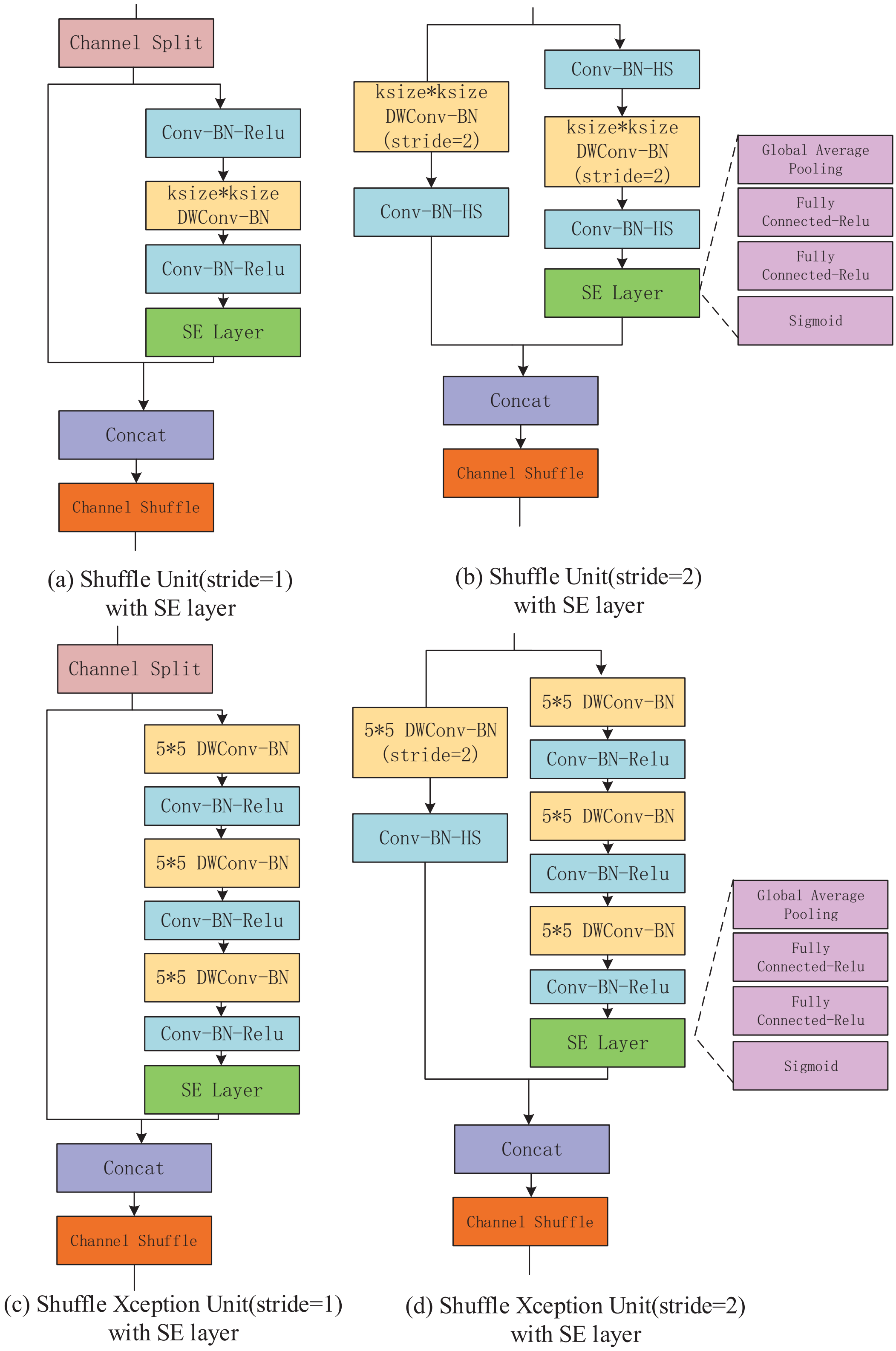


Figure 6.7: Network architecture design for face reconstruction.

Shufflenet block units that are structured in four stages, finally, with FC layers. Blocks in gray are 3×3 ShuffleNet unit, larger green blocks are ShuffleNet Xception unit, which is deeper than 3×3 ShuffleNet unit, yellow and orange are also ShuffleNet unit with the sizes of kernels as 5 and 7, respectively. H-swish[x]=$x\frac{\text{ReLU6(x+3)}}{6}$ was utilized as a drop-in replacement for ReLU. ReLU6 is a modification of ReLU where constrains the activation to a maximum size of 6. Attention mechanisms can automatically learn the importance of each feature channel and enhance useful features and suppress non-informative features by the learned importance metric, which are used widely in many applications [88, 212, 213]. The addition of SE attentional mechanisms can enhance the representation power of the proposed network structure. The computational amount in the ShufflenetV2 network is small on DWConv, and the main computation is on 1×1 convolution. Therefore, extending the convolution kernel of DWConv can improve the effect without increasing the computational weights. Here, the size of the reconstructed the Shuffle Xception unit is 5.

Real-world tasks have motivated much work to design lightweight architectures. The core of lightweight networking is to lighten the network in size and speed while maintaining as much accuracy as possible. Several lightweight network architectures proposed in the last two years mainly include MobileNet [211, 214, 215], ShuffleNet [86, 87], GhostNet [216], Xception [217]. Following the insight from some literature [211, 214, 215, 217], Group-wise Convolution and Depthwise Convolutions (DWConv) are both crucial in these works [87].



(a) Shuffle Unit(stride=1) with SE layer

(b) Shuffle Unit(stride=2) with SE layer

(c) Shuffle Xception Unit(stride=1) with SE layer

(d) Shuffle Xception Unit(stride=2) with SE layer

Figure 6.8: Structure of a Shuffle unit with SE layer and a reconstructed Shuffle Xception unit with SE layer.

An efficient ShuffleNet unit includes DWConv with Batch Normalization layer. Fig. 6.8 shows the operation of the reconstructed ShuffleNet unit in the network structure. SE layer induces the model to pay more attention to the contribution of critical feature areas of the human face and reduces the influence of other unrelated features. First, the SE layer uses a global average pooling as a squeeze operation. Then, two fully connected layers form a bottleneck structure to model the correlation between channels and output the same number of weights as input features numbers. After that, a normalized weight between 0 and 1 obtain through a sigmoid function.

6.2.2 3D Morphable Model

Blanz et al. [34, 85] design a 3D Morphable Model (3DMM) to recover the 3D facial geometry. It is composed of a parameterized generative 3D shape, a parameterized albedo model, together with an associated probability density on the model coefficients [61].

The 3DMM renders a 3D face shape of S_f with a linear combination over a set of Principal Component Analysis (PCA) basis functions. It is one of the most widely used methods for describing the space of 3D faces nowadays. The 3DMM model can be expressed as follows:

$$S_f = \bar{S_f} + \alpha_{id} A_{id} + \alpha_{exp} A_{exp} \tag{6.1}$$

where $\bar{S_f}$ represents the mean shape. The identity basis A_{id} and the expression basis A_{exp} come from the Basel Face Model (BFM) [141] and the Face Warehouse model [142] respectively. Following [10], they use 40 bases from BFM to generate the face shape component and 10 bases from Face Warehouse to generate the face expression component. α_{id} and α_{exp} are the corresponding coefficient of identity and expression.

In the process of 3DMM fitting, the Weak Perspective Projection to project 3DMM onto the 2D face plane [218]. This process can express as follows:

$$V_p = f P_r R S_f + t \tag{6.2}$$

where V_p stores the coordinates of the 3D vertices projected onto the 2D plane, f is the scale factor, P_r is the orthographic projection matrix $P_r = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$, R is a rotation matrix consisting of 9 parameters and t is the translation vector.

The goal is to predict the coefficients $p = [f, R, t, \alpha_{id}, \alpha_{exp}]^T$ for rendering a 3D face, which includes a 40d identity parameter vector, a 10d expression parameter vector, and other 12d vector (f, R, t).

6.2.3 Loss functions

Much of the previous literature on 3D reconstruction [10, 130, 218] has described common loss functions that can be used to minimize the distance between the ground truth $\boldsymbol{p}^g$ and the model predicted parameter $\boldsymbol{p}^0$.

Weighted Parameter Distance Cost (WPDC) [10]: the researchers claimed that the parameters in the 3DMM contribute to the accuracy of fitting with different impacts. The basic idea is explicitly to model the importance of each parameter by Eq. 6.3.

$$E_{wpdc} = (\boldsymbol{p}^g - \boldsymbol{p}^0)^T W (\boldsymbol{p}^g - \boldsymbol{p}^0) \tag{6.3}$$

$$\begin{gathered} W = diag(w_1, ..., w_{62}) \\ w_i = ||V(\boldsymbol{p}^0(i)) - V(\boldsymbol{p}^g)|| / \sum w_i \end{gathered} \tag{6.4}$$

where the diagonal matrix W contains the weights of each parameter, w_i indicates the importance of the i-th coefficients, computed from how much error it introduces to the locations of 2D landmarks after projection.

Vertex Distance Cost (VDC) [10]: minimize the vertex distances between the fitted and the ground truth 3D face by Eq. 6.5.

$$E_{vdc} = ||V(\boldsymbol{p}^g) - V(\boldsymbol{p}^0)||^2 \tag{6.5}$$

where $V(\cdot)$ is the face construction and weak perspective projection as defined in Eq. 6.2.

For 3D face vertices reconstructed with the estimated 3D parameters, Wing Loss [55, 218] is able to adapt its shape to different types of ground truth heatmap pixels. The advantage of this loss function have a constant influence when error is large, so it will be robust to inaccurate annotations and occlusions. In this experiment, $\omega = 10, \epsilon = 2.1$ are positive values to be most effective.

$$E_{wing}(V(\boldsymbol{p}^g), V(\boldsymbol{p}^0)) = \begin{cases} \omega\, ln(1 + |\Delta T(P)|/\epsilon) \quad if \;\; |\Delta T(P)| < \omega \\ |\Delta T(P)| - C \quad otherwise \end{cases} \tag{6.6}$$

$$\Delta T(P) = V(\boldsymbol{p}^g) - V(\boldsymbol{p}^0); \;\; C = \omega - \omega\, ln(1 + \omega/\epsilon \tag{6.7}$$

For comparison, the network is optimized with the $L1$ and $L2$ loss function, respectively.

$$\begin{aligned} L1 &= \lambda_1 * E_{vdc} + E_{wpdc} \\ L2 &= \lambda_2 * E_{wing} + E_{wpdc} \end{aligned} \tag{6.8}$$

Fig. 6.9 also gives the results of different loss function with different λ_1 and λ_2 settings. We verified that $L2$ loss function with $\lambda_2 = 0.7$ can obtain better experimental results in our face alignment experiment.

6.3 Experiments

6.3.1 Evaluation metric

A commonly used metric for 3D face alignment tasks is the Normalized Mean Error (NME) as the average Euclidean distance between the predicted facial landmark locations and their corresponding ground-truth facial landmark locations. In this experiment, NME as the average landmark error normalized by the bounding box size [10, 218] instead of the common inter-pupil.

The formula for NME can write as Eq.6.9.

$$NME(k_i, k_i^*) = \frac{1}{N} \sum_{i=1}^{N} \frac{||k_i - k_i^*||_2}{\sqrt{wbbox \times hbbox}} \tag{6.9}$$

Herein, N is the number of facial landmarks of each images. k_i is the estimated landmarks points, k_i^* is their corresponding ground truth. *wbbox* and *hbbox* are defined as the width and height of the ground-truth bounding box.

6.3.2 Experimental results

Table 6.1 lists the mean error normalized by diagonal length of the bounding box and shows the comparison result (NME(%)) with the best results highlighted. The model is evaluated for the 3D dense face alignment task on the AFLW2000-3D and AFLW datasets, which are divided into three groups ([0°,30°], [30°,60°], [60°,90°]) by comparing it with several of the baseline methods, such as RCPR [7], ESR [23], SDM [191], 3DDFA [9], 3DFAN [58], DAMDNet [218], SRN [219].

Table 6.1: NME errors by diagonal length of the bounding box on the AFLW and AFLW2000-3D datasets.

	AFLW DataSet(21 pts)					AFLW 2000-3D Dataset(68 pts)				
Method	[0°,30°]	[30°,60°]	[60°,90°]	Mean	Std	[0°,30°]	[30°,60°]	[60°,90°]	Mean	Std
RCPR [7]	5.43	6.58	11.530	7.85	3.24	4.26	5.96	13.18	7.8	4.74
ESR [23]	5.66	7.12	11.94	8.24	3.29	4.6	6.7	12.67	7.99	4.19
SDM [191]	4.75	5.55	9.34	6.55	2.45	3.67	4.94	9.76	6.12	3.21
3DDFA [9]	5.00	5.06	6.74	5.60	0.99	3.78	4.54	7.93	5.42	2.21
3DDFA+SDM [9]	4.75	4.83	6.38	5.32	0.92	3.43	4.24	7.17	4.94	1.97
3DFAN [58]	-	-	-	-	-	3.38	4.46	5.59	4.48	1.11
SRN [219]	-	-	-	-	-	2.97	3.85	5.09	3.97	1.07
DAMDNet [218]	4.359	5.209	6.028	5.199	0.835	2.907	3.830	4.953	3.897	1.02
Ours((*wpdc_vdc_*1)	4.871	6.019	6.857	5.916	0.997	3.302	4.333	5.662	4.432	1.183
Ours(*wpdc_wing_*0.7)	**4.217**	**5.004**	**5.77**	**4.997**	**0.777**	**2.795**	**3.593**	**4.695**	**3.695**	**0.954**

Table 6.1 results show that in comparison to these baseline methods, the proposed method can achieve a lower mean (NME(%)). It achieves a 0.2% and 0.33% reduction in error on the AFLW datasets compared to DAMDNet [218] and 3DDFA+SDM [9]. For further comparison, the network is optimized with the different L1, L2 loss functions and different weights (e.g., Ours (*wpdc_wing_*0.7) indicates using the L2 from Eq. 6.8 with a λ_2 equal to 0.7). Experimental results demonstrate that the L2 loss function has the best results, and the overall mean NME (%) eventually drops to 4.432 from 3.741 on AFLW2000-3D when compared with the L1 loss function.

Fig. 6.9 shows the mean NME(%) values of optimization with different loss functions L1 and L2 in different λ_1, λ_2 for the proposed model. Lower NME is much better. The results show that optimizing the model with the loss function of L2 (blue) will be better than L1 (green). For the L1 loss function, the minimum value of NME(%) is obtained when the λ_1 is equal to 1, and for the L2 loss function, the model performs best when the λ_2 is equal to 0.7, which can significantly improve the face alignment accuracy in a full range of poses.

Table 6.2 compares current mainstream neural network structures with the proposed model. For the operational efficiency of lightweight networks, the commonly used evaluation factors are the

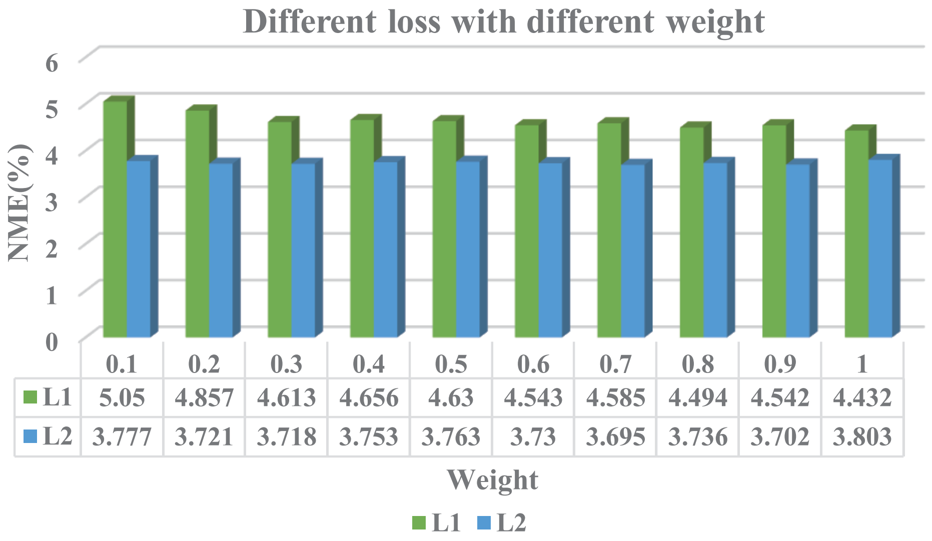


Figure 6.9: Model optimization results for different weights.

complexity of GFLOPs and the number of model parameters.

Table 6.2: NME errors on AFLW and AFLW2000-3D dataset. And comparisons of model size and complexity.

Method		MobileNet [215]	GhostNet [216]	DAMDNet[218]	Without SE	With SK	With CA	Ours
GFLops	-	0.70	**0.05**	0.12	0.12	0.17	0.12	0.12
Params	-	12.82M	3.98M	**2.76M**	4.41M	7.29M	4.65M	5.58M
AFLW Datasets (21 pts)	[0°,30°]	4.31	4.472	4.359	**4.204**	4.413	4.229	4.217
	[30°,60°]	5.044	5.285	5.209	5.058	5.187	5.054	**5.004**
	[60°,90°]	5.874	6.149	6.028	6.127	6.17	5.919	**5.77**
	Mean	5.076	5.302	5.199	5.13	5.256	5.067	**4.997**
	Std	0.782	0.839	0.835	0.963	0.881	0.845	**0.777**
AFLW 2000-3D (68 pts)	[0°,30°]	2.853	3.039	2.907	2.797	2.995	2.862	**2.795**
	[30°,60°]	3.757	3.9	3.83	3.629	3.9	3.693	**3.593**
	[60°,90°]	4.921	5.2	4.953	4.834	5.26	4.812	**4.695**
	Mean	3.844	4.047	3.897	3.753	4.052	3.789	**3.695**
	Std	1.037	1.088	1.02	1.025	1.14	0.979	**0.954**

In Table 6.2, even the GFLOPs and parameters of the proposed framework are much higher than GhostNet's GFLOPs and DAMDNet's parameters, but in terms of accuracy, the result of the proposed experiment performs better. Compared to GhostNet, the proposed method reduces face alignment error by 0.3% in the AFLW dataset and 0.35% in the AFLW2000-3D dataset. Here, the experiments compared the original network without attention mechanism, with SK attention mechanism [212], and with the CA attention mechanism [213]. In comparison with CA and SK attention mechanisms, the SE module can improve the precision. In summary, the proposed framework can significantly improve the accuracy of the network without adding too many network parameters and GFLOPs.

Some visualization results of the proposed method are shown in Fig. 6.10 and Fig. 6.11. The proposed algorithm not only predicts the key points but also estimates the 3D face structure.

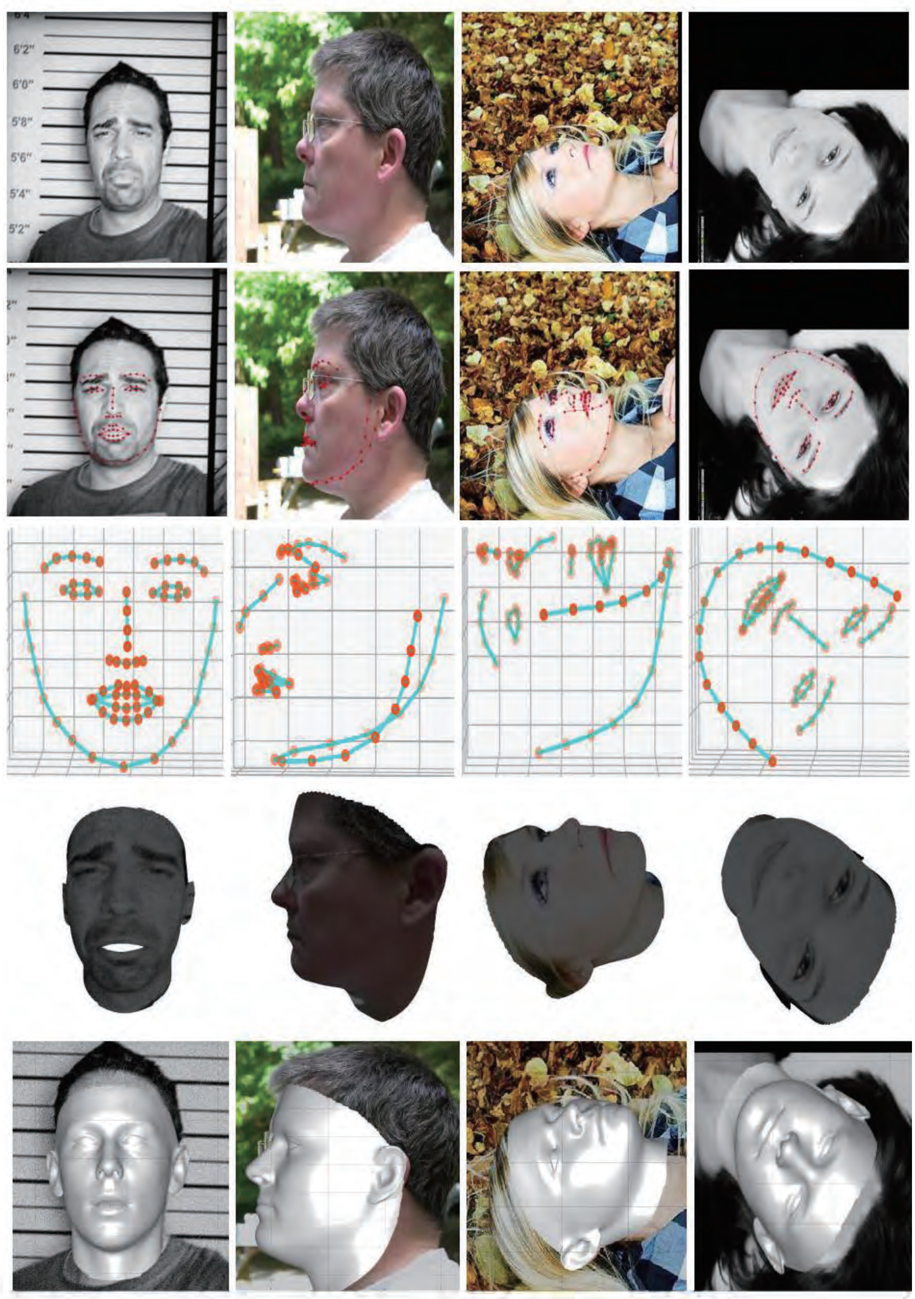


Figure 6.10: Face alignment and 3D face reconstruction visualization results of the proposed method.

Top row are the input images which come from Menpo-3D [1], AFLW2000-3D [10] and 300W-test-3D [6]. Second and third rows are the 3D landmarks plotted for different display views. Fourth and fifth rows are the 3D face model with the texture image of the face and the reconstructed 3D face projection on the input images.

Fig. 6.11 shows more examples that are randomly chosen from challenging situations.

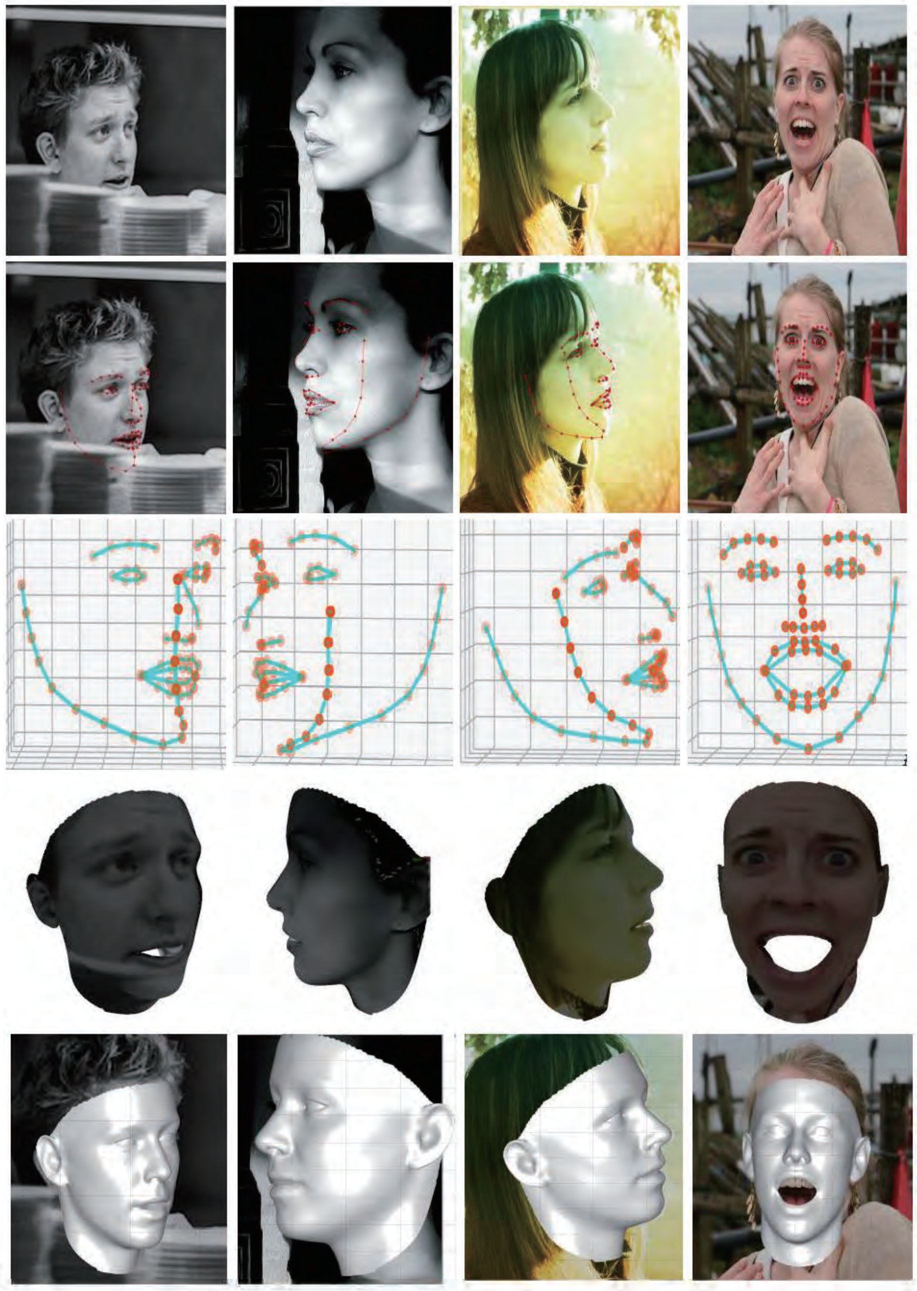


Figure 6.11: Face alignment and face reconstruction visualization results on challenging datasets.

Fig. 6.10 and Fig. 6.11 show the results where the proposed framework is robust to occlusions, illumination, and large pose and expression challenging conditions.

According to [9, 10], it can generate the Projected Normalized Coordinate Code (PNCC) information (see Fig. 6.12 (b)). The authors define the Normalized Coordinate Code (NCC) (see Fig. 6.12 (a)), which can be considered as a vertex index. In the fitting process, with a model parameter p, the author applies ZBuffer to render the projected 3D face colored by NCC.

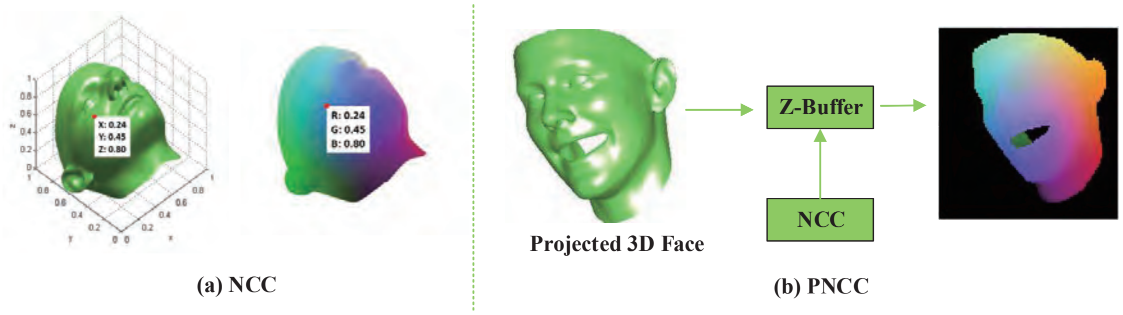


Figure 6.12: Generation of PNCC: the projected 3D faces are rendered by Z-Buffer with NCC [9].

In Fig. 6.13, the top row is the input images, which show in the above figures, second and third rows represent the images of the predicted 3D pose and depth estimates, respectively.

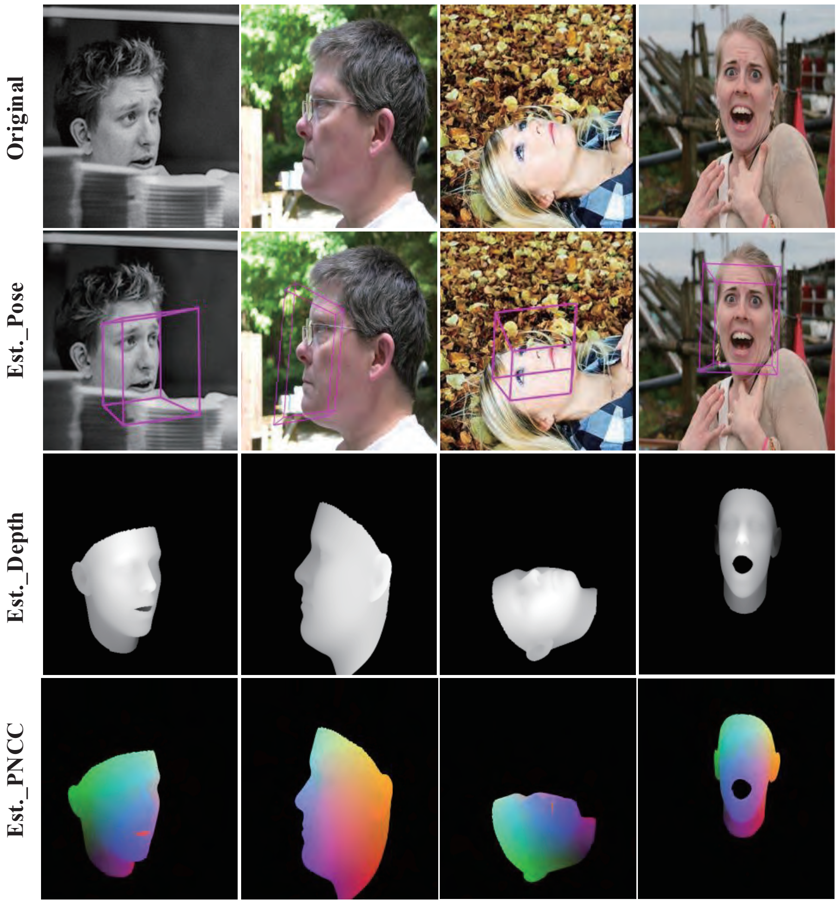


Figure 6.13: Face pose and depth estimate and PNCC results.

In real-life scenarios, faces with multiple challenging elements, such as simultaneously occluded and large pose faces and large expressions faces due to blurring. The algorithm (RA+G) described in Chapter 5 fails for face keypoint detection when dealing with large occlusions and expressions.

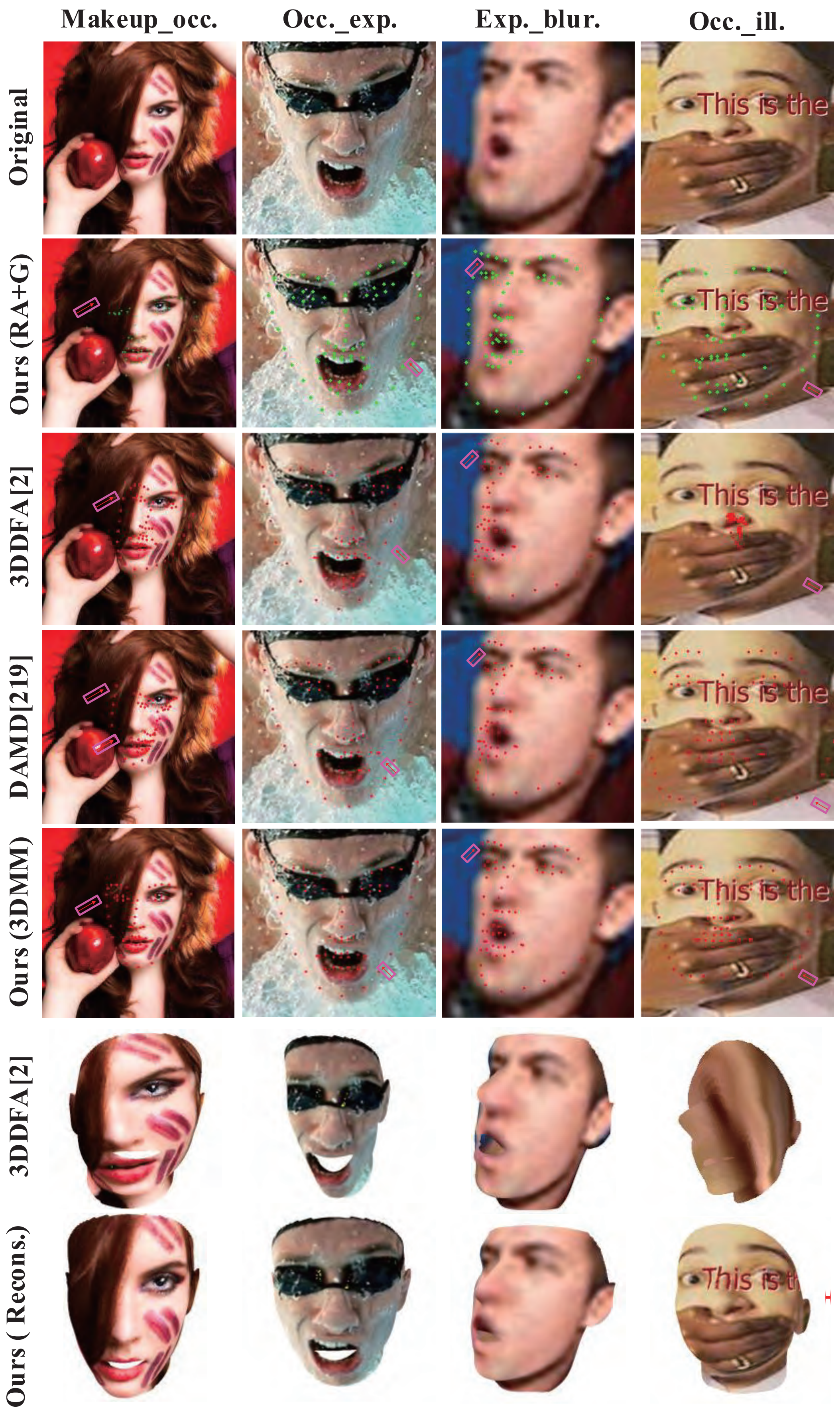


Figure 6.14: Comparative experiments of the algorithms in Chapter 3 and Chapter 4 under challenging environments (e.g.,occlusion and large expressions).

Key points predicted by the RA+G algorithm are marked in green, while the proposed 3DMM-based algorithm mark in red. Other comparative results with latest method (e.g., 3DDFA [9], DAMDNet [218], Hourglass [58]) to show the different. Face alignment and face reconstruction results illustrate the proposed algorithm's robustness even under multiple challenging conditions.

6.4 Summary

In this section, the goal is to regress the 3D geometry of the face and its dense counterpart information. Instead of a sparse feature point shape, the proposed algorithm is based on a 3D dense face model to match the face image. By adding 3D information, appearance changes and self-obscuration caused by 3D transformations can be addressed. The main contributions are summarized as below: Inspired by the effectiveness of the Lightweight network [86, 87], which exploits pointwise group convolution and channel shuffle. The proposed network structure combines three major achievements: 3DMM [34, 85], ShuffleNetV2 Plus series of units [86, 87], and Squeeze-and-excitation (SE) attention mechanism [88] to improve the representation ability of the network. For 3D face reconstruction, the proposed network significantly reduces the computational cost while maintaining the model accuracy. Compared to several popular networks, the proposed algorithm achieves better performance between accuracy and efficiency. In addition, the experimental results in Table 6.2 demonstrate the effectiveness of the SE attention mechanism embedded in the proposed network. The visualization results indicate that the proposed framework can handle face reconstruction across multiple challenging variations.

Chapter 7

Discussion and future work

Large-scale real-world datasets are difficult to collect and hard to control in terms of variability in the dataset. It can not annotate all the variations of interest conditions such as pose, expression, or illumination. In addition, the labor-intensive ground truth annotation process is also error-prone and time-consuming. However, the advantage of synthetic data is that it provides controlled and detailed reliable annotation at no cost. The experiments demonstrate that synthetic face images are complementary to biased real-world datasets. And combining synthetic with real data sets can improve performance in face alignment and face reconstruction.

7.1 Conclusion

This paper focuses on analyzing the main challenges facing face datasets today in Chapter 1. Chapter 2 lists the previous work related to this research. Chapter 3 to Chapter 5 present the synthetic datasets on face alignment in detail.

1) The advantages of using synthetic datasets were first analyzed and investigated. A synthetic database was created based on FaceGen's model and added to the face alignment network model for training. The phenomenon can indicate that using synthetic datasets to train a competitive face alignment system reduces the required real training dataset.

2) An improved bidirectional transformation of the GAN ($R \to S$) model is proposed to address the large "gap" between the generated synthetic dataset and the real-world dataset. When testing real-world images, the given test image is first transformed using the $R \to S$ network.

Then predicting the coordinate points of the transformed image using the alignment network, and finally corresponding the coordinate point location information to the original test input image.

3) The optimized GAN ($S \rightarrow R$ model) is used to convert the synthetic images into real datasets as much as possible to increase the dataset for the training model. The results show that those transformed synthetic datasets can further improve the detection accuracy of the model.

The experimental results in Chapter 3 to Chapter 5 demonstrate that combining large-scale real-world data with synthetic data can improve performance. It suggests that the additional variability of the synthetic data in terms of pose and facial identity is a key factor in improving the performance of face alignment. Finally, in Chapter 6, since FaceGen's model is still limited in generating the diversity of faces (especially for the multiple challenging factors). The synthetic datasets generated by 3D Face Rendering (300W-LP) are used to solve the 3D reconstruction of faces. These datasets can be used to close real-to-virtual performance gaps. The proposed algorithm can handle face reconstruction and face alignment even under multiple challenging conditions. Chapter 7 summarises the main contributions and overviews of each chapter and gives directions for future work.

7.2 Face Image Generator

This work opens up future research directions through potential synthetic datasets. Especially data limiting fields, the generation of the synthetic datasets, and augmentation using statistical shape models would offer an alternative approach. In the previous works, Adam et al. [220] introduce the data generator that generates synthetic face images with a precise annotation such as shape and texture, but also of nuisance parameters such as light, camera, and head pose.

The generator relies on a 3DMM of the shape, color, and expression. It exploits the Basel Face Model (BFM2017) [146] which is learned from 200 neutral face scans and 160 expression deformations. They also sample spherical harmonics illumination parameters from the Basel Illumination Prior (BIP) [221] to obtain natural illumination from synthetic face images. It can also choose random background textures from the data provided in the describable texture database by using a non-parametric background model [222] (see Fig. 7.1 top row images). Lastly, a 2D image is generated from a sample of the 3D model using computer graphics. You can control the variation of parameters such as pose, shape, color, camera, and illumination based on your demand and application.

Fig. 7.1 shows some synthesized examples from the generator. It can simulate background changes with an anon-parametric background model by sampling randomly from a set of background images of the describable texture database [223] (see Fig. 7.1 second and third rows images).

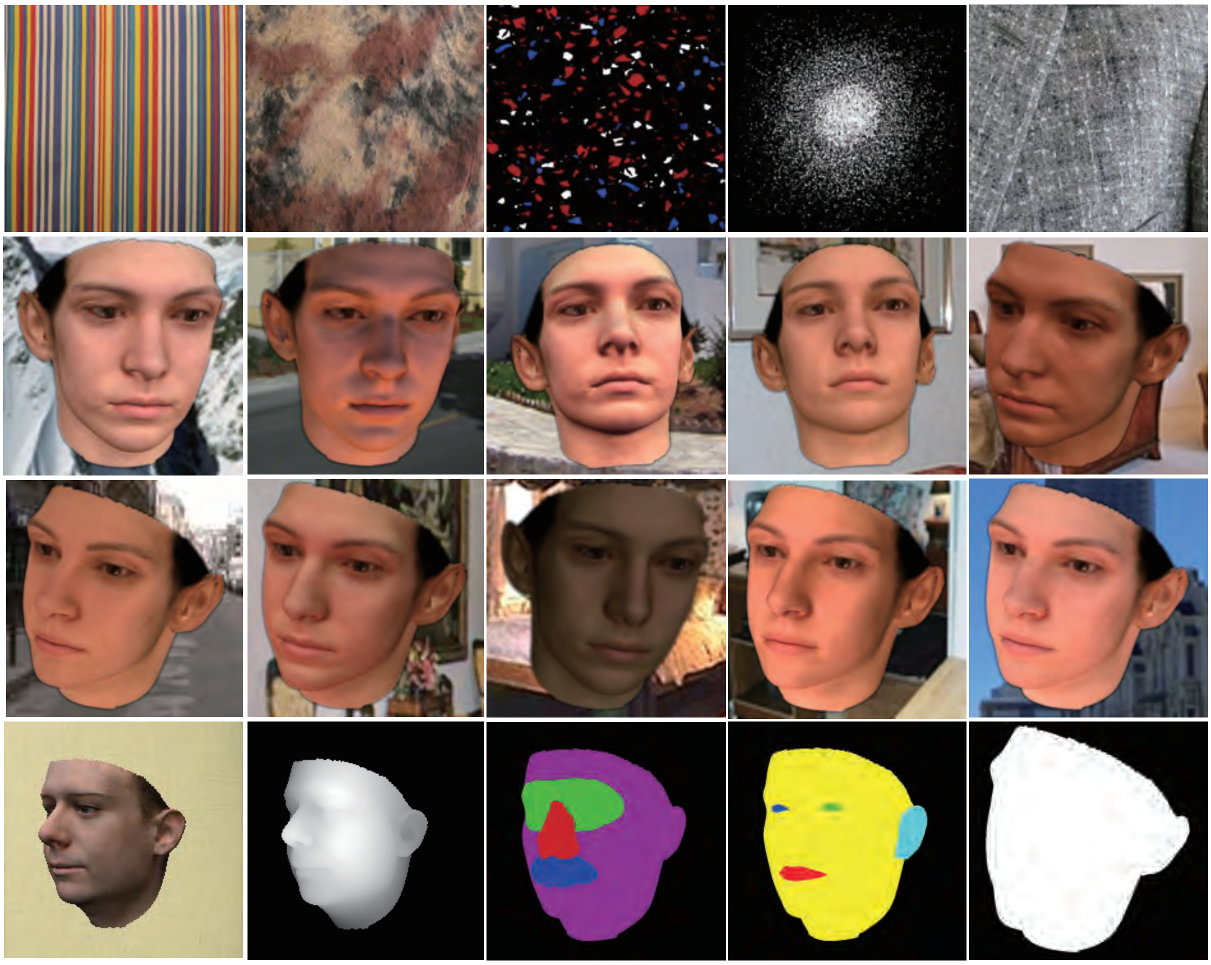


Figure 7.1: Synthetic face images sampled renderings generated by parametric image generator.

Such synthetics images can be used to do more face-related research. For example, you can render different region maps for face segmentation, while it provides two default ones (see Fig. 7.1 the third and fourth images of the last row). For face reconstruction, it gives the texture image of face and depth images (see Fig. 7.1 the first and second images of the last row). You can use this generator to generate images with arbitrary amounts of facial features and with control variations of the image, such as pose, lighting, and background. For example, the second and third rows in Fig. 7.1 show different samples of the same facial identity.

Fig. 7.2 shows samples of generating synthetic datasets in various viewpoints, reflectance, and illumination. You can render various image modalities such as depth images, color-coded (PNCC) correspondence images, normals, albedo, and illumination. The parameters of the model are sampled randomly from a custom distribution. For each face sample, the location and visibility of facial landmarks are written in an file in the following format: facial landmark name, visibility, x position, and y position.

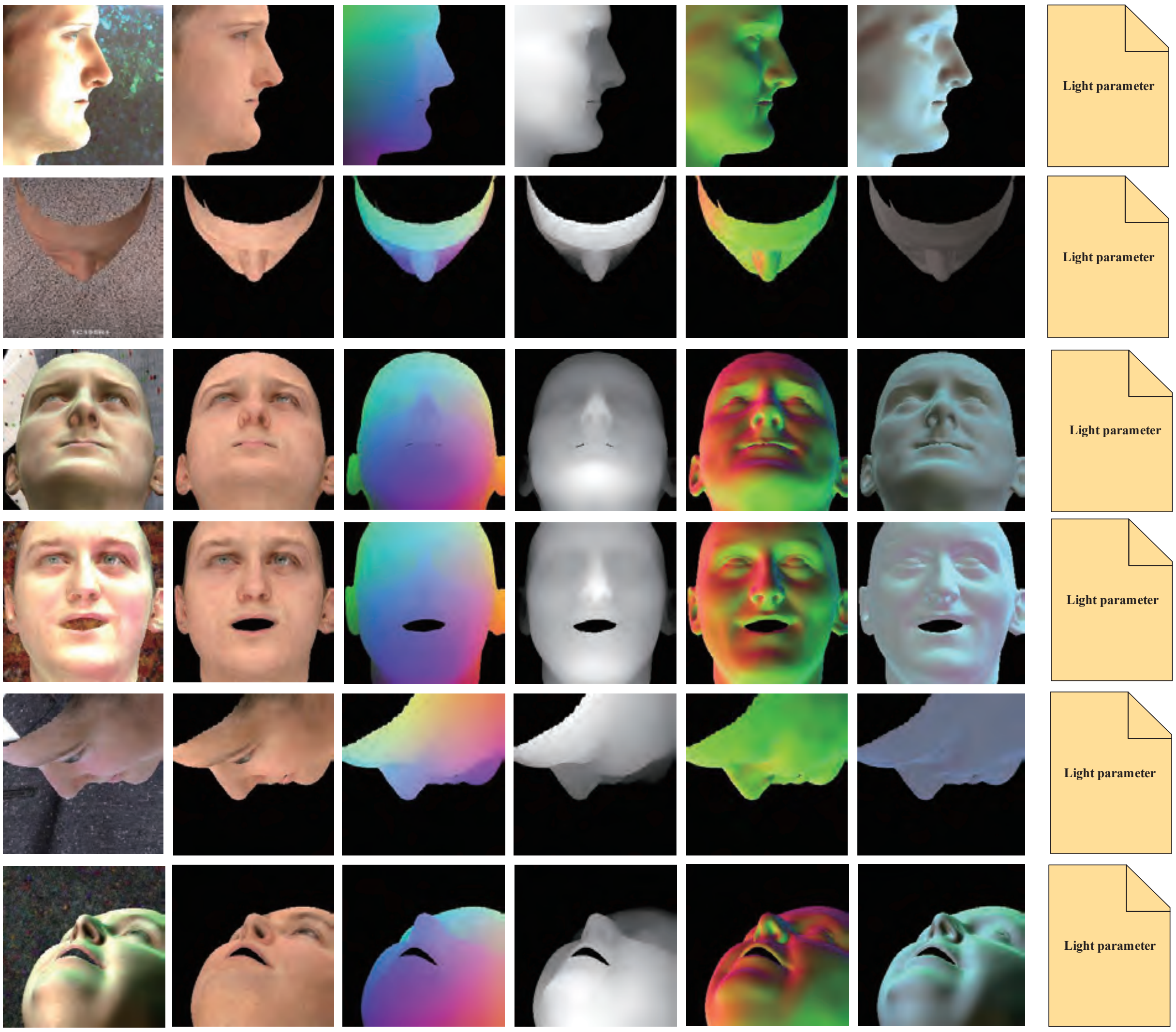


Figure 7.2: More render smaples of various image modalities.

7.3 Future work

The goal is to use GANs method to generate the normal, albedo, and shading images directly for 3D face reconstruction, rather than predicting the parameters of a 3DMM by using the complex structure of deep learning. The training datasets include synthetic data with ground-truth normal, albedo, and lighting combined with real-world images.

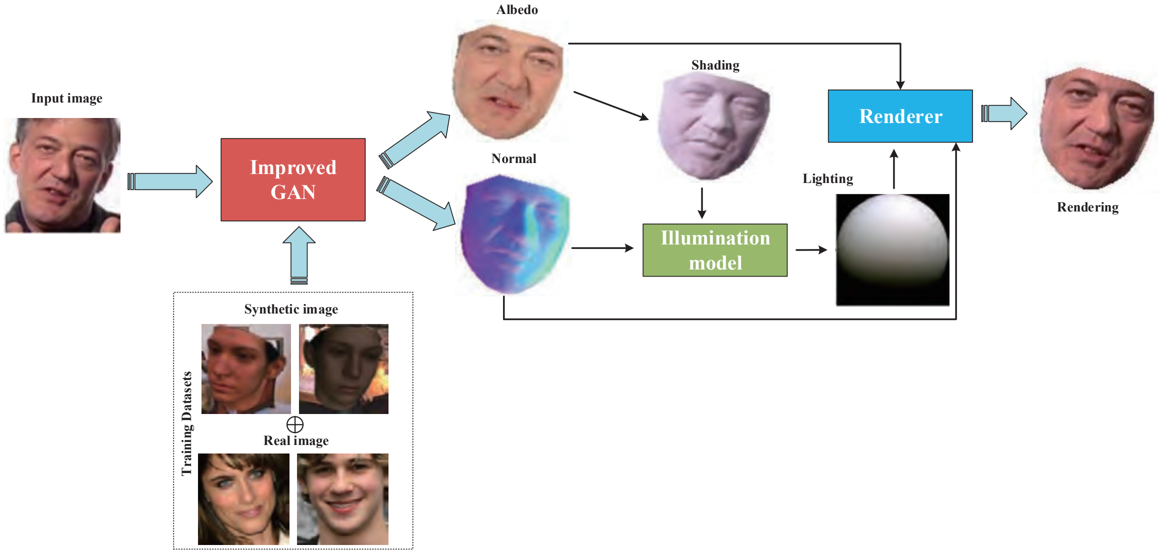


Figure 7.3: Structure for generating normal, albedo, and shading images directly from synthetic data for face 3D reconstruction.

The structure direction is described in Fig. 7.3. Following [75, 224], the image formation process under Lambertian reflectance is represented in Eq. 7.1.

$$I(p) = f_{render}(N(p), A(p), L) \tag{7.1}$$

where f_{render} is a differentiable function. $N(p)$, $A(p)$, and $I(p)$ are the normal, albedo, and image intensity at each pixel p, respectively. Lighting L is defined as nine-dimensional second-order spherical harmonics coefficients for each of the RGB channels. Normally, models are used 27-dimensional spherical harmonics coefficients (9 for each RGB channel).

Bibliography

[1] Stefanos Zafeiriou, George Trigeorgis, Grigorios Chrysos, Jiankang Deng, and Jie Shen. The menpo facial landmark localisation challenge: A step towards the solution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 170–179, 2017.

[2] Martin Koestinger, Paul Wohlhart, Peter M Roth, and Horst Bischof. Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. In *2011 IEEE international conference on computer vision workshops (ICCV workshops)*, pages 2144–2151. IEEE, 2011.

[3] Xiangxin Zhu and Deva Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *2012 IEEE conference on computer vision and pattern recognition*, pages 2879–2886. IEEE, 2012.

[4] Vuong Le, Jonathan Brandt, Zhe Lin, Lubomir Bourdev, and Thomas S Huang. Interactive facial feature localization. In *European conference on computer vision*, pages 679–692. Springer, 2012.

[5] Peter N Belhumeur, David W Jacobs, David J Kriegman, and Neeraj Kumar. Localizing parts of faces using a consensus of exemplars. *IEEE transactions on pattern analysis and machine intelligence*, 35(12):2930–2940, 2013.

[6] Christos Sagonas, Georgios Tzimiropoulos, Stefanos Zafeiriou, and Maja Pantic. 300 faces in-the-wild challenge: The first facial landmark localization challenge. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 397–403, 2013.

[7] Xavier P Burgos-Artizzu, Pietro Perona, and Piotr Dollár. Robust face landmark estimation

under occlusion. In *Proceedings of the IEEE international conference on computer vision*, pages 1513–1520, 2013.

[8] Hang Du, Hailin Shi, Dan Zeng, and Tao Mei. The elements of end-to-end deep face recognition: A survey of recent advances. *arXiv preprint arXiv:2009.13290*, 2020.

[9] Xiangyu Zhu, Zhen Lei, Xiaoming Liu, Hailin Shi, and Stan Z Li. Face alignment across large poses: A 3d solution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 146–155, 2016.

[10] Xiangyu Zhu, Xiaoming Liu, Zhen Lei, and Stan Z Li. Face alignment in full pose range: A 3d total solution. *IEEE transactions on pattern analysis and machine intelligence*, 41(1):78–92, 2017.

[11] Feng-Ju Chang, Anh Tuan Tran, Tal Hassner, Iacopo Masi, Ram Nevatia, and Gerard Medioni. Faceposenet: Making a case for landmark-free face alignment. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 1599–1608, 2017.

[12] Iacopo Masi, Anh Tuấn Trần, Tal Hassner, Jatuporn Toy Leksut, and Gérard Medioni. Do we really need to collect millions of faces for effective face recognition? In *European conference on computer vision*, pages 579–596. Springer, 2016.

[13] Xiangyu Zhu, Zhen Lei, Xiaoming Liu, Hailin Shi, and Stan Z Li. Supplemental material for face alignment across large poses: A 3d solution. *Database*, 2:300W–3D, 2016.

[14] Araceli Morales, Gemma Piella, and Federico M Sukno. Survey on 3d face reconstruction from uncalibrated images. *Computer Science Review*, 40:100400, 2021.

[15] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001*, volume 1, pages I–I. IEEE, 2001.

[16] Paul Viola and Michael J Jones. Robust real-time face detection. *International journal of computer vision*, 57(2):137–154, 2004.

[17] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[18] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pages 886–893. Ieee, 2005.

[19] Stefanos Zafeiriou, Cha Zhang, and Zhengyou Zhang. A survey on face detection in the wild: past, present and future. *Computer Vision and Image Understanding*, 138:1–24, 2015.

[20] Timothy F Cootes, Christopher J Taylor, David H Cooper, and Jim Graham. Active shape models-their training and application. *Computer vision and image understanding*, 61(1):38–59, 1995.

[21] Timothy F Cootes, Gareth J Edwards, and Christopher J Taylor. Active appearance models. In *European conference on computer vision*, pages 484–498. Springer, 1998.

[22] Brais Martinez, Michel F Valstar, Xavier Binefa, and Maja Pantic. Local evidence aggregation for regression-based facial point detection. *IEEE transactions on pattern analysis and machine intelligence*, 35(5):1149–1163, 2012.

[23] Xudong Cao, Yichen Wei, Fang Wen, and Jian Sun. Face alignment by explicit shape regression. *International Journal of Computer Vision*, 107(2):177–190, 2014.

[24] Xin Jin and Xiaoyang Tan. Face alignment in-the-wild: A survey. *Computer Vision and Image Understanding*, 162:1–22, 2017.

[25] Nannan Wang, Xinbo Gao, Dacheng Tao, Heng Yang, and Xuelong Li. Facial feature point detection: A comprehensive survey. *Neurocomputing*, 275:50–65, 2018.

[26] Ki-Chung Chung, Seok Cheol Kee, and Sang Ryong Kim. Face recognition using principal component analysis of gabor filter responses. In *Proceedings International Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems. In Conjunction with ICCV'99 (Cat. No. PR00378)*, pages 53–57. IEEE, 1999.

[27] Timo Ahonen, Abdenour Hadid, and Matti Pietikäinen. Face recognition with local binary patterns. In *European conference on computer vision*, pages 469–481. Springer, 2004.

[28] Kamran Etemad and Rama Chellappa. Discriminant analysis for recognition of human face images. *Josa a*, 14(8):1724–1733, 1997.

[29] Xiaofei He and Partha Niyogi. Locality preserving projections. *Advances in neural information processing systems*, 16(16):153–160, 2004.

[30] Guodong Guo, Stan Z Li, and Kapluk Chan. Face recognition by support vector machines. In *Proceedings fourth IEEE international conference on automatic face and gesture recognition (cat. no. PR00580)*, pages 196–201. IEEE, 2000.

[31] Guo-Dong Guo and Hong-Jiang Zhang. Boosting for fast face recognition. In *Proceedings IEEE ICCV Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems*, pages 96–100. IEEE, 2001.

[32] Iacopo Masi, Yue Wu, Tal Hassner, and Prem Natarajan. Deep face recognition: A survey. In *2018 31st SIBGRAPI conference on graphics, patterns and images (SIBGRAPI)*, pages 471–478. IEEE, 2018.

[33] Guodong Guo and Na Zhang. A survey on deep learning based face recognition. *Computer vision and image understanding*, 189:102805, 2019.

[34] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 187–194, 1999.

[35] Andrzej Maćkiewicz and Waldemar Ratajczak. Principal components analysis (pca). *Computers & Geosciences*, 19(3):303–342, 1993.

[36] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.

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

[38] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10):1499–1503, 2016.

[39] Maxime Oquab, Leon Bottou, Ivan Laptev, and Josef Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1717–1724, 2014.

[40] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[41] B Yang, J Yan, Z Lei, and SZ Li. Convolutional channel features for pedestrian. *Face and Edge Detection, ICCV*, 2015.

[42] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[43] Shuo Yang, Ping Luo, Chen-Change Loy, and Xiaoou Tang. From facial parts responses to face detection: A deep learning approach. In *Proceedings of the IEEE international conference on computer vision*, pages 3676–3684, 2015.

[44] Sachin Sudhakar Farfade, Mohammad J Saberian, and Li-Jia Li. Multi-view face detection using deep convolutional neural networks. In *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*, pages 643–650, 2015.

[45] Hongwei Qin, Junjie Yan, Xiu Li, and Xiaolin Hu. Joint training of cascaded cnn for face detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3456–3465, 2016.

[46] Haoxiang Li, Zhe Lin, Xiaohui Shen, Jonathan Brandt, and Gang Hua. A convolutional neural network cascade for face detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5325–5334, 2015.

[47] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *arXiv preprint arXiv:1506.01497*, 2015.

[48] Xudong Sun, Pengcheng Wu, and Steven CH Hoi. Face detection using deep learning: An improved faster rcnn approach. *Neurocomputing*, 299:42–50, 2018.

[49] Wang Yang and Zheng Jiachun. Real-time face detection based on yolo. In *2018 1st IEEE international conference on knowledge innovation and invention (ICKII)*, pages 221–224. IEEE, 2018.

[50] Jiankang Deng, Jia Guo, Evangelos Ververas, Irene Kotsia, and Stefanos Zafeiriou. Retinaface: Single-shot multi-level face localisation in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5203–5212, 2020.

[51] Yi Sun, Xiaogang Wang, and Xiaoou Tang. Deep convolutional network cascade for facial point detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3476–3483, 2013.

[52] Erjin Zhou, Haoqiang Fan, Zhimin Cao, Yuning Jiang, and Qi Yin. Extensive facial landmark localization with coarse-to-fine convolutional network cascade. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 386–391, 2013.

[53] Zhanpeng Zhang, Ping Luo, Chen Change Loy, and Xiaoou Tang. Facial landmark detection by deep multi-task learning. In *European conference on computer vision*, pages 94–108. Springer, 2014.

[54] Maheen Rashid, Xiuye Gu, and Yong Jae Lee. Interspecies knowledge transfer for facial keypoint detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6894–6903, 2017.

[55] Zhen-Hua Feng, Josef Kittler, Muhammad Awais, Patrik Huber, and Xiao-Jun Wu. Wing loss for robust facial landmark localisation with convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2235–2245, 2018.

[56] Jiangjing Lv, Xiaohu Shao, Junliang Xing, Cheng Cheng, and Xi Zhou. A deep regression architecture with two-stage re-initialization for high performance facial landmark detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3317–3326, 2017.

[57] Jing Yang, Qingshan Liu, and Kaihua Zhang. Stacked hourglass network for robust facial landmark localisation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 79–87, 2017.

[58] Adrian Bulat and Georgios Tzimiropoulos. How far are we from solving the 2d & 3d face alignment problem?(and a dataset of 230,000 3d facial landmarks). In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1021–1030, 2017.

[59] Shaoqing Ren, Xudong Cao, Yichen Wei, and Jian Sun. Face alignment at 3000 fps via regressing local binary features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1685–1692, 2014.

[60] Jiankang Deng, George Trigeorgis, Yuxiang Zhou, and Stefanos Zafeiriou. Joint multi-view face alignment in the wild. *IEEE Transactions on Image Processing*, 28(7):3636–3648, 2019.

[61] Amin Jourabloo and Xiaoming Liu. Large-pose face alignment via cnn-based dense 3d model fitting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4188–4196, 2016.

[62] Yao Feng, Fan Wu, Xiaohu Shao, Yanfeng Wang, and Xi Zhou. Joint 3d face reconstruction and dense alignment with position map regression network. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 534–551, 2018.

[63] Xiaoguang Tu, Jian Zhao, Zihang Jiang, Yao Luo, Mei Xie, Yang Zhao, Linxiao He, Zheng Ma, and Jiashi Feng. Joint 3d face reconstruction and dense face alignment from a single image with 2d-assisted self-supervised learning. *arXiv preprint arXiv:1903.09359*, 1(2), 2019.

[64] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014.

[65] Ming Yang, Marc’ Aurelio Ranzato, Lior Wolf, and Yaniv Taigman. Web-scale training for face identification.

[66] Yi Sun, Xiaogang Wang, and Xiaoou Tang. Deep learning face representation from predicting 10,000 classes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1891–1898, 2014.

[67] Yi Sun, Ding Liang, Xiaogang Wang, and Xiaoou Tang. Deepid3: Face recognition with very deep neural networks. *arXiv preprint arXiv:1502.00873*, 2015.

[68] Xiaofeng Liu, BVK Kumar, Chao Yang, Qingming Tang, and Jane You. Dependency-aware attention control for unconstrained face recognition with image sets. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 548–565, 2018.

[69] Lingxiao Song, Man Zhang, Xiang Wu, and Ran He. Adversarial discriminative heterogeneous face recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[70] Feng Liu, Ronghang Zhu, Dan Zeng, Qijun Zhao, and Xiaoming Liu. Disentangling features in 3d face shapes for joint face reconstruction and recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5216–5225, 2018.

[71] Pengfei Dou, Shishir K Shah, and Ioannis A Kakadiaris. End-to-end 3d face reconstruction with deep neural networks. In *proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5908–5917, 2017.

[72] Anh Tuan Tran, Tal Hassner, Iacopo Masi, and Gérard Medioni. Regressing robust and discriminative 3d morphable models with a very deep neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5163–5172, 2017.

[73] Yu Deng, Jiaolong Yang, Sicheng Xu, Dong Chen, Yunde Jia, and Xin Tong. Accurate 3d face reconstruction with weakly-supervised learning: From single image to image set. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[74] Kyle Genova, Forrester Cole, Aaron Maschinot, Aaron Sarna, Daniel Vlasic, and William T Freeman. Unsupervised training for 3d morphable model regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8377–8386, 2018.

[75] Soumyadip Sengupta, Angjoo Kanazawa, Carlos D Castillo, and David W Jacobs. Sfsnet: Learning shape, reflectance and illuminance of facesin the wild'. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6296–6305, 2018.

[76] A Martinez and R Benavente. The ar face database, cvc. *Copyright of Informatica (03505596)*, 1998.

[77] Kieron Messer, Jiri Matas, Josef Kittler, Juergen Luettin, and Gilbert Maitre. Xm2vtsdb: The extended m2vts database. In *Second international conference on audio and video-based biometric person authentication*, volume 964, pages 965–966, 1999.

[78] Oliver Jesorsky, Klaus J Kirchberg, and Robert W Frischholz. Robust face detection using the hausdorff distance. In *International conference on audio-and video-based biometric person authentication*, pages 90–95. Springer, 2001.

[79] Michael M Nordstrøm, Mads Larsen, Janusz Sierakowski, and Mikkel Bille Stegmann. The imm face database-an annotated dataset of 240 face images. 2004.

[80] Gary B Huang, Marwan Mattar, Tamara Berg, and Eric Learned-Miller. Labeled faces in the wild: A database forstudying face recognition in unconstrained environments. In *Workshop on faces in'Real-Life'Images: detection, alignment, and recognition*, 2008.

[81] Andrzej Kasinski, Andrzej Florek, and Adam Schmidt. The put face database. *Image Processing and Communications*, 13(3-4):59–64, 2008.

[82] Wayne Wu, Chen Qian, Shuo Yang, Quan Wang, Yici Cai, and Qiang Zhou. Look at boundary: A boundary-aware face alignment algorithm. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2129–2138, 2018.

[83] Yinglu Liu, Hao Shen, Yue Si, Xiaobo Wang, Xiangyu Zhu, Hailin Shi, Zhibin Hong, Hanqi Guo, Ziyuan Guo, Yanqin Chen, et al. Grand challenge of 106-point facial landmark localization. In *2019 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, pages 613–616. IEEE, 2019.

[84] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *arXiv preprint arXiv:1406.2661*, 2014.

[85] Volker Blanz, Albert Mehl, Thomas Vetter, and H-P Seidel. A statistical method for robust 3d surface reconstruction from sparse data. In *Proceedings. 2nd International Symposium on 3D Data Processing, Visualization and Transmission, 2004. 3DPVT 2004.*, pages 293–300. IEEE, 2004.

[86] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6848–6856, 2018.

[87] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *Proceedings of the European conference on computer vision (ECCV)*, pages 116–131, 2018.

[88] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.

[89] Yaojie Liu, Amin Jourabloo, William Ren, and Xiaoming Liu. Dense face alignment. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 1619–1628, 2017.

[90] Hongwen Zhang, Qi Li, Zhenan Sun, and Yunfan Liu. Combining data-driven and model-driven methods for robust facial landmark detection. *IEEE Transactions on Information Forensics and Security*, 13(10):2409–2422, 2018.

[91] Xinyao Wang, Liefeng Bo, and Li Fuxin. Adaptive wing loss for robust face alignment via heatmap regression. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6971–6981, 2019.

[92] Akshay Asthana, Stefanos Zafeiriou, Shiyang Cheng, and Maja Pantic. Incremental face alignment in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1859–1866, 2014.

[93] Jie Zhang, Shiguang Shan, Meina Kan, and Xilin Chen. Coarse-to-fine auto-encoder networks (cfan) for real-time face alignment. In *European conference on computer vision*, pages 1–16. Springer, 2014.

[94] George Trigeorgis, Patrick Snape, Mihalis A Nicolaou, Epameinondas Antonakos, and Stefanos Zafeiriou. Mnemonic descent method: A recurrent process applied for end-to-end face alignment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4177–4187, 2016.

[95] Shengtao Xiao, Jiashi Feng, Junliang Xing, Hanjiang Lai, Shuicheng Yan, and Ashraf Kassim. Robust facial landmark detection via recurrent attentive-refinement networks. In *European conference on computer vision*, pages 57–72. Springer, 2016.

[96] Yue Wu, Tal Hassner, KangGeon Kim, Gerard Medioni, and Prem Natarajan. Facial landmark detection with tweaked convolutional neural networks. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):3067–3074, 2017.

[97] Xin Miao, Xiantong Zhen, Xianglong Liu, Cheng Deng, Vassilis Athitsos, and Heng Huang. Direct shape regression networks for end-to-end face alignment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5040–5049, 2018.

[98] Lei Yue, Xin Miao, Pengbo Wang, Baochang Zhang, Xiantong Zhen, and Xianbin Cao. Attentional alignment networks. In *BMVC*, volume 2, page 7, 2018.

[99] Xuanyi Dong, Shoou-I Yu, Xinshuo Weng, Shih-En Wei, Yi Yang, and Yaser Sheikh. Supervision-by-registration: An unsupervised approach to improve the precision of facial landmark detectors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 360–368, 2018.

[100] Meilu Zhu, Daming Shi, Mingjie Zheng, and Muhammad Sadiq. Robust facial landmark detection via occlusion-adaptive deep networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3486–3496, 2019.

[101] Xiaojie Guo, Siyuan Li, Jinke Yu, Jiawan Zhang, Jiayi Ma, Lin Ma, Wei Liu, and Haibin Ling. Pfld: a practical facial landmark detector. *arXiv preprint arXiv:1902.10859*, 2019.

[102] Yuanyuan Xu, Wan Yan, Genke Yang, Jiliang Luo, Tao Li, and Jianan He. Centerface: joint face detection and alignment using face as point. *Scientific Programming*, 2020, 2019.

[103] Adrian Bulat and Georgios Tzimiropoulos. Human pose estimation via convolutional part heatmap regression. In *European Conference on Computer Vision*, pages 717–732. Springer, 2016.

[104] Lichao Huang, Yi Yang, Yafeng Deng, and Yinan Yu. Densebox: Unifying landmark localization with end to end object detection. *arXiv preprint arXiv:1509.04874*, 2015.

[105] Adrian Bulat and Georgios Tzimiropoulos. Convolutional aggregation of local evidence for large pose face alignment. 2016.

[106] Xi Peng, Rogerio S Feris, Xiaoyu Wang, and Dimitris N Metaxas. A recurrent encoder-decoder network for sequential face alignment. In *European conference on computer vision*, pages 38–56. Springer, 2016.

[107] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *European conference on computer vision*, pages 483–499. Springer, 2016.

[108] Jiankang Deng, Yuxiang Zhou, Shiyang Cheng, and Stefanos Zaferiou. Cascade multi-view hourglass model for robust 3d face alignment. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pages 399–403. IEEE, 2018.

[109] Amit Kumar and Rama Chellappa. Disentangling 3d pose in a dendritic cnn for unconstrained 2d face alignment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 430–439, 2018.

[110] Xuanyi Dong, Yan Yan, Wanli Ouyang, and Yi Yang. Style aggregated network for facial landmark detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 379–388, 2018.

[111] Keqiang Sun, Wayne Wu, Tinghao Liu, Shuo Yang, Quan Wang, Qiang Zhou, Zuochang Ye, and Chen Qian. Fab: A robust facial landmark detection framework for motion-blurred videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5462–5471, 2019.

[112] Lisha Chen, Hui Su, and Qiang Ji. Face alignment with kernel density deep neural network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6992–7002, 2019.

[113] Shangzhe Wu, Christian Rupprecht, Andrea Vedaldi, Jamie Watson, Michael Firman, Aron Monszpart, Simron Thapa, Nianyi Li, Jinwei Ye, Boyang Deng, et al. Cvpr 2020.

[114] Yang Zhao, Yifan Liu, Chunhua Shen, Yongsheng Gao, and Shengwu Xiong. Mobilefan: transferring deep hidden representation for face alignment. *Pattern Recognition*, 100:107114, 2020.

[115] Chandrasekhar Bhagavatula, Chenchen Zhu, Khoa Luu, and Marios Savvides. Faster than real-time facial alignment: A 3d spatial transformer network approach in unconstrained poses. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3980–3989, 2017.

[116] Amin Jourabloo, Mao Ye, Xiaoming Liu, and Liu Ren. Pose-invariant face alignment with a single cnn. In *Proceedings of the IEEE International Conference on computer vision*, pages 3200–3209, 2017.

[117] Shengtao Xiao, Jiashi Feng, Luoqi Liu, Xuecheng Nie, Wei Wang, Shuicheng Yan, and Ashraf Kassim. Recurrent 3d-2d dual learning for large-pose facial landmark detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1633–1642, 2017.

[118] Aaron S Jackson, Adrian Bulat, Vasileios Argyriou, and Georgios Tzimiropoulos. Large pose 3d face reconstruction from a single image via direct volumetric cnn regression. In *Proceedings of the IEEE international conference on computer vision*, pages 1031–1039, 2017.

[119] Huawei Wei, Shuang Liang, and Yichen Wei. 3d dense face alignment via graph convolution networks. *arXiv preprint arXiv:1904.05562*, 2019.

[120] Jianzhu Guo, Xiangyu Zhu, Yang Yang, Fan Yang, Zhen Lei, and Stan Z Li. Towards fast, accurate and stable 3d dense face alignment. *arXiv preprint arXiv:2009.09960*, 2020.

[121] Xiaoguang Tu, Jian Zhao, Mei Xie, Zihang Jiang, Akshaya Balamurugan, Yao Luo, Yang Zhao, Lingxiao He, Zheng Ma, and Jiashi Feng. 3d face reconstruction from a single image assisted by 2d face images in the wild. *IEEE Transactions on Multimedia*, 2020.

[122] Elad Richardson, Matan Sela, and Ron Kimmel. 3d face reconstruction by learning from synthetic data. In *2016 fourth international conference on 3D vision (3DV)*, pages 460–469. IEEE, 2016.

[123] Ayush Tewari, Michael Zollhofer, Hyeongwoo Kim, Pablo Garrido, Florian Bernard, Patrick Perez, and Christian Theobalt. Mofa: Model-based deep convolutional face autoencoder for unsupervised monocular reconstruction. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 1274–1283, 2017.

[124] Ayush Tewari, Michael Zollhöfer, Pablo Garrido, Florian Bernard, Hyeongwoo Kim, Patrick Pérez, and Christian Theobalt. Self-supervised multi-level face model learning for monocular reconstruction at over 250 hz. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2549–2559, 2018.

[125] Ayush Tewari, Michael Zollhoefer, Florian Bernard, Pablo Garrido, Hyeongwoo Kim, Patrick Perez, and Christian Theobalt. High-fidelity monocular face reconstruction based on an unsupervised model-based face autoencoder. *IEEE transactions on pattern analysis and machine intelligence*, 42(2):357–370, 2018.

[126] Fanzi Wu, Linchao Bao, Yajing Chen, Yonggen Ling, Yibing Song, Songnan Li, King Ngi Ngan, and Wei Liu. Mvf-net: Multi-view 3d face morphable model regression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 959–968, 2019.

[127] Matan Sela, Elad Richardson, and Ron Kimmel. Unrestricted facial geometry reconstruction using image-to-image translation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1576–1585, 2017.

[128] Tatsuro Koizumi and William AP Smith. "look ma, no landmarks!" –unsupervised, model-based dense face alignment. In *European Conference on Computer Vision*, pages 690–706. Springer, 2020.

[129] Zhixin Shu, Duygu Ceylan, Kalyan Sunkavalli, Eli Shechtman, Sunil Hadap, and Dimitris Samaras. Learning monocular face reconstruction using multi-view supervision. In *2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020)(FG)*, pages 762–769. IEEE Computer Society, 2020.

[130] Riza Alp Guler, George Trigeorgis, Epameinondas Antonakos, Patrick Snape, Stefanos Zafeiriou, and Iasonas Kokkinos. Densereg: Fully convolutional dense shape regression in-the-wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6799–6808, 2017.

[131] Hongwei Yi, Chen Li, Qiong Cao, Xiaoyong Shen, Sheng Li, Guoping Wang, and Yu-Wing Tai. Mmface: A multi-metric regression network for unconstrained face reconstruction. In

Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 7663–7672, 2019.

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

[133] Singular Inversions. Facegen modeller [computer software]. *Toronto, Ontario, Canada*, 2016.

[134] Koyo Nakamura and Katsumi Watanabe. Data-driven mathematical model of east-asian facial attractiveness: the relative contributions of shape and reflectance to attractiveness judgements. *Royal Society open science*, 6(5):182189, 2019.

[135] Nikolaas N Oosterhof and Alexander Todorov. The functional basis of face evaluation. *Proceedings of the National Academy of Sciences*, 105(32):11087–11092, 2008.

[136] Clare AM Sutherland, Gillian Rhodes, and Andrew W Young. Facial image manipulation: A tool for investigating social perception. *Social Psychological and Personality Science*, 8(5):538–551, 2017.

[137] Alexander Todorov and Nikolaas N Oosterhof. Modeling social perception of faces [social sciences]. *IEEE Signal Processing Magazine*, 28(2):117–122, 2011.

[138] Laszlo A Jeni, Hideki Hashimoto, and Andras Lorincz. Efficient, pose invariant facial emotion classification using constrained local model and 2d shape information. In *Extended Abstracts, Workshop on Gesture Recognition at CVPR*, volume 2011, 2011.

[139] Brahim Aksasse, Hamid Ouanan, and Mohammed Ouanan. Novel approach to pose invariant face recognition. *Procedia Computer Science*, 110:434–439, 2017.

[140] James Booth, Anastasios Roussos, Stefanos Zafeiriou, Allan Ponniah, and David Dunaway. A 3d morphable model learnt from 10,000 faces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5543–5552, 2016.

[141] Pascal Paysan, Reinhard Knothe, Brian Amberg, Sami Romdhani, and Thomas Vetter. A 3d face model for pose and illumination invariant face recognition. In *2009 sixth IEEE international conference on advanced video and signal based surveillance*, pages 296–301. Ieee, 2009.

[142] Chen Cao, Yanlin Weng, Shun Zhou, Yiying Tong, and Kun Zhou. Facewarehouse: A 3d facial expression database for visual computing. *IEEE Transactions on Visualization and Computer Graphics*, 20(3):413–425, 2013.

[143] Alan Brunton, Timo Bolkart, and Stefanie Wuhrer. Multilinear wavelets: A statistical shape space for human faces. In *European Conference on Computer Vision*, pages 297–312. Springer, 2014.

[144] Patrik Huber, Guosheng Hu, Rafael Tena, Pouria Mortazavian, P Koppen, William J Christmas, Matthias Ratsch, and Josef Kittler. A multiresolution 3d morphable face model and fitting framework. In *Proceedings of the 11th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, 2016.

[145] James Booth, Anastasios Roussos, Allan Ponniah, David Dunaway, and Stefanos Zafeiriou. Large scale 3d morphable models. *International Journal of Computer Vision*, 126(2):233–254, 2018.

[146] Thomas Gerig, Andreas Morel-Forster, Clemens Blumer, Bernhard Egger, Marcel Luthi, Sandro Schönborn, and Thomas Vetter. Morphable face models-an open framework. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pages 75–82. IEEE, 2018.

[147] Hang Dai, Nick Pears, William AP Smith, and Christian Duncan. A 3d morphable model of craniofacial shape and texture variation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3085–3093, 2017.

[148] Hang Dai, Nick Pears, William Smith, and Christian Duncan. Statistical modeling of craniofacial shape and texture. *International Journal of Computer Vision*, 128(2):547–571, 2020.

[149] Brian Amberg, Sami Romdhani, and Thomas Vetter. Optimal step nonrigid icp algorithms for surface registration. In *2007 IEEE conference on computer vision and pattern recognition*, pages 1–8. IEEE, 2007.

[150] Andriy Myronenko and Xubo Song. Point set registration: Coherent point drift. *IEEE transactions on pattern analysis and machine intelligence*, 32(12):2262–2275, 2010.

[151] Xiangyu Zhu, Zhen Lei, Junjie Yan, Dong Yi, and Stan Z Li. High-fidelity pose and expression normalization for face recognition in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 787–796, 2015.

[152] Marcel Piotraschke and Volker Blanz. Automated 3d face reconstruction from multiple images using quality measures. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3418–3427, 2016.

[153] James Booth, Epameinondas Antonakos, Stylianos Ploumpis, George Trigeorgis, Yannis Panagakis, and Stefanos Zafeiriou. 3d face morphable models" in-the-wild". In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 48–57, 2017.

[154] James Booth, Anastasios Roussos, Evangelos Ververas, Epameinondas Antonakos, Stylianos Ploumpis, Yannis Panagakis, and Stefanos Zafeiriou. 3d reconstruction of "in-the-wild" faces in images and videos. *IEEE transactions on pattern analysis and machine intelligence*, 40(11):2638–2652, 2018.

[155] Peng Liu, Yao Yu, Yu Zhou, and Sidan Du. Single view 3d face reconstruction with landmark updating. In *2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 403–408. IEEE, 2019.

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

[157] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. *arXiv preprint arXiv:1606.03657*, 2016.

[158] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR, 2017.

[159] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved training of wasserstein gans. *arXiv preprint arXiv:1704.00028*, 2017.

[160] Jon Gauthier. Conditional generative adversarial nets for convolutional face generation. *Class Project for Stanford CS231N: Convolutional Neural Networks for Visual Recognition, Winter semester*, 2014(5):2, 2014.

[161] Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Joshua Susskind, Wenda Wang, and Russell Webb. Learning from simulated and unsupervised images through adversarial training. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2107–2116, 2017.

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

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

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

[165] Junho Kim, Minjae Kim, Hyeonwoo Kang, and Kwanghee Lee. U-gat-it: Unsupervised generative attentional networks with adaptive layer-instance normalization for image-to-image translation. *arXiv preprint arXiv:1907.10830*, 2019.

[166] Jinjin Gu, Yujun Shen, and Bolei Zhou. Image processing using multi-code gan prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3012–3021, 2020.

[167] Jie Gui, Zhenan Sun, Yonggang Wen, Dacheng Tao, and Jieping Ye. A review on generative adversarial networks: Algorithms, theory, and applications. *arXiv preprint arXiv:2001.06937*, 2020.

[168] Weihao Xia, Yulun Zhang, Yujiu Yang, Jing-Hao Xue, Bolei Zhou, and Ming-Hsuan Yang. Gan inversion: A survey. *arXiv preprint arXiv:2101.05278*, 2021.

[169] Heng Liu, Xiaoyu Zheng, Jungong Han, Yuezhong Chu, and Tao Tao. Survey on gan-based face hallucination with its model development. *IET Image Processing*, 13(14):2662–2672, 2019.

[170] Andrew Brock, Theodore Lim, James M Ritchie, and Nick Weston. Neural photo editing with introspective adversarial networks. *arXiv preprint arXiv:1609.07093*, 2016.

[171] David Berthelot, Thomas Schumm, and Luke Metz. Began: Boundary equilibrium generative adversarial networks. *arXiv preprint arXiv:1703.10717*, 2017.

[172] Luan Tran, Xi Yin, and Xiaoming Liu. Disentangled representation learning gan for pose-invariant face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1415–1424, 2017.

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

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

[175] Rui Huang, Shu Zhang, Tianyu Li, and Ran He. Beyond face rotation: Global and local perception gan for photorealistic and identity preserving frontal view synthesis. In *Proceedings of the IEEE international conference on computer vision*, pages 2439–2448, 2017.

[176] Jian Zhao, Lin Xiong, Jayashree Karlekar, Jianshu Li, Fang Zhao, Zhecan Wang, Sugiri Pranata, Shengmei Shen, Shuicheng Yan, and Jiashi Feng. Dual-agent gans for photorealistic and identity preserving profile face synthesis. In *NIPS*, volume 2, page 3, 2017.

[177] Yujun Shen, Ping Luo, Junjie Yan, Xiaogang Wang, and Xiaoou Tang. Faceid-gan: Learning a symmetry three-player gan for identity-preserving face synthesis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 821–830, 2018.

[178] Yu Chen, Ying Tai, Xiaoming Liu, Chunhua Shen, and Jian Yang. Fsrnet: End-to-end learning face super-resolution with facial priors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2492–2501, 2018.

[179] Adrian Bulat and Georgios Tzimiropoulos. Super-fan: Integrated facial landmark localization and super-resolution of real-world low resolution faces in arbitrary poses with gans. In *Pro-*

ceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 109–117, 2018.

[180] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2019.

[181] Tamar Rott Shaham, Tali Dekel, and Tomer Michaeli. Singan: Learning a generative model from a single natural image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4570–4580, 2019.

[182] Ran Yi, Yong-Jin Liu, Yu-Kun Lai, and Paul L Rosin. Apdrawinggan: Generating artistic portrait drawings from face photos with hierarchical gans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10743–10752, 2019.

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

[184] Wenyan Wu and Shuo Yang. Leveraging intra and inter-dataset variations for robust face alignment. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 150–159, 2017.

[185] Sina Honari, Pavlo Molchanov, Stephen Tyree, Pascal Vincent, Christopher Pal, and Jan Kautz. Improving landmark localization with semi-supervised learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1546–1555, 2018.

[186] Zhiqiang Tang, Xi Peng, Shijie Geng, Lingfei Wu, Shaoting Zhang, and Dimitris Metaxas. Quantized densely connected u-nets for efficient landmark localization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 339–354, 2018.

[187] Yu Chen, Chunhua Shen, Xiu-Shen Wei, Lingqiao Liu, and Jian Yang. Adversarial posenet: A structure-aware convolutional network for human pose estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1212–1221, 2017.

[188] Grigorios G Chrysos, Epameinondas Antonakos, Patrick Snape, Akshay Asthana, and Ste-

fanos Zafeiriou. A comprehensive performance evaluation of deformable face tracking "in-the-wild" . *International Journal of Computer Vision*, 126(2):198–232, 2018.

[189] Heng Yang, Xuhui Jia, Chen Change Loy, and Peter Robinson. An empirical study of recent face alignment methods. *arXiv preprint arXiv:1511.05049*, 2015.

[190] Marek Kowalski, Jacek Naruniec, and Tomasz Trzcinski. Deep alignment network: A convolutional neural network for robust face alignment. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 88–97, 2017.

[191] Xuehan Xiong and Fernando De la Torre. Supervised descent method and its applications to face alignment. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 532–539, 2013.

[192] Junfeng Zhang and Haifeng Hu. Exemplar-based cascaded stacked auto-encoder networks for robust face alignment. *Computer Vision and Image Understanding*, 171:95–103, 2018.

[193] Hadi Kazemi, Fariborz Taherkhani, and Nasser M Nasrabadi. Unsupervised facial geometry learning for sketch to photo synthesis. In *2018 international conference of the biometrics special interest group (BIOSIG)*, pages 1–5. IEEE, 2018.

[194] Chuan Li and Michael Wand. Precomputed real-time texture synthesis with markovian generative adversarial networks. In *European conference on computer vision*, pages 702–716. Springer, 2016.

[195] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016.

[196] Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2794–2802, 2017.

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

[198] Roberto Valle, José M Buenaposada, and Luis Baumela. Cascade of encoder-decoder cnns with learned coordinates regressor for robust facial landmarks detection. *Pattern Recognition Letters*, 136:326–332, 2020.

[199] Dong Chen, Shaoqing Ren, Yichen Wei, Xudong Cao, and Jian Sun. Joint cascade face detection and alignment. In *European conference on computer vision*, pages 109–122. Springer, 2014.

[200] Rafal Pilarczyk and Władysław Skarbek. Tuning deep learning algorithms for face alignment and pose estimation. In *Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2018*, volume 10808, page 108081A. International Society for Optics and Photonics, 2018.

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

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

[203] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016.

[204] Sheng Chen, Yang Liu, Xiang Gao, and Zhen Han. Mobilefacenets: Efficient cnns for accurate real-time face verification on mobile devices. In *Chinese Conference on Biometric Recognition*, pages 428–438. Springer, 2018.

[205] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

[206] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1501–1510, 2017.

[207] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[208] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6924–6932, 2017.

[209] Shizhan Zhu, Cheng Li, Chen Change Loy, and Xiaoou Tang. Face alignment by coarse-to-fine shape searching. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4998–5006, 2015.

[210] Sami Romdhani and Thomas Vetter. Estimating 3d shape and texture using pixel intensity, edges, specular highlights, texture constraints and a prior. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 986–993. IEEE, 2005.

[211] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1314–1324, 2019.

[212] Xiang Li, Wenhai Wang, Xiaolin Hu, and Jian Yang. Selective kernel networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 510–519, 2019.

[213] Qibin Hou, Daquan Zhou, and Jiashi Feng. Coordinate attention for efficient mobile network design. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13713–13722, 2021.

[214] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[215] Andrew Howard, Andrey Zhmoginov, Liang-Chieh Chen, Mark Sandler, and Menglong Zhu. Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation. 2018.

[216] Kai Han, Yunhe Wang, Qi Tian, Jianyuan Guo, Chunjing Xu, and Chang Xu. Ghostnet: More features from cheap operations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1580–1589, 2020.

[217] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258, 2017.

[218] Lei Jiang, Xiao-Jun Wu, and Josef Kittler. Dual attention mobdensenet (damdnet) for robust 3d face alignment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[219] Xu Luo, Pengfei Li, Fuxuan Chen, and Qijun Zhao. Improving large pose face alignment by regressing 2d and 3d landmarks simultaneously and visibility refinement. In *Chinese Conference on Biometric Recognition*, pages 349–357. Springer, 2018.

[220] Adam Kortylewski, Bernhard Egger, Andreas Schneider, Thomas Gerig, Andreas Morel-Forster, and Thomas Vetter. Analyzing and reducing the damage of dataset bias to face recognition with synthetic data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[221] Bernhard Egger, Sandro Schönborn, Andreas Schneider, Adam Kortylewski, Andreas Morel-Forster, Clemens Blumer, and Thomas Vetter. Occlusion-aware 3d morphable models and an illumination prior for face image analysis. *International Journal of Computer Vision*, 126(12):1269–1287, 2018.

[222] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3606–3613, 2014.

[223] Ralph Gross, Iain Matthews, Jeffrey Cohn, Takeo Kanade, and Simon Baker. Multi-pie. *Image and vision computing*, 28(5):807–813, 2010.

[224] Ronen Basri and David W Jacobs. Lambertian reflectance and linear subspaces. *IEEE transactions on pattern analysis and machine intelligence*, 25(2):218–233, 2003.

Chapter A

Appendix

A.1 Appendix 1

This section introduce the different discrimitor network based on CycleGAN model (included Pixel GAN, PatchGAN2 and ImageGAN architecture).

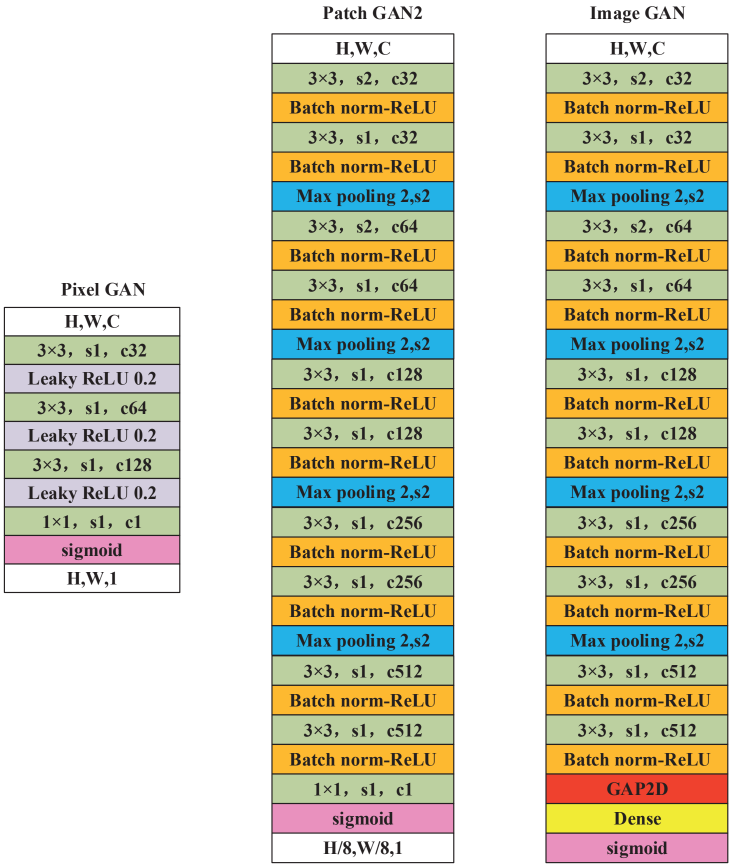


Figure A.1: The different Discrimitor structure.

A.2 Appendix 2

This section introduce the network detail based on UGATIT model (included generator, local and global discrimitor architecture).

Table A.1: The detail of generator architecture

Model part	Input → Output Shape	Detail information
Conv Layer	(h,w,3)→ (h,w,64)	CONV-(N64,K7,S1,P3),IN,ReLU
Hourglass	(h,w,64)→ (h,w,64)	Hourglass block1
	(h,w,64) →(h,w,64)	Hourglass block2
Down Block	(h,w,64)→ (h/2,w/2,128)	CONV-(N128,K3,S2,P1),IN,ReLU
	(h/2,w/2,128–(h/4,w/4,256)	CONV-(N256,K3,S2,P1),IN,ReLU
Encoder Block	(h/4,w/4,256)→ (h/4,w/4,256)	ResBlock-(N256,K3,S1,P1),IN,ReLU
	(h/4,w/4,256)→ (h/4,w/4,256)	ResBlock-(N256,K3,S1,P1),IN,ReLU
	(h/4,w/4,256)→ (h/4,w/4,256)	ResBlock-(N256,K3,S1,P1),IN,ReLU
	(h/4,w/4,256)→ (h/4,w/4,256)	ResBlock-(N256,K3,S1,P1),IN,ReLU
CAM	(h/4,w/4,256)→ (h/4,w/4,512)	Global Average/Max Pooling MLP-(N1)
	(h/4,w/4,512)→ (h/4,w/4,256)	CONV-(N256,K1,S1),ReLU
Decoder Block	(h/4,w/4,256)→ (h/4,w/4,256)	ResnetSoftAdaLINBlock-(N256,K3,S1,P1),SoftAdaLIN,ReLU
	(h/4,w/4,256)→ (h/4,w/4,256)	ResnetSoftAdaLINBlock-(N256,K3,S1,P1),SoftAdaLIN,ReLU
	(h/4,w/4,256)→ (h/4,w/4,256)	ResnetSoftAdaLINBlock-(N256,K3,S1,P1),SoftAdaLIN,ReLU
	(h/4,w/4,256)→ (h/4,w/4,256)	ResnetSoftAdaLINBlock-(N256,K3,S1,P1),SoftAdaLIN,ReLU
UP Block	(h/4,w/4,256)→ (h/2,w/2,128)	Up-CONV-(N128,K3,S1,P1),LIN,ReLU
	(h/2,w/2,128)→ (h,w,64)	Up-CONV-(N128,K3,S1,P1),LIN,ReLU
Conv Layer	(h,w,64)→ (h,w,3)	CONV-(N3,K7,S1,P3),Tanh

Table A.2: The detail of local discriminator architecture

Model part	Input → Output Shape	Detail information
Down Block	(h,w,3)→ (h/2,w/2,64)	CONV-(N64,K4,S2,P1),SN,Leaky-ReLU
	(h/2,w/2,64)→ (h/4,w/4,128)	CONV-(N128,K4,S2,P1),SN,Leaky-ReLU
	(h/4,w/4,128)→ (h/8,w/8,256)	CONV-(N1256,K4,S2,P1),SN,Leaky-ReLU
	(h/8,w/8,256)→ (h/8,w/8,512)	CONV-(N512,K4,S1,P1),SN,Leaky-ReLU
CAM	(h/8,w/8,512)→ (h/8,w/8,1024)	Global Average/Max Pooling MLP-(N1)
	(h/4,w/4,1024)→ (h/8,w/8,512)	CONV-(N512,K1,S1),Leaky-ReLU
Classifier	(h/8,w/8,512)→ (h/8,w/8,1)	CONV-(N1,K4,S1,P1),SN

Table A.3: The detail of global discriminator architecture

Model part	Input → Output Shape	Detail information
Down Block	(h,w,3)-(h/2,w/2,64)	CONV-(N64,K4,S2,P1),SN,Leaky-ReLU
	(h/2,w/2,64)-(h/4,w/4,128)	CONV-(N128,K4,S2,P1),SN,Leaky-ReLU
	(h/4,w/4,128)-(h/8,w/8,256)	CONV-(N1256,K4,S2,P1),SN,Leaky-ReLU
	(h/8,w/8,256)-(h/16,w/16,512)	CONV-(N512,K4,S2,P1),SN,Leaky-ReLU
	(h/16,w/16,512)-(h/32,w/32,1024)	CONV-(N1024,K4,S2,P1),SN,Leaky-ReLU
	(h/32,w/32,1024)-(h/32,w/32,2048)	CONV-(N2048,K4,S1,P1),SN,Leaky-ReLU
CAM	(h/32,w/32,2048)-(h/32,w/32/4096)	Global Average/Max Pooling MLP-(N1)
	(h/32,w/32,4096)-(h/32,w/32,2048)	CONV-(N2048,K1,S1),Leaky-ReLU
Classifier	(h/32,w/32,2048)-(h/32,w/32,1)	CONV-(N1,K4,S1,P1),SN

A.3 Appendix 3

This section introduced the website sources for some of the images cited in this paper.

a. https://auto.ifeng.com/quanmeiti/20190103/1256536.shtml

b. https://chejiahao.autohome.com.cn/info/3527000/

c. https://www.deliyun.com/newsinfo/2365504.html

d. https://zhuanlan.zhihu.com/p/111047818

e. http://m.facegl.com/detail/html/1822.html

f. https://finance.eastmoney.com/a/202003011402122866.html

g. https://market.aliyun.com/products/57124001/cmapi031848.html

h. https://www.sohu.com/a/396458856_463965

i. https://moore.live/news/23348/detail/

j. http://www.jvttech.com/Mobile/Show/index/cid/73/id/78.html

k. https://www.sohu.com/a/145679340_116132

l. https://zhuanlan.zhihu.com/p/180271463

m. http://www.cs.wisc.edu/˜lizhang/projects/face-parsing/

n. https://github.com/minivision-ai/photo2cartoon

o. https://zhuanlan.zhihu.com/p/270958248