Research on 3D Reconstruction Based on 2D Face Images

Faculty of Media and Communication

Master of Research degree Thesis

MRes Student: **Yu ZHOU**

Supervisor: **HongChuan YU**

May 2021

Abstract

3D face reconstruction is a popular research area in the field of computer vision and has a wide range of applications in various fields such as animation design, virtual reality, medical guidelines, and face recognition. Current commercial 3D face reconstruction generally relies on large image scanning equipment to fuse multiple images through sensors for 3D face reconstruction. However, this approach requires manual modelling, which is costly in terms of time and money, and expensive in terms of equipment, making it unpopular in practical applications. Compared to 3D face construction with multiple images, the single-image approach reduces computational time and economic costs, is relatively simple to implement and does not require specific Hardware equipment. Therefore, we focus on single-image approach in this dissertation and contribute in terms of research novelty and practical use. The main work is as follows：

A unique pre-processing process is designed to separate face alignment from face reconstruction. In this dissertation, the Active Shape Model (ASM) algorithm is used for face alignment to detect the face feature points in the image. The face data is posing corrected so that the corrected face is better adapted to the face pose of the UV-Position Map. The UV coordinates are then used to map the 3D information onto the 2D image, creating a UV-3D mapping map. In order to enhance the effect, this dissertation also does face cropping to fill the whole space as much as possible with face data and expands the face dataset using rotation, scaling, panning and noise addition.

Improving the neural network model by using the idea of residual learning to train the network model incrementally, emphasizing the reconstruction of the model for deep information. Face data characteristics are first extracted using the encoding and decoding layers, and then face features are learned using the residual learning layer. By comparing with the previous algorithm, we achieved a considerable lead on

the 300W-LP face dataset, with a 35% reduction in NME error accumulation over the RPN algorithm.

Based on the pre-processing methods and residual structures we proposed, the experimental results have shown good performance on 3D reconstruction of faces. The end-to-end approach based on deep learning achieves better reconstruction quality and accuracy compared to traditional, model-based face reconstruction methods.

Key words: 3DMM model; deep learning; face reconstruction; residual learning; CNN

1. Introduction

1.1 Background and significance of the study

Compared to 2D faces, 3D faces have an intrinsic invariance of face pose and expression, i.e., the shape and features of each person's face are approximately the same in distribution, but there are many differences in the details of the features and face texture, which constitute the recognizable features of each face. The construction of highly accurate and realistic 3D face models is also a very challenging task in the field of computer graphics. 3D face reconstruction has practical value and significance, for example in face recognition, security applications, facial expression analysis, 3D gaming, criminal investigation, biomedical and cosmetic scenarios.

In the field of film and television, on 22 February 2019, the American science fiction *Alita: Battle Angel* was released, constructing the lead character Alita's face shape through face 3D reconstruction technology, undergoing over 5,000 iterations of reconstruction, with 200 designs produced for the fusion of various parts of the human face, together with the most advanced lighting engine, human physical action engine, and skin texture rendering engine, through professional equipment to Capture the facial features of the actors behind the scenes, then migrate the actors' features into the Alita character model, and finally use this to drive the Alita character's face transformation movements. The resurrection of Paul Walker in *Fast and Furious 7* uses several face reconstruction algorithms, and the character Rocket in the *Guardians of the Galaxy* series uses graphic 3D reconstruction techniques. In the field of medical aesthetics, 3D reconstruction can also be used to model patients in 3D, which can be used to visually demonstrate the effects of plastic surgery and to develop more appropriate aesthetic solutions. 3D face modelling is also widely used in the gaming and entertainment field, where 3D games attract many gamers with their realistic visual effects and stunning gaming experience. For example, games such as "NBA2K" and "Live Soccer", in which 3D face reconstruction technology is used to turn real-world people into realistic virtual 3D characters, greatly increasing

the player's experience.

3D reconstruction is an important branch of computer graphics, and 3D reconstruction of the human face is the focus and bottleneck in 3D reconstruction. There are currently many solutions to complete 3D face reconstruction, mainly through professional 3D face modelling software, such as Maya, 3DMax, etc... Or through equipment such as structured light scanners to build 3D face models. These methods have high cost of equipment, high complexity of operation and limitation on the use of scenarios, are complex to operate and have limitations on the use of scenarios. Face 3D reconstruction based on a single RGB image is more scientifically valuable and technically challenging due to the limited availability of 3D face data. It uses basic information such as color, texture and illumination in 2D images to reconstruct the spatial information of the face shape and surface in 3D space in the world coordinate system, and its essence lies in digitizing the real face model in reality. The method of recovering the 3D face structure through a single face photograph requires minimal input information and minimal hardware equipment, and the whole process is relatively researchable and challenging, so this research focuses on a single image based 3D model reconstruction method of the face.

1.2 Current status of research

Although 3D face reconstruction has been widely integrated into various industries, the current face 3D reconstruction technology is still in its infancy and has limitations of application and other shortcomings. At present, 3D face reconstruction is mainly divided into traditional 3D face reconstruction methods, 3D face reconstruction methods based on variable models and end-to-end 3D face reconstruction methods based on machine learning, As shown in Figure 1.1

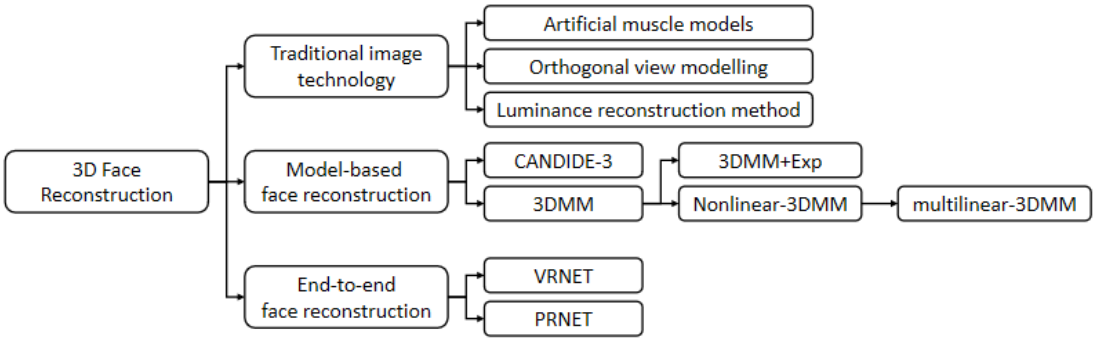

Figure1.1 Classification of 3D face reconstruction algorithms

Traditional methods are modelled by image information, for example, 3D face reconstruction based on one or more kinds of information such as image brightness, edge information, linear perspective, color, relative height, parallax. Most of these methods are based on image information, which can only do simple abstract expressions and cannot obtain accurate and precise face models, and are not expressive enough; and then, face reconstruction based on statistical deformable models was developed, which establishes a face database through accurate 3D scanning tools, carries out statistical modelling based on statistical theory, and then expresses the face model with triangular grid and point cloud set. The mainstream variable models are the CANDIDE-3 (Ahlberg, J. et al. 2001) and 3DMM (Blanz, V. et al. 1999) models, and many excellent 3D face reconstruction algorithms have been developed based on their ideas, including both traditional approaches [23] and machine learning methods. The main bottleneck of such methods is that they rely heavily on statistical databases and have a small range of applicability. With the rise of deep learning in recent years, a major breakthrough has been made in 3D face reconstruction with deep learning, and many end-to-end 3D face reconstruction methods have been proposed, which do not require a cumbersome face model similar to 3DMM, but only need to build their own 3D face representation, bypassing the face model and using a deep learning network structure for direct regression, end-to-end reconstruction of 3D faces. With the popularity of smart devices such as mobile phones, the threshold for acquiring RGB images has become lower, so 3D reconstruction of faces based on single color photographs is of great scientific and

practical value, however, single images, which do not have depth information, and RGB information are also susceptible to large poses, occlusions, lighting, etc., pose many challenges and problems for 3D reconstruction of faces.

1.2.1 Traditional image technology

In the traditional domain, developers usually use image information to obtain generic models of faces, such as image luminance, edge information, color, relative height, linear perspective, parallax, etc. One of the more common algorithms is the modified shadow recovery shape method (SFS) (Bruckstein, A.M. 1988) proposed by Hansen et al. which uses SFS to estimate illumination and reflection parameters and use the lighting parameters to smooth the face mesh model. The core principle is to exploit the luminance information to correspond to the grey scale values of the face greyscale images to obtain the normal vectors of the pixels in the figure in 3D space, pixel by pixel. The depth information is then obtained from the normal vector using Lambert's cosine theorem and Lambert's reflector surface. The system for generating the luminance information of a greyscale map generally consists of four determinants:

（1）**Camera model**: consisting mainly of camera parameters to obtain projected images.

（2）**Illumination model**: contains mainly the position of light, the direction of illumination and the energy equation of light.

（3）**Material surface reflectance**: in SFS, called the light-free color information, which determines the rate of light is reflected on the surface of an object and is a physical property of the object material itself.

（4）**Geometric construction of the material surface**: this is applied in the interpolation of the graphic rendering.

With the four factors above, it is possible to perform lighting simulation in the rendering pipeline. Given a 3D model in a rendered scene, a vertex shader and a pixel shader are calculated to obtain a 2D image with lighting information, and the SFS algorithm is just the reverse of this rendering process: given a 2D grayscale image, its

lighting model is imitated and the material reflectance of the model and then we can obtain the 3D depth information of object. However, this traditional method is only capable to simple 3D geometric reconstruction of the face, relies on the lighting model, which can lead to significant errors when the lighting simulation is not accurate, and the algorithm generally requires a frontal image of the human face, which is difficult to obtain directly for large posed, obscured face images.

Kemelmacher et al. used face image shadow changes to estimate the illumination and reflection parameters of the face mesh model and reconstructed a realistic face mesh model. There are some studies using laser scanning techniques for 3D reconstruction, where a 3D laser scanner emits a laser to illuminate a target object, and then a digital camera captures the curved light along the contours of the object, records the relationship between the curve and the original straight laser, and finally obtains the 3D contour information of the target object according to the principles of triangulation. Lee et al. of the University of Toronto first proposed the construction of 3D face models with the aid of laser scanning in 1995. The 3D face data obtained by the scanner is pre-processed by means of the Laplace transform, and then the face feature points are extracted in the Laplace space, and the template 3D face model is adjusted to obtain a more accurate 3D face model of the target. As 3D laser scanners are expensive and have high requirements for the data acquisition environment, structured light scanners have been developed and used. Beumier et al. proposed a structured light-based acquisition system suitable for 3D faces. Meanwhile, the design balanced the three factors of acquisition speed, cost, and resolution. This system also constructed a 3D face dataset containing more than 100 people with guaranteed correct pose and was able to capture the entire face data in half a second using a single shot.

1.2.2 Face Reconstruction based on Parametric Models

（1）Candide-3 Model

The Candide model, proposed by Mikael Rydfalk at Linköping University in

1987, is a parametric face model based on facial action coding, controlled by global and local Action Units (AUs), and meets the requirements of rapid face reconstruction. The latest Candide-3 model has 113 vertices and 168 faces. This method abstracts the face into a simple mathematical geometric model. Once the complex face structure is parameterized, the deformable parameters of the generic model can be modified by algorithms such as progressive iteration or rigid matching, so that the features represented by the model match the input image that needs to be reconstructed, and the model generally contains both global and local parameters. The global parameters are adjusted for the overall contour of the model or the size of the model, and through specific dependencies such as feature point correspondence (typically 58 feature points), the deformable model is generally aligned with the relative positions of the five senses in the input image. The local parameter adjustment is to optimise the local details, especially that in the organs such as the eyes, nose and mouth, so that the local details fit the input data more accurately. The advantage of the CANDIDE-3 model is that the overall point-line data is small, the computational effort is small, and the model can be quickly converged to a corresponding face model in a few iterations, which is why it is often used in security applications where details are not required. For these applications, high resolution face models are often not required, and recognition can be generated simply by the geometric structure, which then generates a cipher encryption from geometric information, such as vertex density, the angle of certain lines and faces, etc. Therefore, its drawbacks are also obvious. It is difficult to simulate a complex system of faces with more than 110 vertices, which predestines the algorithm to be applied in scenarios that require rich texture information, and with a larger base, more information overlap will be generated, so the model is intelligently applied in scenarios that do not require high face accuracy, such as face encryption. Although the number of points and faces in the Candide-3 model is not enough to obtain highly accurate results, it also has the advantages of fast computation rate, easy expansion and friendly coding for 3D expression animation.

（2）3DMM Model

In 1999, V. Blanz and T. Vetter from Switzerland in Europe proposed an automated face modelling process, which is the classical 3DMM (3D Morphable Model) morphological model. The dissertation presents a complete automated reconstruction of a variable face model of the human face for the first time. The main key point of the morphable technique is that if there is a correspondence between the representational properties of two images, then a smooth linear model can be abstracted by this property, and other images in between can be obtained by parameter adjustment. 3DMM model is based on the PCA (Wold, S. et al. 1987) statistical idea, by feature abstraction of a certain 3D face database, divided into two linear expressions as face SHAPE and face TEXTURE, also can add lighting, expression, pose and other linear models, the abstraction of a high-precision three-dimensional face model. 3DMM is a single linear face variable model.

The 3DMM model database is a linear combination of deformable vectors in the face model. On the basis of the standard 3D face representation, it is assumed that the 3D face model is generated by a mixture of face models (m)that represent certain properties, where each face model m contains the corresponding Shape vector Si and Texture vector Ti, (Si and Ti are generally generated through PCA statistical abstraction) so that the final 3D face model can be expressed by the following equations 1.1 and 1.2.

$$S_{newModel} = \bar{S} + \sum_{i=1}^{m-1} \alpha_i s_i \qquad \textbf{Equation 1.1}$$

$$T_{newModel} = \bar{T} + \sum_{i=1}^{m-1} \beta_i t_i \qquad \textbf{Equation 1.2}$$

Where S and T in the formula denote the standard face model and average face shape model information, α_i $\beta_i$correspond to the variable coefficients of Shape and Texture respectively, so the face reconstruction problem turns into a problem of finding the α，β coefficients, and by inputting a face image, the two coefficients can be adjusted iteratively to obtain the model with the smallest difference.

The algorithm is less complex, but lacks computation for the representation of

facial expressions, so there has been a large number of related works making improvements to the 3DMM method. Amin Jourabloo et al. published a paper in 2016 (Jourabloo, A. et al. 2016), which proposed using a cascaded convolutional neural network to estimate the parameters of the 3DMM and complete the prediction of the 68 facial key points. This method can aid the prediction of 3DMM parameters. And the paper proposed two types of pose invariant features to enhance face alignment. In 2018, Kyle Genova et al. of Princeton University published a paper (Genova, K. et al. 2018) using unsupervised learning methods to obtain the parameters of 3DMM.

Models like 3DMM are generally based on an average face data and also store information about deviations from the average model. For example, in most cases, people with long faces will have longer noses. Based on the relevance of these characteristics, it is possible to generate a facial model of a particular face during the reconstruction process, even if the full representational information of the face is not available, as long as the deviations of the face to be reconstructed from the average face are several hundred mathematical parameters. Of course, this deviation information also includes age differences, gender differences and face aspect ratios. But storing all the deviation coefficients of all face information from the average face model is obviously very impractical, and the 3DMM model would thus appear bloated and degraded in performance. The current publicly available datasets based on 3DMM models are generally based on scanning a large number of real faces and relying on careful manual labelling of the features that need to be represented. This method not only costs massive human effort, but also reduces the reliability of the model due to artificial errors. The model also has limited ability to imitate faces of different ages and races due to the limited information collected from the crowd.

（3）LSFM Model

The LSTM model (Booth, J. et al. 2018) is also based on the 3DMM variable model idea, and this method abstracts 9663 unique face markers from the human face pictures and draws them automatically. According to the researchers, LSFM is the largest deformable model of the face constructed in the field of face 3D reconstruction,

and it contains statistical information extracted from a very large dataset of face variables. To reduce the manual effort required to build such a model, the researchers have built a novel and fully automated deformable model building pipeline which is stable and reliable and can validate the information obtained from the face model with the best available dense alignment techniques.

The dataset for training LSFM collects all types of mainstream face information representation features and also includes a large amount of face statistics for each subject, allowing not only the training of globally applicable standard 3DMM models, but also the purposeful building of variable models for specific age, gender, or ethnicity. The results from the highly qualitative and quantitative tests presented by the researchers show that this 3DMM variable model achieves the best fit, significantly outperforming existing models. Finally, to benefit research in the field of face reconstruction, the researchers have made the source code of the automated 3DMM construction pipeline publicly available so that global stakeholders can join together to build a global 3DMM model and be able to subdivide the global model into various thematically models tailored to age, gender, and race. Secondly, LSFM also proposes many innovative methods for 3D face reconstruction, such as aligning the 68 feature points (Cootes, T.F. et al. 1995) of the model by directional gradient histogram (Antonakos, E. et al. 2014), which can effectively solve the problems of occlusion and ill pose; aligning with the standard model BFM(Basel Face Model)(Paysan, P. et al. 2009)and proposing a standard mesh model to prevent the differences in model vertex information representation due to too many model standards; mapping sparse alignment points onto the 3D model; fitting a 3D Face that best matches the scanned model by the non-rigid progressive iterative method (NICP) and Procrustes alignment method(Davies, R. et al. 2008) with reference to the BFM model

（4）Nonlinear-3DMM Model

The classical 3DMM learns from a face dataset, which consists of 2D face images and corresponding 3D face scan meshes, typically represented by two sets of

PCA basis functions, Shape and Texture. This results in a model acquisition that is heavily dependent on the type and amount of training data and the linear basis of the representation model, with limited linearity in the deformation capability of the 3DMM. Nonlinear-3DMM (Tran, L. et al. 2018) transforms the linear representation of PCA into a non-linear representation in order to better represent face information, exploiting the non-linearity of the neural network activation function of deep learning. For the data set, the 3D model is projected from the Cartesian coordinate system into the 2D UV coordinate system (Bookstein, F.L. 1989) according to the UV unfolding algorithm, which is then fed into the neural network as the input, and a mesh is generated by a Shape encoder, while a UV is generated by a Texture encoder, and then the mesh and UV are decoded by two corresponding decoders (Tewari, A. et al. 2017). The final output is a layer of rendering, so that both the input and output of the training are images. The authors created a loss function for training evaluation, with a total loss function as shown in Equation 1.3.

$$L = L_{rec} + \lambda_{adv} L_{adv} + \lambda_L L_\lambda \qquad \textbf{Equation 1.3}$$

The first term L_{rec} is the regression network loss, which compares the vector parametrization of the two images, the second part $\lambda_{adv} L_{adv}$ is the adversarial loss (Shrivastava, A. et al. 2017), where the authors use adversarial neural networks to solve the face occlusion problem to some extent; the third part $\lambda_L L_\lambda$ is the 68 geometric feature points corresponding to the matrix parametrization. The non-linear nature of the activation function of the neural network enables it to represent the face model non-linearly (Paysan, P. et al. 2009). However, the training network is more complex, containing not only a regression network and an adversarial neural network, but also a rendering layer that learns the mesh and texture, as well as the rendering parameters, i.e., the orientation of the shape after rotating and scaling and translating the projection to the face of the input image. This not only makes training relatively slow, but also makes it more difficult to train to obtain an accurate model.

（5）Multilinear-Model

The Multilinear-Model model (Abrevaya, V.F. et al. 2018), on the other hand, is based on the SVD decomposition and is therefore expressed as a multi-linear relationship. A 3D face is expressed using a tensor decomposition as in Equation 1.4 below.

$$X = B_{(1)}\left(A^{(2)} \odot A^{(3)} \cdots A^{(M)}\right) = B_{(1)}\left(\bigodot_{M}^{m=2} A^{(m)}\right) \qquad \textbf{Equation 1.4}$$

$B_{(1)}$ in the formula is obtained by learning from data in the face dataset, similar to the average model or standard face model in 3DMM, while $A^{(M)}$ is a series of coefficients that reconstructs X. ⊙ is the Kronecker product. It is also the same as linear summation in 3DMM. A(m) can be obtained by higher order singular value decomposition (HOSVD) for different feature classifications, and $A^{(M)}$ can express large feature classifications of faces, such as skin colour, expression, lighting, gender, etc. The following equation is optimised to determine the Loss function, as shown in Equation 1.5：

$$\underset{B_{(1)},\{A^{(m)}\}_{m=2}^{M}}{argmin} \left\| X - B_{(1)}\left(\bigodot_{m=2}^{M} A^{(m)}\right) \right\|_F^2 \qquad \textbf{Equation 1.5}$$

1.2.3 End-to-end Face Reconstruction

With the rise of deep learning in recent years, many researchers have started to experiment with the combination of deep learning and 3D face reconstruction. Convolutional Neural Networks (CNNs) have been widely used in image processing, bypassing the complex modelling process of feasible variant models such as 3DMM and designing their own representation that facilitates the training of input and output neural networks without losing 3D face information.

（1）VRNET（Volumetric Convolutional Neural Networks Regression

Network）

Jackson et al. proposed a new 3D face geometry representation method (Jackson, A.S. et al. 2017) to construct a mapping relationship between 2D pixels to 3D pixels, transforming the construction of 3D face models into a semantic segmentation problem, and then transforming directly from 2D facial images to the corresponding 3D models by training appropriate CNN networks (Liu, F. et al. 2016), instead of predicting the parameters of 3DMM models. It is independent of face pose, expression, and occlusion, thus avoiding the tedious initial work of 3DMM methods, such as 3D scanning, convex surface optimization, as well as all those complex aspects of Texture, Albedo, Render, etc. in 2.5D reconstruction methods. The trained dataset has a complete statistical model, so it can solve arbitrary pose, expression and occlusion problems, as well as by fitting to fix unrepresentable parts of the face. This approach effectively solves the limitation of traditional methods to be limited to a specific face model and can be more widely adapted to natural face photographs.

The method constructs two sets of encoding/decoding structures through a simple convolutional neural network, similar to the funnel network model of the hourglass network (HG) (Long, J. et al. 2015). The first set of convolutional layers is used to compute a feature representation of fixed dimensions, and this representation is further processed back into the spatial domain to re-establish the spatial correspondence between the input image and the output volume, with features of different resolutions hierarchically combination for pixel-by-pixel prediction. The second group has exactly the same structure as the first group and is used to refine the output results. the network input to the VRN requires only a single 2D face image and does not require the images to be precisely aligned or dense correspondence to be established to enable end-to-end direct regression from a single 2D image to a voxel representation of 3D facial geometry.

VRNET corresponds to the xyz axis size of 192*192*200, xy corresponds to 2D face pictures, the training set selected by VRNET is 300W-LP, where each face picture corresponds to a 3D mesh deformed from the BFM model. The face pictures are put into the xy coordinate system, and the mesh is put into the xyz voxel space.

after voxelization and smooth filling, the three-dimensional structure of the face can be represented by 200 two-dimensional images with size of 192*192 . The advantages of the method are that it designs its own 3D face representation method with a simple and brutal structure and receives unimpressive results. The disadvantages are also obvious: the vertices of the 3D face predicted by the CNN are not fixed, which means that we also need to perform a step of spatial alignment to align a template of fixed vertices to the 3D face predicted by the CNN; the reconstruction resolution is not easily scalable and the overall computational effort is huge, Although the method does not suffer from model space constraints, the face representation loses the semantic relationships of the mesh vertices, and the prediction of Voxel Data requires an extremely complex network structure and a very large amount of time, according to the paper, the authors' experiments taking up to 2 days for one batch of training.

（2）PRNET（Position Map Regression Network）

PRNET (Feng, Y. et al. 2018) is an unconstrained end-to-end modelling approach that allows for both face 3D reconstruction and face high-density alignment. Compared to VRN, the biggest advantage is that only one sheet of 2D data is needed to represent 3D MESH, the amount of data to be processed is significantly reduced. Its core novelty is the design of a UV position relationship control map, which can record the 3D information of all facial point clouds on a 2D UV image, using UV relationships, preserving the 3D information as much as possible to a certain extent, and the regression network can use a lightweight network structure, such as a residual encoder plus a convolutional regression decoder for learning, and then trained by locally weighted masks from a single 2D face images end-to-end to obtain a UV position control map, and finally obtain the 3D mesh model by inverse conversion.

The task of 3D face modelling based on a single RGB image is to obtain the 3D structural information of the face and the dense relationship between 3D structure and single RGB image from the regression of a single face image. Existing deep learning networks cannot directly regress the 3D space, so it demands to find a proper way to represent the data, and the most intuitive way is to reshape the 3D space into a one-

dimensional vector, i.e., using one dimension preserving the spatial information of the 3D vertices, but this will lose the two-dimensional spatial information and is not effective. Other research results have applied deformation models such as 3DMM for feature extraction, training only the relevant model parameters, but such methods are severely constrained by the expressiveness of the feasible variational model. Another example is the voxel representation method (Zhu, X. et al. 2016) used by VRENT, which successfully solves the previous problem and can restore the original 3D spatial information to the maximum extent (losing the information and accuracy such as coming to vertex index), but his network needs to train a 192*192*200 voxel space, and the computational volume can be huge. In response to the above problem, the author was inspired by the graphic UV texture and mapped the xyz of the 3D coordinates, eliminating the z-axis information, to the UV coordinates by the following equation 1.6.

$$v \to \alpha_1 \arctan\left(\frac{x}{z}\right) + \beta_1, \quad u \to \alpha_2 \cdot y + \beta_2 \qquad \textbf{Equation 1.6}$$

The network structure of this method is simple, easy to implement, fast and can even handle high frame rate videos, and the obtained face models meet a certain level of granularity. However, according to our observation during the experiments, in some case, the generated face MESH has obvious stripes and does not work as well as it should when the face appears in large poses or when there is occlusion, the results over-fitting.

1.3 Related applications

1.3.1 Reconstruction of 3D human face models

An image containing a face image can directly generate an OBJ file containing 3D vertex information and texture information. As the method in this dissertation only targets the reconstruction of the model and does not involve the repair or prediction of

the face texture, a clear positive face image is required if a more complete and clearer textured 3D model is to be obtained.

1.3.2 Three-dimensional pose estimation

For the estimation of face pose, the centre point of the bounding box of face pose is determined by the average of the first 37 of 68 key points, the boundary is determined by the angle of pose rotation obtained from the Bounding Box by face detection, and the depth of the bounding box is determined by calculating the maximum Z value of MESH, so that the pose and rotation angle of the face can be better displayed.

1.3.3 Human face depth estimation

Similarly, using the generated 3D MESH, the corresponding depth greyscale map is rendered by a projection operation and preserves the depth value Z.

1.3.4 Human face editing

Face editing is one of the most popular applications in the field of faces, and applications such as beauty algorithms and fine-tuning of features can all be considered as face editing. An important prerequisite for face editing is the detection of key points and alignment of faces, so face-based 3D reconstruction can be very useful for face editing. Figure 1.5 shows a typical operation of face editing - face swapping. Firstly, the algorithm in this dissertation generates face models that need to be replaced with face textures, and since they are all based on the BFM model, the semantic information of each vertex is the same, so we can swap the corresponding vertex colours and then project them onto the corresponding 2D images to achieve face skin replacement.

1.3.5 Film and Animation field

Visual engine systems are typical application of the CG field. With the advent of the 3D film generation, not only face reconstruction but also the entire graphic 3D reconstruction technology has been fully applied, and CG technology and the film industry have developed in leaps and bounds, complementing and promoting each other. This has enabled many unreal worlds such as cyberpunk, fantasy worlds and space, or acting actions that actors could not perform, to be brought to the cinema screen through CG technology. In *Fast & Furious 7*, the character Paul Walker is brought back to life by applying a large number of face reconstruction algorithms. Firstly, a large amount of face and expression data is collected through existing video material, then the data is used to reconstruct a model of Paul Walker's face, then feature points are found to be marked in the stand-in actor, and finally a moving image is formed through face alignment algorithms. Using these techniques, a combination of CG and live action human characters is realized and delivers a stunning visual experience to the audience.

1.3.6 Game Production

Game making proper is about creating a virtual world through a certain logical organization. With the dramatic increase in computer computing power, the emergence of AR and VR has broken the boundaries between virtual and reality. Through face 3D reconstruction technology, it is possible to create highly realistic and personalized character images, and users can also do image capture and create personalized and customized player characters, 3D avatars, 3D expressions, etc., enabling players to immerse themselves in the real world and have more fun with the experience.

1.3.7 Security applications

There are no two identical leaves in the world, nor are there two identical faces, and this uniqueness makes it possible for face modelling to be used as a means of identifying identity. As a result, face technology can then be used in areas such as mobile phones, finance, justice, security, transport and education. For example, unlocking mobile phones with 3D face information, conducting business at financial terminals with face recognition, monitoring traffic or apprehending criminals using the SkyEye system, and carrying out face biometric identification in educational premises and examination centers.

1.3.8 Medical Beauty

3D face reconstruction technology can also be used in medical aesthetics. When it comes to problems with the human head, prior modelling of the patient by means of 3D face reconstruction technology enables a more visual analysis of the condition and more accurate location of the lesion, which can be practiced repeatedly through simulation operations in the virtual world, thus reducing the risk of surgery and increasing the success rate of the cure. As for aesthetics, by modelling the client's face in 3D and making changes on the model in advance in order to develop a better aesthetic plan, the client can preview the results of the operation more visually.

1.4 Research results

In this dissertation, in order to study the implementation of 3D face reconstruction based on a single view of a face, existing techniques are learnt, developed, integrated and this dissertation also proposed effective innovations, so in order to improve the efficiency of 3D face reconstruction and achieve the goal of 3D face reconstruction relies on a single view of a face, a reasonable experimental control

group is designed, and the experimental results are analysed. The research in this dissertation are as follows：

A theoretical study of 3D face reconstruction. In this dissertation, the 3D face reconstruction method based on the orthographic view of the face and the related knowledge of graphics needed are presented.

Face dataset selection and pre-processing. This research used the open-source face dataset 300W-LP, which is publicly available in academia. The data is based on the 3DMM model deformation representation, where a face image corresponds to a deformation parameter file, and each portrait is selected with many pose angles, while the deformation parameter file can be used to obtain a 3D MESH file through a certain rendering process, and the corresponding UV map is generated through the UV-Pos Map algorithm as ground-truth. Secondly, in order to expand the dataset and improve the training quality, data enhancement operations can also be performed on samples of the dataset.

Feature extraction. The 68 commonly used feature points have their corresponding fixed vertex indexes and are therefore simple to manipulate. If we chose to use third party datasets or custom photos, face detection, cropping and feature point labelling can be performed by some excellent face detection algorithms.

The design of neural network structure and the use of residual structure. The size of the training data in this dissertation is 256*256*3 image color information and the ground-truth is 256*236*3 three-dimensional coordinate information. Therefore, the dataset needs to be scaled and cropped to a certain scale space to ensure that the range of the human face fills the space as much as possible and retains the maximum amount of information. By using residual learning, a new network structure is constructed to obtain a better 3D reconstruction of the human face.

Model training and experimentation. A large number of model training and comparison experiments are conducted. For example, dataset enhancement experiments, normalization experiments, network model comparison experiments, loss function comparison experiments, etc. Through analyzing the comparison results, the most suitable method for human face 3D reconstruction is selected by us.

2. Fundamentals of Graphology

2.1 Human face model representation

The face model in this dissertation is based on the BFM model. A human face model, in general, contains three parts of graphically relevant data: vertex data, vertex index, color information, and three parts of face information: key point coordinate index, deformation parameters, symmetry index, etc. The BFM model has about 60K vertices, which, together with the index, fixes a model template, and the BFM is symmetrically optimized for vertices. The advantage of templating is that the use of a fixed set of vertex indexes to represent a part of the face makes full use of the vertex indexing information to preserve the face space language, for example, any face expressed in the BFM face model has a fixed vertex index for the nose portion, and the nose areas on the face can be obtained by correlating the indexes. BFM textures are stored using a vertex color scheme and the color information corresponds to the color scheme of a single vertex, which is less effective and has a low resolution 256*256 when stored as a 2D image. The vertex-to-vertex color information is obtained by graphical rendering, using linear interpolation. There are also standard UV textures or deformations of UV textures used to store face textures.

The 300W-LP utilized in this research uses UV to store spatial information rather than color information. The BFM model is a 3DMM-based face model consisting of 53,215 vertices, and the UV coordinate mapping table records the relationship between each vertex of the BFM model and the UV position mapping map, so it is an effectively 53215*3 size transformation matrix. This means that the coordinates of

each vertex of the BFM model in the UV position map can be found using this UV coordinate mapping table. Once the point cloud data of the face model has been obtained, the dashed purple line indicates a weak perspective projection of the face, which results in a corresponding 2D image, the solid blue box in Figure 2.1 being the 2D planar image of the face. It is also assumed that the point cloud coordinates of the 3D face are defined in the left-handed Cartesian coordinate system when the weak perspective projection is performed, and it is aligned with the origin in the pixel coordinate system of the 2D image. Finally, all vertices of the model are traversed one by one, assuming that the traversal reaches vertex k, whose 3D coordinates are (X_k, Y_k, Z_k). Then consult the UV coordinate mapping table and find the pixel coordinates (U_k, V_k) of vertex k in the 2D UV position mapping map according to the mapping relationship.

Thus, the mathematical expression 2.1 for the UV position mapping map is:

$$(U_k, V_k) = \mathrm{Pos}(X_k, Y_k, Z_k) \qquad \textbf{Equation 2.1}$$

The Pos function represents the UV position mapping table and (X_k, Y_k, Z_k) denotes the coordinates of the ith vertex in the BFM model, while the coordinates of that point on the UV position mapping map 2.1 are (U_k, V_k).

Figure 2.1 UV position mapping of the human face model

2.2 Image Rendering

In some non-end-to-end face reconstruction algorithms, it is often necessary to use a rendering pipeline to generate a 2D face picture by projecting a 3D face model.

This involves the 3D manipulation of graphics and imaging principles.

As shown in Equation 2.2 below, this is a typical rendering process.

$$g(\alpha, m) = V = f * Pr * R * S + t_{2d} = M(m) * \begin{bmatrix} S \\ 1 \end{bmatrix} \qquad \textbf{Equation 2.2}$$

f represents a scaling vector on the xyz axis; Pr represents a projection operation, either orthogonal or perspective projection; R is a rotation matrix; T_{2d} is a translation vector in the xy direction; S is the source model Shape, and the whole PSRT process can also be represented by a four-dimensional matrix M. The point of this operation is to obtain the corresponding pose angle by adjusting the model angle. Alternatively, the projected image can be used to compare the loss with the source image.

2.3 Human face detection and key point detection

Face detection is required, as the input face image may not contain just the target face in practice but may also contain distracting factors such as background. Face detection generally involves searching any given image using a certain strategy to determine whether it contains a face, and if so, returning the position, size and pose of the face. Face detection method is currently better with deep learning-based detection algorithms, which are more robust and resistant to interference than traditional algorithms. For example, the Compact Cascade CNN (Kouris, A. et al. 2018) based on Convolutional Neural Network has fewer network parameters, but it can be implemented on devices with lower computing power, the process of running is extremely fast and has a good accuracy rate; then there is the MTCNN (Xiang, J. et al. 2017) based on Multi-Task Cascade Convolutional Neural Network, which improves both the speed and accuracy of face detection based on the Compact Cascade CNN.

In recent years, face feature point detection algorithms have been divided into two main categories: data-driven optimisation algorithms and supervised learning-based regression algorithms. The essence of data-driven optimisation algorithms is to establish a physical model between the face feature points and the input image, and to construct the whole problem as an energy-optimal based optimisation problem using a priori information from the dataset as constraints or solution directions. The classical

Active Shape Model forms the feature points into a global face shape assumes that the face feature points are essentially located at the strong edge of the image. The objective of optimisation is to maximize the magnitude of the image gradient near the feature points, and the statistical information of the face shape is combined to calculate the feature point positions in the subspace of the face shape. Other models are the Active Appearance Model, the Constrained Local Model and the 3DMM model, which lifts the statistical shape and texture of the face into 3D space, searches for the optimal 3D shape of the face in 3D space and projects the texture onto the 2D image space to minimize the difference between the projected image and the target image. The 2D image space is used to minimize the difference between the projected image and the target image and obtaining the feature points of the model. The more common supervised learning regression algorithms are face feature point detection algorithms based on cascaded regression models, in which the weak regressors are all linear regressions, and the difference lies in the different local feature description operators used, such as the Supervised Decent Method algorithm based on Sift features, the multi-angle recognition algorithm based on Hog features, the and the Random Fern algorithm based on LBP features.

2.4 Human face alignment correction

Face alignment is the process by which a computer algorithm finds a predefined face shape, or feature point, in a picture of face. Face alignment is usually a gradual iterative process, starting with a rough estimate of the outline and then refining the estimate of the final shape through progressive iterations. Two types of data are typically utilized in the alignment process, the texture information of the face and the mesh of the face, which provides a three-dimensional spatial constraint. The purpose of face alignment is mainly to detect the semantic parts of the face such as eyes, nose, mouth, and eyebrows, then semantically align them with a standard face or a pre-defined face model. Therefore, in general, face alignment is performed simultaneously with face feature detection, and in practice is also influenced by scale, pose, lighting,

occlusion, complex expressions, etc.

From a technical implementation point of view, face alignment can be divided into two types: generative methods and discriminative methods. Generative methods use face shape and texture to generate models, which treat face alignment as an optimisation problem and seek to fit the optimal shape and texture parameters of the input face, including methods such as AAM (Active Appearance Model) (Zhang, W. et al. 2017) and ASM (Active Shape Model) (Milborrow, S. et al. 2008). These methods typically use a local detector or regressor that learns independently to locate key points of the face, and a global face model to fine-tune and normalise the predicted results, including Constrained local models (CLMs) (Cristinacce, D. et al. 2008), Deformable part models (DPMs) (Felzenszwalb, P. F. et al. 2010), etc. and other classical algorithms. Most of these algorithms are based on feature points or certain model parameters, which are all sparse alignments, but there is also a class of algorithms called density alignments. This type of algorithm, generally through the training of neural networks, uses face images to estimate a 3D model of the face, and uses that model to fit the corresponding 3D face model. This method not only obtains 3D face feature points, but also matches SIFT feature points and face contours. The problem of incompatibility between different databases due to the definition of feature points is also solved. Among such methods, the traditional ones are Thin Plate Splines (TPS) plate splines interpolation, Optical Flow optical flow estimation, Non-Rigid Iterative Closest Point (NICP) non-rigid progressive iteration, etc.; the deep learning-based methods are Dense Face Alignment, DenseReg, PRNet, etc.

3.Methods

This chapter will focus on the whole process and work of face 3D reconstruction. Firstly, to achieve end-to-end deep learning training, we selected the 300W-LP dataset, which is a large pose open-source dataset based on the BFM model, from a variety of datasets. To enhance the expressiveness of the dataset, we made certain enhancements to it, such as image rotation and scaling, modification of color information, addition of noise. After that, we cropped the data for the face so that it could render the maximum face content. After these operations, the dataset is brought closer to reality. We then perform a face pose correction process to roughly align the face pose with the face UV template and reduce the effect of pose errors in the face reconstruction.

We then describe the deep learning network structure and choice of loss function used in this dissertation. Our approach is based on end-to-end reconstruction, so the input is a 256*256*3 RGB image, and the output is a 256*256*3 UV-xyz map with a similar structure to the RGB image. Therefore, we can use some ideas from image processing for learning optimisation. We propose a neural network structure based on a hierarchical residual learning framework to enhance the learning of image details. The whole network structure is divided into two parts, the first part is a commonly used face 3D reconstruction structure, including an encoder network structure and a decoder network structure, which is mainly used to obtain an approximate face model; The second part is a residual network, which is used to learn the mapping of the residuals in the image with the deeper level, so as to enhance the expressiveness of the first network and thus improve the 3D reconstruction quality.

3.1 Introduction to the data set

3.1.1 300W-LP data set

The 300W-LP dataset (Zhu, X. et al. 2016) consists of sub-datasets such as AFW, LFPW, HELEN, IBUG, etc. The faces in each image are labelled with 68 face feature

points, and the face images within the dataset were collected in unconstrained natural scenes with large variations in expression, lighting conditions, pose, occlusion, face size, etc. The 300W-LP dataset was developed by the team 3DDFA. The team extended the 300W dataset by using a 3DMM-based fitting method to obtain a 3D dataset of large pose face data, so all the 300W-LP data is annotated by the 3DMM deformation model. The team also transformed the Pose, Light and Color parameters of the annotated 3DMM deformation model by data augmentation, thus artificially composing many face images of the same individual in different poses. 61225 face images of 3837 faces are available in the 300W-LP dataset (of which 37676 are from HELEN, 16556 from LFPW, 5207 from AFW and 1786 from IBUG), containing different face angles with associated face feature points, camera parameters and annotations of 3DMM coefficients. In the field of face 3D modelling, 300W-LP is one of the most widely used datasets. This dataset is used as the training dataset in this dissertation.

3.1.2 AFLW data set

AFLW (Annotated Facial Landmarks in the Wild) is a large-scale unconstrained face dataset containing over 20,000 face images, of which 59% are female and 41% are male. The face images in this dataset have multi-pose and multi-view attributes, with deflection angles ranging from -90 degrees to 90 degrees, involving different poses, expressions, lighting, and skin tones, with most being color images and only a few being grey images. Each of these faces is annotated with 21 visible feature key points; invisible key points are not annotated as they are marked by the naked eye. This dataset is ideally suited to face research tasks related to multi-angle, multi-face and multi-pose problems, and is also an important dataset in the field of face key point detection and alignment.

3.1.3 XM2VTS data set

The XM2VTS dataset is an extended dataset of the M2VTS dataset, containing 2360 frontal photographs of faces, derived from 295 individuals. Each face image contains 68 facial feature points, and the shape of the face from which these features are formed.

3.1.4 AFLW2000-3D data set

The AFLW dataset lacks paired 2D images and 3D models, it has only 21 visible key points, which can lead to significant ambiguity in the 3D shape and such images contain large poses in an unconstrained environment. Therefore, there are difficulties in evaluating 3D face alignment and 3D face reconstruction for the data in the AFLW dataset. To address this issue, the first 2000 images of the AFLW dataset were selected to fit 3D faces using the 3DMM deformation model approach, resulting in the construction of the AFLW2000-3D dataset for evaluating 3D face alignment and 3D face reconstruction on challenging unconstrained images. Compared to the AFLW dataset, the AFLW2000-3D dataset extends its annotation to include 3DMM coefficients corresponding to real 3D faces and 68 3D facial key points for each face image, making the AFLW2000-3D dataset a challenging dataset to test the performance of the algorithm for both face alignment and face reconstruction tasks. The AFLW2000-3D dataset is a highly applicable dataset when used as a test dataset. Because of the reasons above, this dataset is used as a test dataset in this dissertation.

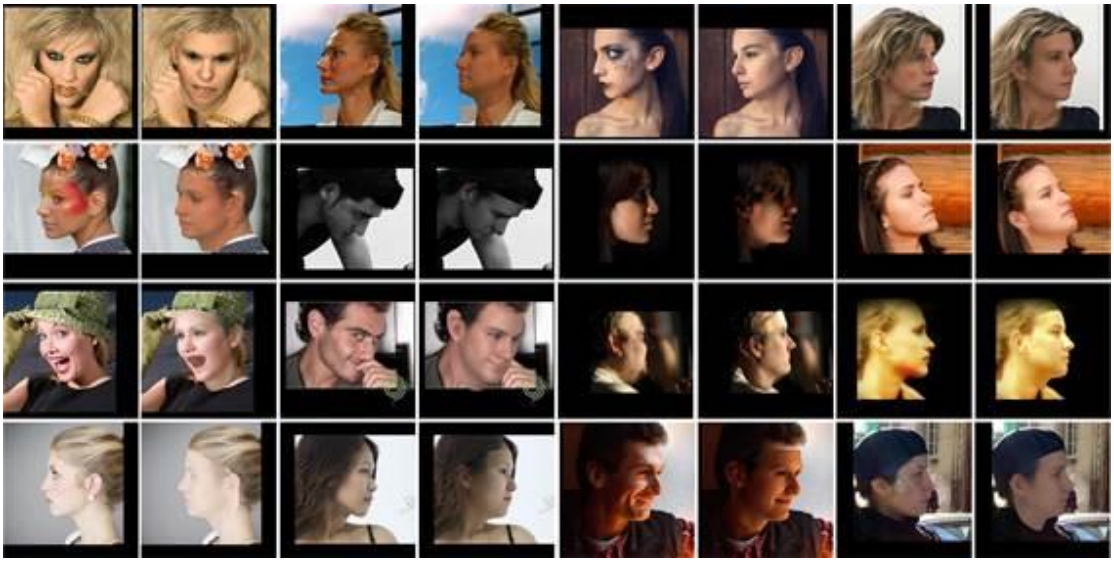

Figure 3.1 Some samples of AFLW2000-3D dataset

A sample AFLW2000-3D dataset is shown in Figure 3.1. For each pair of faces, the left image is the original face image, and the right image is the reconstructed face model by using the 3DMM model. The output from the figure shows that the dataset recovered the overall shape and expression information of the face better, but lost local details such as the eyes, nose and mouth with the texture details of the face.

3.2 Pre-processing of data

3.2.1 Generation of Ground Truth

We chose the 300W-Lp dataset as the training data, and its 3D mesh model data is a deformation parameter based on the standard BFM model, i.e., a MATLAB code .mat file, so firstly we need to obtain the 3D vertex data by running the .mat file, after the standard BFM model deformation. As deep learning cannot directly train 3D models, the 3D data needs to be downscaled to 2D without significant loss of spatial information, and the UV-Pos Map method is utilized in this dissertation.

The UV map is a graphical concept, "UV" in this case is short for the u, v axis of the texture mapping coordinate system, similar to the x, y and z axes of the Cartesian 3D coordinate system. It not only carries color information, but also defines the position of each UV point on the image in 3D space. These 2D pixel points are

interconverted with the vertices on the 3D model through a functional mapping relationship. One UV point may correspond to more than one 3D vertex, but each 3D vertex can only correspond to one coordinate at UV. UV mapping is the 3D modelling process of projecting a 2D image onto the surface of a 3D model for texture mapping. This means that every point on the surface of the model object can be precisely mapped to the UV image. The empty parts between the points of the object model are usually smoothly interpolated by a pixel shader in the rendering pipeline. This is known as UV mapping. The globe UV mapping process is shown in Figure 3.2.

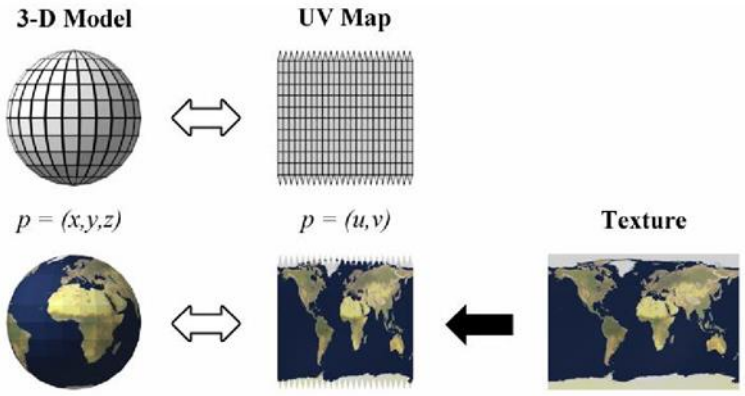

Figure 3.2 UV mapping of the Earth

When creating a model as a polygon mesh using the 3D modeler, it is possible to generate the corresponding UV texture coordinates for each vertex in the mesh. A common method is for the 3D modeler to automatically lay out triangles on a plane by expanding the triangular mesh at the seams. If the mesh is a UV sphere, the modeler may convert it to an equirectangular rectangular projection; once the model is opened, the modeler can use the expanded mesh as a template to draw a separate texture on each triangle. When the scene is rendered, each triangle will be mapped to the corresponding texture in the 'decal sheet'. The UV mapping can therefore be generated manually by the modeler, or automatically generated by a software application, or some combination of the two, which will then be adjusted and optimized by the modeler to minimize seams and overlaps. If the model is symmetrical, the modeler may overlap opposing triangles to allow two faces to be drawn simultaneously, with UV coordinates optionally applied to each face. This means that a shared spatial

vertex position can generate different UV coordinates for each of its corresponding triangles, and ultimately adjacent triangles can be cut and positioned on different areas of the texture map. In this dissertation, a Tutte embedding algorithm with Laplace weights is used to expand the triangular face piece of the 3D object in two dimensions, and the boundary of the MESH is finally mapped into a square. the BFM model has more than 50,000 vertices, so a UV of 256*256 size is chosen to be able to store all the vertex information on it. At the same time, we also need to map the index of the face region in the triangular index of the BFM to the UV coordinates, and the index of the 68 feature points of the BFM to the UV coordinates through the correspondence.

3.2.2 Cropping

In order to make full use of the feature space, we fill the entire input space with face data, so some cropping of the original images is required. To enrich the expressiveness of the dataset, the face images can also be rotated at certain angles to train the face model from different angles. The ground-truth size is first preset to 256*256, and then both the image and the 3D MESH are scaled down to a fixed scale space. The dataset we use is not cropped for faces, which means that although face information is present in the image, the center of the face is not in the center of the image. Therefore, we need to first detect the face. In this dissertation, we choose fastMTCNN (Xiang, J. et al. 2017), which is better in terms of speed and efficiency, to detect the face, and then adjust the face to the center of the image, which also achieves cropping out the meaningless data in the original image. For datasets where face location information is provided, cropping can be done directly by coordinates.

3.2.3 Enhancement

In the dataset used, the direction of the central axis of the face is mostly vertical, so we can rotate the face pose to expand the dataset, and the following method is used in this dissertation:

（1） Rotation operation

The face images in the dataset are rotated from -45 degrees to 45 degrees to generate new face images and, of course, the same operation is performed for the corresponding 3D labels of each image to keep the data consistent.

（2）Zooming operations

The face images in the dataset are scaled or expanded by a factor of 0.7 to 1.3 to produce face images of different sizes. We also perform the same operation on the corresponding three-dimensional labels to keep the data consistent.

（3）Color adjustment

As it is difficult to guarantee identical lighting, color and skin tone conditions in the dataset, random perturbations need to be added to one of the RGB channels, so we will randomly select 1-3 of the RGB channels and multiply each by a value somewhere between 0.5 and 1.5 to achieve color enhancement.

（4）Adding noise for the data set

As the approximate area of the face is known, for each image, a combination of Gaussian noise, random noise and random mask patch is used, from which 1-3 types of noise are selected and applied to the face area in the image to achieve image enhancement in terms of noise.

（5）Image translation

For the pre-processed images, to enhance the spatial generalisation performance of the algorithm model for the face region, we perform a random translation operation of -10% to 10% in both the X and Y directions.

3.2.4 Fixed size and normalisation

Based on the consideration of balancing the performance and efficiency of the algorithm, the method we have designed specifies the size of the input image to be 256*256*3. In addition, the inputs to the deep learning network model, which are typically between 0 and 1, it also needs to be normalised to the input values. Since some of the data has a UV map, we apply two normalisation methods. When the

activation function is sigmoid, the max-min normalisation is employed and the final result is to be between 0 and 1. When the activation function is tanh, max-min normalisation is also employed, but the final result is required to be between -1 and 1.

3.3 Human face processing

3.3.1 Human face detection

The implementation phase of the algorithm in this dissertation is the detection and cropping of the human face image. We use methods such as Dlib (Kazemi, V. et al. 2014) and fastMTCNN, these two algorithms are mature algorithms for face detection on 2D images. The algorithm is able to obtain the bounding box (i.e., the target detection box) of the face and 68 face feature points.

Through the bounding box, we can obtain the upper, lower, left, and right boundaries of the face, so that we can fully obtain the whole face information. From Figure 3.3 we can see that the face information is concentrated in the centre of the whole image, which facilitates the implementation of the subsequent algorithm.

3.3.2 Human face correction

In the traditional PRN approach, face alignment and face reconstruction will be trained together. This type of approach could have a poor result when adapted to large pose faces, especially when face inversion occurs at the input, and will fail to generate a correct face model, because during the UV-Pos mapping, the UV template is a standard positive front face texture with one-dimensional information lost, and after correction, the X and Y of the vertices will have less effect on Z. At larger face poses, the X, Y of the vertices differs significantly from the template, resulting in a large training error, which is reflected in the 3D reconstruction process, so that the face model cannot be obtained. To address this phenomenon, the alignment of the face and the reconstruction of the face has processed in separate steps in this dissertation.

For face pose correction, we need to make the face pose is adjusted to roughly

the vertical orientation of the 2D image, as the dataset contains tiny poses of the face, so it needs not be strictly vertical, nor is the face facing forward, as long as the eyes position are roughly horizontal.

In the face detection process we did before, among the 68 features already acquired, we selected 12 feature points indexes of the eyes, the nose index between the two eyes and the jaw centre point index. The nose index and the jaw centre point are rotated, so that they form a T-shaped structure, then keep rotating the nose index and the jaw centre point by 90 degrees and then a straight line is fitted through these 14 points, the slope of this line is the angle of the face deflected by the pose, by rotating the angle the corrected face picture can be obtained. If the X-value of the left eye feature point is greater than the X-value of the right eye feature point, the whole picture will also need to be rotated by 180 degrees and flipped. As shown in Figure 3.4, after the correction, the face can be reconstructed in 3D using the deep learning model.

3.4 The network model of 3D reconstruction

We propose a layered, residual-based face 3D reconstruction network structure, which is divided into two main parts, as shown in Figure 3.5. The first layer is an encoding-decoding structure, with the encoder consisting of a cascade of 10 residual blocks and the decoder consisting of 17 layers of deconvolution [21] to generate a rough Brief-output. When we designed this part of model, we refer to PRNet, which is a mature and general face reconstruction network. The second layer is a typical ResNet structure with 8 residual blocks at its core, which are used to learn the difference characteristics between the summary data and the original image, and the output is overlaid with the summary data to generate our face vertex data, and the final 3D MESH is generated by fixing the UV face index. This 28-layer network model structure, through the residual structure of connect, can fuse features of different granularity to obtain the final 3D reconstruction results. Residual Block 1 is a three-layer residual structure and Residual Block 2 is a two-layer residual structure.

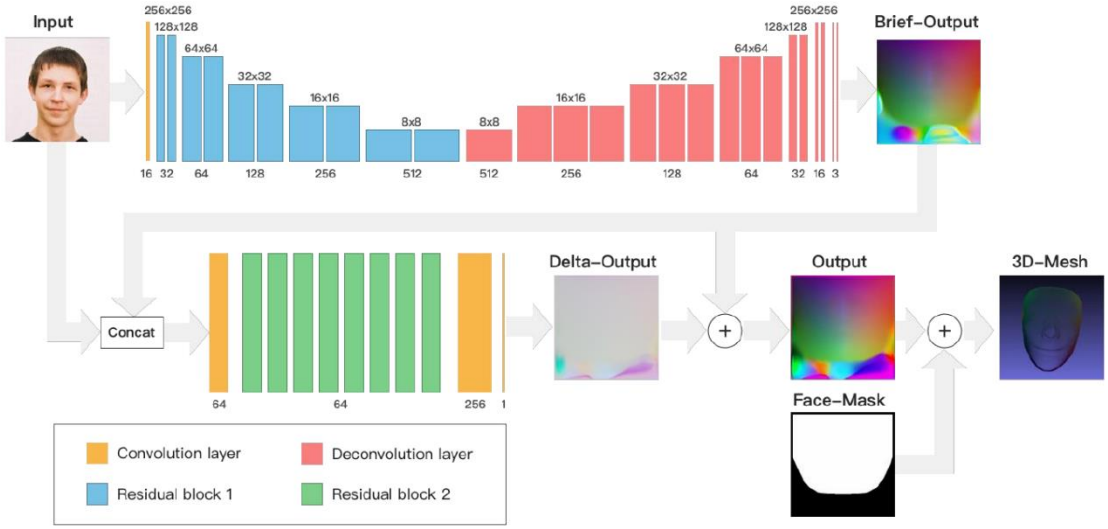

Figure3.5 The network model structure

3.4.1 Encoders for models

In general, the more complex the design of the network structure and the more layers of the network, the better the results tend to be, it means that the network has better expressiveness. In fact, however, as the network stacks up, the backpropagation of gradient information between the layers starts to become problematic and the gradients either tend to disappear or the gradients explode, and the results become worse instead. As a rule of thumb, the depth of a neural network can have a relatively positive effect on its performance to some extent. However, depth is not an absolute positive correlation, and after a certain maximum number of layers reached, the computational cost of training the network increases exponentially, and performance is not improved, but rather 'degraded'. In addition, too many parameters can lead to a gap between the results on the test set and the results on the training set, that is the overfitting problem. The emergence of residual neural networks (He, K. et al. 2016) has solved these problems by significantly improving the network structure of convolutional neural networks by combining multiple layers and fitting them into a residual function.

In summary, this dissertation also adopts a multi-channel residual network structure. Through continuous experimental optimisation, the best performance was

achieved with the parameter configurations shown in Table 3.1, and the parameters of the residual structure are shown in Table 3.2.

Table3.1 The parameters of the coding layer

layer	conv size	stride	output size
Conv1	4	1	256*256*16
ResBlock1	4	2	128*128*32
ResBlock2	4	1	128*128*32
ResBlock3	4	2	64*64*64
ResBlock4	4	1	64*64*64
ResBlock5	4	2	32*32*128
ResBlock6	4	1	32*32*128
ResBlock7	4	2	16*16*256
ResBlock8	4	1	16*16*256
ResBlock9	4	2	8*8*512
ResBlock10	4	1	8*8*512

Table 3.2 the parameters of the residual structure

layer	conv size	stride	output size
Shorcut	1	L	256*256*N
Conv1	1	1	256*256*N/2
Conv2	4	L	256/L*256/L*N/2
Conv3	1	1	256/L*256/L*N

For the input face image, it first passes through a convolutional layer with a kernel size of 4 and a step size of 1 with a number of 16, to obtain 16 256*256 feature maps. These residual blocks have three control parameters, the number of convolution kernels N and the step size S. The specific parameters of each residual block are shown in Table 3.1. The main path of the residual block consists of three convolutional layers with the parameters shown in Table 3.2. The first layer has a convolutional kernel size of 1, a step size of 1 and the number of convolutional kernels is N/2; the second layer has a convolutional kernel size of 4, a step size of L

and the number of convolutional kernels is N/2; the third layer has a convolutional kernel size of 1, a step size of 1 and the number of convolutional kernels is N. The branch path is also a convolutional layer with a convolutional kernel size of 1, a step size of S and the number of convolutional kernels is N. The L depends on the stride in the ResBlock, and the results of the two paths are summed and output by the ReLU activation.

3.4.2 Introduction to the model's decoder

The decoder network structure mainly consists of 17 deconvolution layers, as shown in Figure 3.3. The deconvolution parameters are obtained by experimental tuning and the specific decoder parameters are shown in Table 3.3.

Table3.3 the parameters of decoder

layer	conv size	stride	output size
Deconv1	4	1	8*8*512
Deconv2	4	2	16*16*256
Deconv3	4	1	16*16*256
Deconv4	4	1	16*16*256
Deconv5	4	2	32*32*128
Deconv6	4	1	32*32*128
Deconv7	4	1	32*32*128
Deconv8	4	2	64*64*64
Deconv9	4	1	64*64*64
Deconv10	4	1	64*64*64
Deconv11	4	2	128*128*32
Deconv12	4	1	128*128*32
Deconv13	4	2	256*256*16
Deconv14	4	1	256*256*16
Deconv15	4	1	256*256*3
Deconv16	4	1	256*256*3

Deconv17	4	1	256*256*3

The first deconvolution layer has a convolution kernel size of 4, a step size of 1, and an output feature scale of 8*8*512; the second to fourth deconvolution layers have a convolution kernel size of 4, step sizes of 2, 1, and 1, and an output feature scale of 16*16*256; the fifth to seventh deconvolution layers have a convolution kernel size of 4, step sizes of 2, 1, and 1, and an output feature scale of 32*32 The size of the convolution kernels in layers 8 to 10 is 4, the step sizes are 2, 1 and 1 respectively, and the output feature scale is 64*64*64; the size of the convolution kernels in layers 11 to 12 is 4, the step sizes are 2 and 1 respectively, and the output feature scale is 128*128*32; the size of the convolution kernels in layers 13 to 14 is 4, the step sizes are 2 and 1 respectively, and the output feature scale is 128*128*32; the size of the convolution kernels in layers 13 to 14 is 4, the step sizes are The size of the convolution kernels of the 15th to 17th deconvolution layers is 4, the step sizes are 1, 1 and 1 respectively, and the output feature scale is 256*256*3, i.e. the final output size is the same as the input Image size, and the supervised data is Ground-Truth normalized to -1~1 space, so the selected is the tanh function as the activation function, the output data is the value of -1~1, by multiplying by 128, it can be recovered to the -128~128 space, i.e. the original 256*256*256 3D space, the result can basically express part of the face with clear human face contour and human face semantics, it will be used as the input of the next layer.

3.4.3 Residual structural layer

By inserting intermediate variables, the input information is enhanced, and what is learned is no longer just the original image features, but also the differences and details between the intermediate variables and Ground-Truth, serving to enhance the intermediate number of details. The result generated by the above codec layer can generate a better 3D model of a face, but the detail part will show serious creasing phenomenon and linear texture phenomenon. Residual learning (Guo, C. et al. 2018), on the other hand, retains the ability to retain detail to improve the quality of the

generated mesh detail, thus bringing the final output closer to the face model.

The core structure of the residual learning layer is 8 residual blocks, the inputs are the original graph and the output of the codec. First their third dimension is joined into a 256*256*6 tensor, then a convolution layer with a convolution kernel of size 3 and a step size of 1 is passed through to output a 256*256*64 feature map, followed by a stack of 8 identical residual blocks. The main branch of the residual block is two convolutional layers. The first layer has a convolutional kernel size of 3 and a step size of 1, and outputs a 256*256*64 feature map with ReLU activation, while the second layer has a convolutional kernel size of 3 and a step size of 1, and outputs a 256*256*64 feature map, directly superimposing the original input and the result of the second layer as the output of the residual block. Next are two convolutional layers for scale alignment. The first layer has a convolutional kernel size of 3 and a step size of 1, with an output scale of 256*256*256, and the second layer has a convolutional kernel size of 1 and a step size of 1, with an output tensor Delta-Output of 256*256*3. This tensor is the difference between the Brief-output (we learned about before) and the Ground-Truth, then we superimpose the difference with Brief-output, and make it finally pass through a tanh activation layer to generate the final output. The output is -1 to 1, multiplied by the corresponding scaling to reduce to the size of the spatial coordinates, the RGB value of each pixel represents the XYZ value of the 3D space, combined with the Face Mask of the known face position, the final human face 3D model can be rendered through the face index.

In terms of this, what is designed is a hierarchical progressive network structure, where first the codec layer gets intermediate result A, then intermediate result 1 with the original image as input to the residual network gets intermediate result B. The final result is obtained by combining A and B and passing the activation function. Of course, it is also possible to loop a third and fourth time. The approach in this research is to output a 3D model directly end-to-end from a 2D picture, which is not constrained by a deformable model, and the model structure of the face is only used as an intermediate variable to communicate the semantics.

3.4.4 Loss function and training

For the different layers of the network model, we designed separate loss functions to measure the difference between the network output and Ground-truth.

Mean Square Error (MSE) is a commonly used loss function. In the face reconstruction problem, as we are using UV-Pos Map, the whole Map face is concentrated in the upper area, and there is useless data in the lower left and lower right corners, and the whole face recognition is concentrated in some areas such as eyes, nose, mouth, etc. In order to better highlight these feature areas, we can add a layer of weight mask to the loss function according to Ground-Truth. For different Mask regions, different weights are set. For the non-face region, the weight is set to 0; for the 68 feature points and the gold-plated region, the weight is set to 1; for the mouth, nose and study regions, the weight is set to 0.4; and for the fourth part, the other regions, the weight is set to 0.3.

The first part of the loss function is Equation 3.1. It basically is a simple weighted summation, and we added a Mask weight on it:

$$L_1 = \frac{1}{n}\sum_{i=1}^{n}\left[\frac{1}{wh}\sum_{j=1}^{w}\sum_{k=1}^{h}\|I_i(j,k) - G_i(j,k)\|^2 W(j,k)\right] \quad \textbf{Equation 3.1}[21]$$

Where n is the batch size of each training batch, w、h are the width and height of the image, I represents the input Image, G represents ground truth, W represents Mask weights, and j, k are the coordinates on the image. The loss L_1 is used as the loss estimate for the second part of the residual network.

In our work, a number of symmetry constraints were added to improve the quality of the Encoder-Decoder, the first is the symmetry property of the face, as shown in Equation 3.2:

$$L_{kpt} = \frac{1}{n}\sum_{i\epsilon K, j\epsilon F}^{K}\left[\|V_i - 2_{30}\|_2 - \left\|V_j - 2_{30}\right\|_2\right] \quad \textbf{Equation 3.2}$$

This equation is a normal method in face reconstruction where V is the key point in the output UV-Pos, K and F are the index sets, respectively, and n is the number of symmetric feature point pairs. The key points here are those with symmetry characteristics among the 68 face feature points obtained from face detection in 3.3.1, including the 12 eye feature points mentioned in 3.3.2. This is shown in Figure 3.6, where the 58 key features outside the red area are used as inputs to the symmetric constraint function.

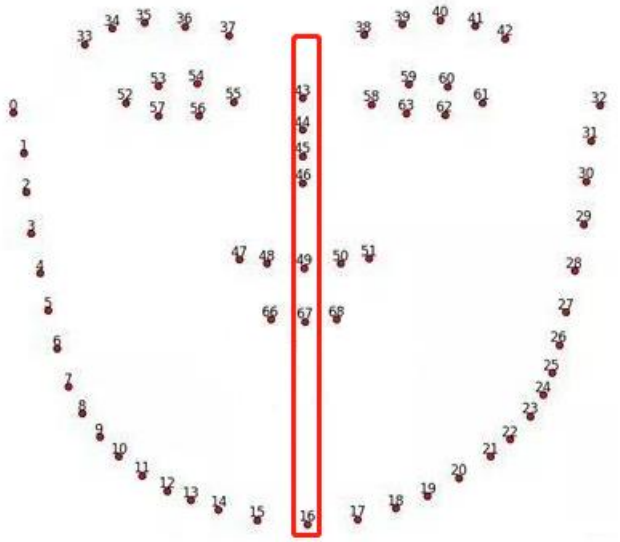

Figure 3.6 68 face feature points

Another constraint we use is the Laplace regularization method, this function is to smooth the model vertices, as shown in Equation 3.3:

$$L_{smooth} = \frac{1}{n}\sum_{i=1}^{n}\left[\sum_{I_i^{uv}\epsilon I^{uv}}\left\|I^{uv}(I_i^{uv}) - \frac{1}{|N_i|}\sum_{v_j^{uv}\epsilon N_i} I^{uv}(I_j^{uv})\right\|_2\right] \quad \textbf{Equation 3.3}$$

where n is the batch size of each training batch, I_i is a point in the output image$I$, and N_i denotes the four domain vertices of I_i. In summary, the total Loss function for the whole depth model is shown in Equation 3.4:

$$L_2 = \lambda_1 L_1 + \lambda_2 L_{kpt} + \lambda_3 L_{smooth} \quad \textbf{Equation 3.4}$$

3.4.5 Model training strategy and the Hyperparameter setting

The whole network can be trained in both distributed and end-to-end training. It has been experimentally proven that simultaneous training is more effective. This will be analysed in detail in the subsequent discussion of the experimental results. After repeated experimental validation, the batch size taken was 4 and the initial learning rate was 0.001. Using the Adam gradient optimisation method, the initial learning rate was reduced to half after every 5 epochs until the performance stopped improving in the last 5 epochs of validation.

4. Experiments and Results

In order to obtain the best face reconstruction results, this dissertation will conduct experiments in six areas. The techniques are validated in a comparative manner to demonstrate the superiority of the algorithms in this dissertation. We use Normalized Mean Error (NME) to calculate the errors, separately for the 68 feature point errors of the model and for the whole face MESH, the former to measure the face alignment pose errors and the latter to measure the face spatial geometry errors. The test set used was AFLW2000-3D.

4.1 Introduction to the experimental infrastructure

The basic information about the experimental platform used in this dissertation is as follows:

(1) Ubuntu16.04 Operating system

(2) I7-8700K CPU

(3) 32G RAM

(4) GPU is Titan X pascal

(5) The deep learning framework is tensorflow 1.5.0-GPU

(6) Python vision is 3.6.13

(7) 3D MESH using MeshLab

4.2 Training and Hyperparameters

In this dissertation, the initialization parameters of the coding and decoding layers are derived from the PRNet, which has been trained previously, and the Brief-Output is obtained from this structure. After freezing the parameters of the coding and decoding layers, the training of the residual network connect to Output is started, where the Output is obtained by summing the Delta-Output and Brief-Output. After the loss function of Output is stabilized, the parameters of the coding and decoding layers are unfrozen and trained together with the residual network to obtain the best

performance of the model in this dissertation.

The reason for initializing the parameters of the coding and decoding layers is that due to the large number of layers in the model, training requires larger computational resources and input data size, however, loading the previous model parameters can not only improve the fitting efficiency of the model, but also reduce the risk of gradient explosion and disappearance.

The reasons for stepwise training are firstly, the model structure of the algorithm designed in this dissertation is multi-channel, and it is difficult to balance the output weights of each channel in one training; secondly, it avoids gradient explosion and disappearance as much as possible; and finally, it is also because it can alleviate the overfitting phenomenon.

The training of hyperparameters was verified through repeated experiments, and the final parameters were set as follows: the batch size was set to 4, the initial learning rate was set to 0.001, the learning rate was optimized using Adam method, and the initial learning rate was reduced to half after every 5 epochs.

4.3 Experimental analysis of data enhancement

First, we performed the dataset augmentation validation using the PRN method, which provides a training parameter model based on the augmented dataset. The parameters were trained before and after the dataset enhancement, respectively, and the results were validated by the test set AFLW2000-3D. The specific results are shown in Figure 4.1.

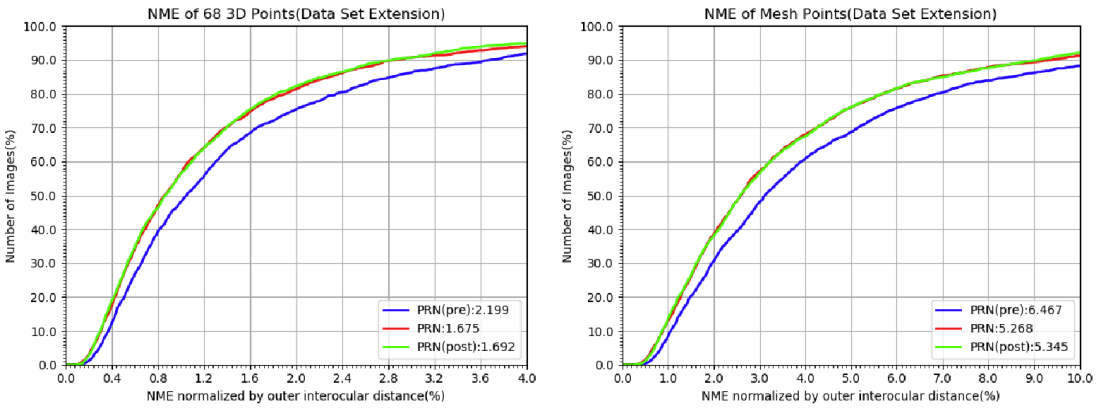

Figure 4.1 Data enhancement on the validation set for NME

In the left panel, the NME cumulative error distribution of 68 face feature points corresponds to the mean squared difference between the feature points of the generated face model and the standard data, and its normalisation criterion is the variance value of the two outer corner feature points of the standard data, the X-axis is the error normalisation percentage, and the Y-axis represents the number of samples included under this X error as a percentage of the total sample. In the left panel, red is the parametric model provided by the authors of PRN of the test results, blue is the 300W-LP without data enhancement, and green is the test results of our method. After data augmentation, our dataset performs similarly to that of the PRN authors and achieves 90% compliance with the test set at an error rate of around 2.8%, compared to 90% compliance with an error of less than 4% before augmentation. Their average errors were 2.199%, 1.675% and 1.692% respectively, a decrease in error of about 30%, thus showing that with data augmentation, the adaptability of the model can be greatly improved. The graph on the right shows the comparative error of 3D MESH, the data enhancement is not very different from the PRN algorithm training error, which is about 21% lower than the untreated error.

4.4 Experimental analysis of normalization

In the experimental process, there is a problem: Ground-Truth stores the XYZ information in 3D space, when we store the whole MESH in the 3D space of 0~255, so the value range of xyz is 0~255, the normalization operation is to divide it by 255 to get the data of 0~1, then in the training, use the sigmoid function to activate it, and finally output the data of 0~1, then multiply it by 255 to get the corresponding scale of 3D face model. The final output of 0~1 data is then multiplied by 255 to obtain a 3D face model of the corresponding scale. This processing will result in the accumulation of errors, but we store the MESH between -128~128 and activate it with the tanh function, this phenomenon is suppressed to some extent, and due to the symmetrical geometry of the face, the errors will not accumulate in one direction. As

shown in Figure 4.2, normalising Ground-Truth to the -1~1 space gives a slightly better cumulative NME error than the 0~1 space, so we chose to normalise to -1~1 and activate the output with the tanh function.

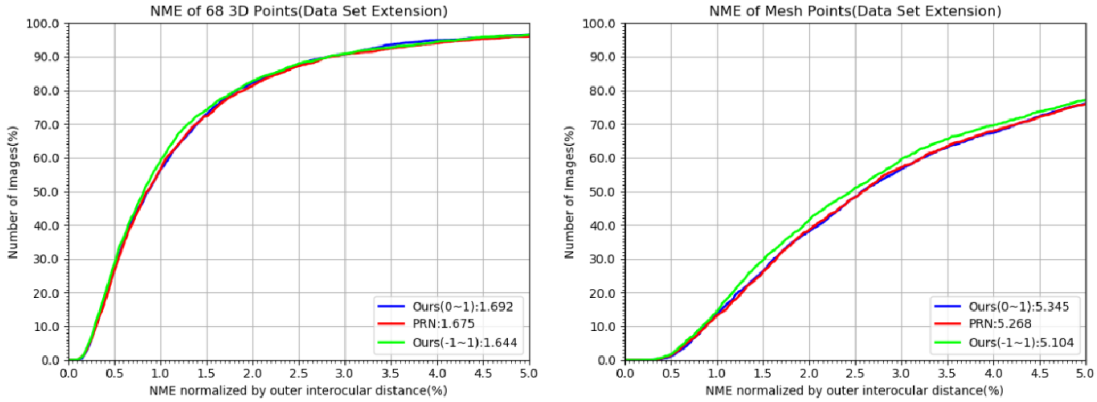

Figure 4.2 Normalisation of NME on the validation set

4.5 Experimenting with multi-stage training strategies and the necessity

In order to compare different network models for face reconstruction effects, we did three sets of training experiments, the first set was a network structure containing only Decoder-Encoder structure, we chose the PRN model and 3DDFA model, the second set was a PRN+ResNet structure, fixing PRN parameter variables and training ResNet separately, the third set was a PRN+ ResNet structure, without fixing the former parameters, for simultaneous training, After the LOSS was stabilized, we verified it with the AFLW2000-3D test set, and its error is shown in Figure 4.3. From the figure, we can see that fixing the PRN parameters and then inputting the results into ResNet for residual learning is instead inferior to the performance of the PRN network alone, while the PRN combined with ResNet for simultaneous learning is slightly improved, and we call it residual learning. The average cumulative error of MESH for PRN is 5.268, and the average cumulative error of MESH for residual learning is 5.104, with an overall effect improvement of about 3%. Looking closely at the left half of the two images, the error accumulation of residual learning is

significantly smaller than that of PRN throughout the dataset, with a maximum improvement of 7% for key point alignment and 19% for MESH error. Analysis through sample inspection in the test set shows that when the face information is richer, often residual learning is able to learn more detailed geometric information and has better mesh reconstruction capability.

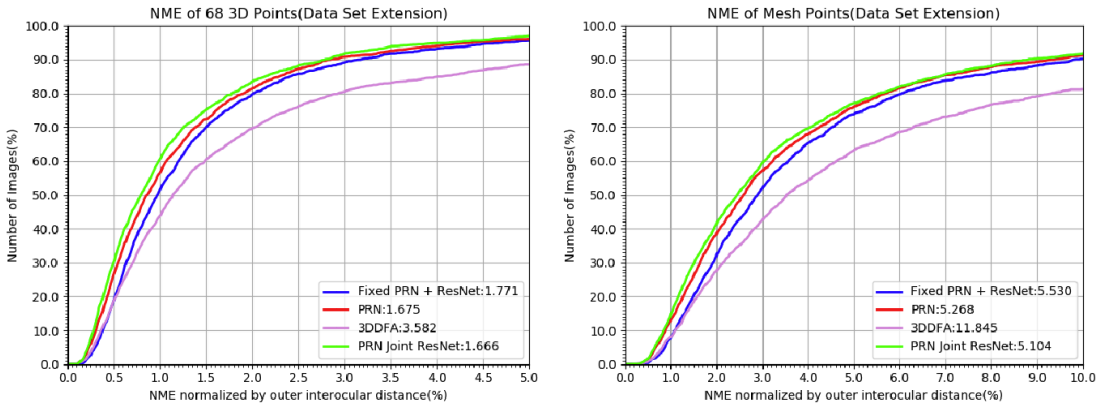

Figure 4.3 Multi-stage strategy for NME on validation sets

4.6 The necessity of experimenting with residual structures

The advantage of residual learning lies in its ability to learn details. In image processing such as 2D image coloring or color recovery, it is often possible to learn the direct difference information between an intermediate quantity and a standard quantity, and then by superimposing the difference information, it is possible to continue to learn more details of color features that cannot even be observed by the human eye, while end-to-end 3D reconstruction of the face is similar to image processing, where the input and output are images.

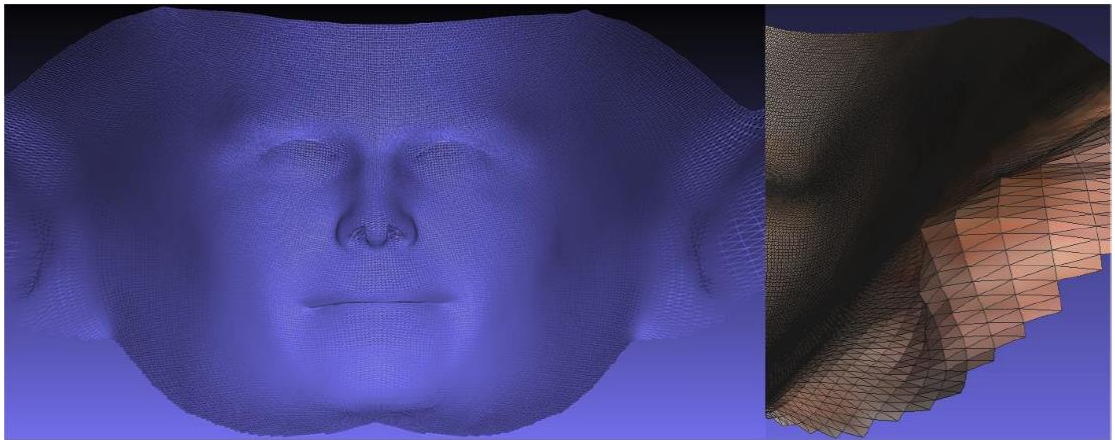

Figure 4.4 Standard grid effect

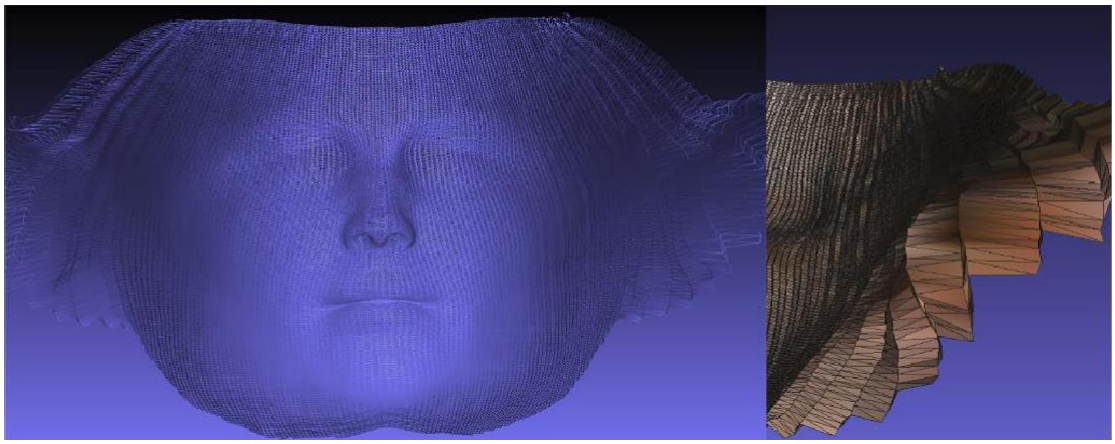

Figure 4.5 PRN grid effect

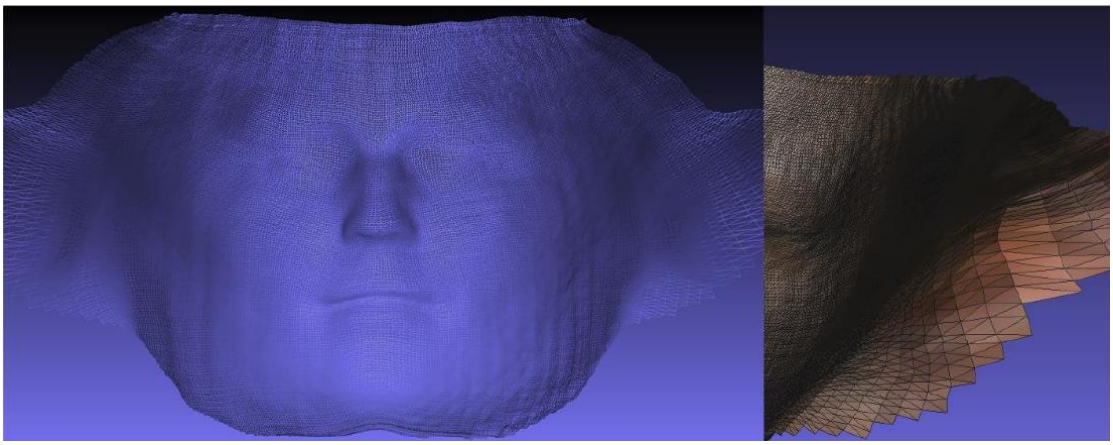

Figure 4.6 Residual learning grid effect

Figure 4.4 shows the standard MESH, in which the face surface is smooth with a uniform distribution of vertices, each triangular slice is similar in shape, has a gently shaped cheek region and a shapely nose region, and has a very uniform density of distributed vertices. Figure 4.5 shows the MESH generated by PRN for the same sample. Due to UV mapping, the 3D MESH of the face generated by this method

shows a linear jagged fold phenomenon in the local details of the face, although the whole generated model also fits the standard face model very well spatially, the vertical direction of the face shows a texture similar to a folding fan with unevenness. Figure 4.6 then shows the MESH generated by PRN+ResNet for the same samples, which is closer to the standard mesh with smooth surface and uniform distribution of vertices. Thus, by learning the difference between an intermediate quantity and Ground-Truth, more geometric features of the face mesh are obtained, which can improve the geometric quality of the mesh, both in terms of getting closer to the smoothness of the standard mesh and in terms of effectively suppressing linear textures. To quantify the smoothness of the MESH surface, Figure 4.7 was obtained by calculating the cosine of the angle between two adjacent triangles in the triangular index of the sample mesh.

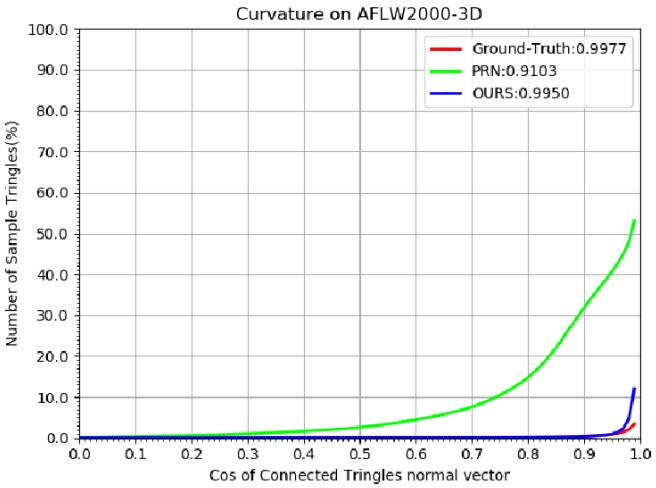

Figure 4.7 NME grid curvature

In Figure 4.7, the X-axis represents the cosine of the pinch angle, and the Y-axis represents the proportion of samples in which the logarithm of adjacent triangles is less than a certain cosine value, the closer the X-value is to 1, the smaller the pinch angle is. The MESH generated by the residual learning is almost close to the standard mesh, and 97% of its adjacent triangles are less than 15 degrees, thus it can be seen that the residual structure greatly improves the smoothness of the mesh and has the ability to learn the geometric properties of the mesh.

4.7 The necessity of LOSS function selection

Based on the geometric properties of faces, two geometric constraints are proposed in this dissertation, L_{kpt} is a symmetry constraint for face feature points to maintain a certain symmetry of the face as a whole, and L_{smooth} is to use the Laplace smoothing principle to constrain the smoothness of the face model. It was found that the inclusion of these geometric constraints may lead to abrupt changes in the model and affect the training of the model parameters, which may be due to the direct loss calculation of Ground-Truth vertices by MSE, where the whole Ground-Truth constrains each vertex element of the face model precisely. As shown in Table 4.1, comparative experiments show that L1 multiplied by the weighted mask is sufficient to obtain good training results.

Table 4.1 Performance comparison of different loss functions

	L_1	$L_1 + L_K$	$L_1 + L_S$	$L_1 + L_K + L_S$	L_1 * MASK
PRN	6.377	5.867	5.567	5.592	5.255
PRN + ResNet	4.435	4.256	4.121	3.911	3.912

4.8 Human face correction analysis

Due to the enhancement of the dataset and the improvement of the network structure by adding a residual structure, we have been able to obtain a better 3D model of the face, however, when we performed NME error analysis on the test set, we found that often when the error of 68 feature points was large, the error of MESH vertices was also large. When the face is flipped, the model still needs to be improved for such large pose test cases. Here, in a way, we separate face alignment and face reconstruction, i.e., face correction is performed first, and the faces are placed vertically so that the XYZ in 3D space do not interfere with each other, thus allowing the deep network to focus more on the learning of face depth information.

4.9 Conclusion

Summarizing all the experiments, after a face correction pre-processing process and our innovations and improvements, our final face 3D reconstruction cumulative error is shown in Figure 4.8, with an average cumulative error of 3.912, PRN algorithm error is 5.255 and 3DDFA algorithm error is 11.845. We achieved a 37% lower cumulative MESH vertex NME error than the current best algorithm, PRN, and a 33% reduction in the 68 key point alignment error. As shown in Figure 4.9, this is the final face reconstruction result

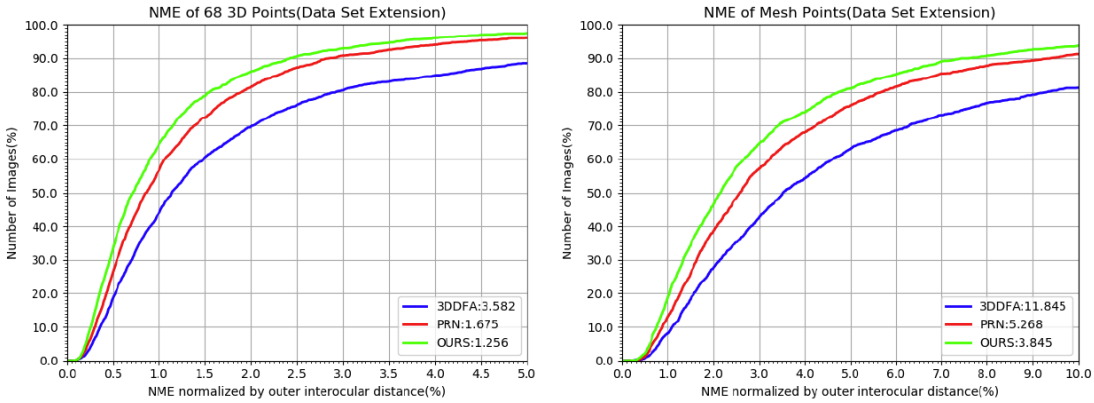

Figure 4.8 Face 3D reconstruction NME cumulative error

Figure 4.9 Three-dimensional reconstruction results

5. Discussion

This dissertation implements an end-to-end face 3D reconstruction system with a simple, fast, and easy process and the expected modelling quality. Extensive pre-processing work was carried out on the dataset 300W-LP to enhance the expressiveness of this face dataset. This dissertation proposes a 3D reconstruction method for faces based on a hierarchical residual learning network model. The face reconstruction process is improved to weaken the influence of the 3D spatial xy-axis on model training, thus focusing more on the generation of depth information, which ultimately greatly improves the quality of face 3D reconstruction. As the face models we generate are based on 3DMM, they can be used for a wide range of applications such as pose estimation, depth estimation, face editing, etc. depending on their characteristics.

We separate the recovery and reconstruction of face information, we transfer the reconstruction of information from the missing parts of the face to the restoration of 2D face information and restore the missing texture information of the face in advance, it is obvious that the 2D restoration is much less difficult than the 3D restoration. Similar to the restoration of images, the incomplete face texture caused by large poses and occlusions of the face is restored in advance for the face content, and the face information is complemented by neural networks or traditional methods, followed by 3D reconstruction of the face with complete face information.

Although the method in this dissertation is able to achieve millisecond face modelling speeds on a high-performance computer, its computational effort is still too large. In the future, we will make it better adapt to mobile terminals or embedded devices to improve the speed of face 3D modelling, and continue our research in terms of model quality, computational complexity, and construction speed.

References

1. Sagonas, C., Tzimiropoulos, G., Zafeiriou, S. and Pantic, M., 2013. 300 faces in-the-wild challenge: The first facial landmark localization challenge. In Proceedings of the IEEE International Conference on Computer Vision Workshops (pp. 397-403).
2. Zhu, X. and Ramanan, D., 2012, June. Face detection, pose estimation, and landmark localization in the wild. In 2012 IEEE conference on computer vision and pattern recognition (pp. 2879-2886). IEEE.
3. Belhumeur, P.N., Jacobs, D.W., Kriegman, D.J. and Kumar, N., 2013. Localizing parts of faces using a consensus of exemplars. IEEE transactions on pattern analysis and machine intelligence, 35(12), pp.2930-2940.
4. Zhou, E., Fan, H., Cao, Z., Jiang, Y. and Yin, Q., 2013. Extensive facial landmark localization with coarse-to-fine convolutional network cascade. In Proceedings of the IEEE international conference on computer vision workshops (pp. 386-391).
5. Koestinger, M., Wohlhart, P., Roth, P.M. and Bischof, H., 2011, November. 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) (pp. 2144-2151). IEEE.
6. Xiang, J. and Zhu, G., 2017, July. Joint face detection and facial expression recognition with MTCNN. In 2017 4th International Conference on Information Science and Control Engineering (ICISCE) (pp. 424-427). IEEE.
7. Kazemi, V. and Sullivan, J., 2014. One millisecond face alignment with an ensemble of regression trees. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1867-1874).
8. He, K., Zhang, X., Ren, S. and Sun, J., 2016. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).
9. Ahlberg, J., 2001. Candide-3-an updated parameterised face.

10. Booth, J., Roussos, A., Ponniah, A., Dunaway, D. and Zafeiriou, S., 2018. Large scale 3d morphable models. International Journal of Computer Vision, 126(2), pp.233-254.
11. Liu, F., Zeng, D., Zhao, Q. and Liu, X., 2016, October. Joint face alignment and 3D face reconstruction. In European Conference on Computer Vision (pp. 545-560). Springer, Cham.
12. Hansen, M.F., Atkinson, G.A., Smith, L.N. and Smith, M.L., 2010. 3D face reconstructions from photometric stereo using near infrared and visible light. Computer Vision and Image Understanding, 114(8), pp.942-951.
13. Kemelmacher-Shlizerman, I. and Basri, R., 2010. 3D face reconstruction from a single image using a single reference face shape. IEEE transactions on pattern analysis and machine intelligence, 33(2), pp.394-405.
14. Lee, S.Y., Chwa, K.Y. and Shin, S.Y., 1995, September. Image metamorphosis using snakes and free-form deformations. In Proceedings of the 22nd annual conference on Computer graphics and interactive techniques (pp. 439-448).
15. Beumier, C. and Acheroy, M., 1999, September. 3D facial surface acquisition by structured light. In International Workshop on Synthetic-Natural Hybrid Coding and Three Dimensional Imaging (pp. 103-106).
16. Rydfalk, M., 1987. CANDIDE: a parameterised face. Linköping Univ..
17. Blanz, V. and Vetter, T., 1999, July. A morphable model for the synthesis of 3D faces. In Proceedings of the 26th annual conference on Computer graphics and interactive techniques (pp. 187-194).
18. Jourabloo, A. and Liu, X., 2016. Large-pose face alignment via CNN-based dense 3D model fitting. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4188-4196).
19. Genova, K., Cole, F., Maschinot, A., Sarna, A., Vlasic, D. and Freeman, W.T., 2018. Unsupervised training for 3d morphable model regression. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 8377-8386).
20. Jackson, A.S., Bulat, A., Argyriou, V. and Tzimiropoulos, G., 2017. Large pose 3D face reconstruction from a single image via direct volumetric CNN regression. In Proceedings of the IEEE international conference on computer vision (pp. 1031-1039).
21. Feng, Y., Wu, F., Shao, X., Wang, Y. and Zhou, X., 2018. Joint 3d face reconstruction and

dense alignment with position map regression network. In Proceedings of the European Conference on Computer Vision (ECCV) (pp. 534-551).

22. DeCarlo, D., Metaxas, D. and Stone, M., 1998, July. An anthropometric face model using variational techniques. In Proceedings of the 25th annual conference on Computer graphics and interactive techniques (pp. 67-74).
23. Bruckstein, A.M., 1988. On shape from shading. Computer Vision, Graphics, and Image Processing, 44(2), pp.139-154.
24. Cao, C., Weng, Y., Zhou, S., Tong, Y. and Zhou, K., 2013. Facewarehouse: A 3d facial expression database for visual computing. IEEE Transactions on Visualization and Computer Graphics, 20(3), pp.413-425.
25. Wold, S., Esbensen, K. and Geladi, P., 1987. Principal component analysis. Chemometrics and intelligent laboratory systems, 2(1-3), pp.37-52.
26. Antonakos, E., Alabort-i-Medina, J., Tzimiropoulos, G. and Zafeiriou, S., 2014, October. Hog active appearance models. In 2014 IEEE International Conference on Image Processing (ICIP) (pp. 224-228). IEEE.
27. Cootes, T.F., Taylor, C.J., Cooper, D.H. and Graham, J., 1995. Active shape models-their training and application. Computer vision and image understanding, 61(1), pp.38-59.
28. Paysan, P., Knothe, R., Amberg, B., Romdhani, S. and Vetter, T., 2009, September. A 3D face model for pose and illumination invariant face recognition. In 2009 sixth IEEE international conference on advanced video and signal based surveillance (pp. 296-301). Ieee.
29. Davies, R., Twining, C. and Taylor, C., 2008. Statistical models of shape: Optimisation and evaluation. Springer Science & Business Media.
30. Tran, L. and Liu, X., 2018. Nonlinear 3d face morphable model. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 7346-7355).
31. Bookstein, F.L., 1989. Principal warps: Thin-plate splines and the decomposition of deformations. IEEE Transactions on pattern analysis and machine intelligence, 11(6), pp.567-585.
32. Tewari, A., Zollhofer, M., Kim, H., Garrido, P., Bernard, F., Perez, P. and Theobalt, C., 2017. Mofa: Model-based deep convolutional face autoencoder for unsupervised monocular reconstruction. In Proceedings of the IEEE International Conference on Computer Vision

Workshops (pp. 1274-1283).

33. Shrivastava, A., Pfister, T., Tuzel, O., Susskind, J., Wang, W. and Webb, R., 2017. Learning from simulated and unsupervised images through adversarial training. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2107-2116).
34. Abrevaya, V.F., Wuhrer, S. and Boyer, E., 2018, March. Multilinear autoencoder for 3d face model learning. In 2018 IEEE Winter Conference on Applications of Computer Vision (WACV) (pp. 1-9). IEEE.
35. Zhu, X., Lei, Z., Liu, X., Shi, H. and Li, S.Z., 2016. Face alignment across large poses: A 3d solution. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 146-155).
36. Long, J., Shelhamer, E. and Darrell, T., 2015. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3431-3440).
37. Kouris, A., Venieris, S. I., & Bouganis, C. S. (2018). CascadeCNN: Pushing the performance limits of quantisation. arXiv preprint arXiv:1805.08743.
38. Zhang, W., Ma, B., Liu, K., & Huang, R. (2017). Video-based pedestrian re-identification by adaptive spatio-temporal appearance model. IEEE transactions on image processing, 26(4), 2042-2054.
39. Milborrow, S., & Nicolls, F. (2008, October). Locating facial features with an extended active shape model. In European conference on computer vision (pp. 504-513). Springer, Berlin, Heidelberg.
40. Cristinacce, D., & Cootes, T. (2008). Automatic feature localisation with constrained local models. Pattern Recognition, 41(10), 3054-3067.
41. Felzenszwalb, P. F., Girshick, R. B., & McAllester, D. (2010, June). Cascade object detection with deformable part models. In 2010 IEEE Computer society conference on computer vision and pattern recognition (pp. 2241-2248). IEEE.
42. Guo, C., Li, C., Guo, J., Cong, R., Fu, H., & Han, P. (2018). Hierarchical features driven residual learning for depth map super-resolution. IEEE Transactions on Image Processing, 28(5), 2545-2557.