

A Comparative Study of Automatic Facial Landmark Detection

by

Ziyu Ye

A thesis submitted in partial fulfilment for the requirements for the degree of
MSc (by Research) at the University of Central Lancashire

June 2023

A Comparative Study of Automatic Facial Landmark Detection

RESEARCH STUDENT DECLARATION FORM

Type of Award MSc (by Research)

School University of Central Lancashire

Sections marked * delete as appropriate

1. Concurrent registration for two or more academic awards

Either *I declare that while registered as a candidate for the research degree, I have not been a registered candidate or enrolled student for another award of the University or other academic or professional institution

or ~~I declare that while registered for the research degree, I was with the University's specific permission, a *registered candidate/*enrolled student for the following award:~~

2. Material submitted for another award

Either *I declare that no material contained in the thesis has been used in any other submission for an academic award and is solely my own work

or ~~I declare that the following material contained in the thesis formed part of a submission for the award of:~~

(state award and awarding body and list the material below):

3. Collaboration

Where a candidate's research programme is part of a collaborative project, the thesis must indicate in addition clearly the candidate's individual contribution and the extent of the collaboration. Please state below:

4. Use of a Proof-reader

Either ~~*The following third party proof-reading service was used for this thesis _____ in accordance with the Policy on Proof-reading for Research Degree Programmes and the Research Element of Professional Doctorate Programmes.~~

~~_____ A copy of the confirmatory statement of acceptance from that service has been lodged with the Academic Registry.~~

or *No proof-reading service was used in the compilation of this thesis.

Signature of Candidate ZI YU YE

Print name: Ziyu Ye

Acknowledgements

I would like to express my gratitude to my primary supervisor, Dr Wei Quan, for providing me with constant and professional guidance, assistance, and encouragement during the last three years. He guided me into the area of computer vision as well as taught me his insights and hard-working attitude toward research. I feel very fortunate to have him as my supervisor.

I would like to express my deep thanks to my second supervisor, Professor Bogdan Matuszewski, for his helpful comments and feedback on my research and thesis. I also would like to thank my colleagues Jianyu and Edward in the Computer Vision and Machine Learning (CVML) group for their helpful advice on my work and all the great time that we shared.

I would like to thank my friends from inside and outside of the university for their support and encouragement, and I would certainly miss a lot of memorable time over the last three years without their accompany.

Most of all, I would like to sincerely thank my family, especially my parents, who always reminded me to be patient throughout my whole life. I appreciate everything they gave me and taught me. Thanks, must also go to my partner, and I wish I could show how much I appreciate her. Not only did she provide great advice on my thesis, but also, she inspired me on the presentation design. I gained a lot of inspiration and motivation from their love and immense support. This thesis would not have been written without them.

This research is one of the most memorable experiences I have ever had in my life, and I believe all these experiences and memories will be my most precious treasure in my future.

Abstract

Facial signs are associated with people's health and general fitness. Among different facial signs, the facial landmark is one of the essential appearances of facial characters, which can be linked with people's emotions, state of consciousness and health. Facial landmark detection can be used for recognising people's expressions, monitoring the conscious status of people's faces, or diagnosing neurological diseases. Recent advances in imaging technology and ever-increasing computing power have opened up the possibility of automatic facial landmark analysis and assessment. Facial landmark detection algorithms play an important role in facial analysis tasks, such as expression recognition, face swapping and medical auxiliary diagnosis. As a result, the accuracy of the facial landmark localisation directly impacts the reliability of facial landmark-based tasks.

The purpose of this project is to conduct a comparative study of existing vision-based methodologies for detecting facial landmarks and to identify appropriate ones that could overcome the challenges, such as pose variation and exaggerated expression. Three effective facial landmark detection methodologies were selected and implemented in this project, including Deep Convolutional Neural Network (DCNN) Cascade, Deep Alignment Network (DAN), and Stacked Dense U-nets (SDU). In order to provide a thorough evaluation, three publicly available datasets were used for the benchmarking, such as Multi-PIE, 300W and Menpo, which contain a large number of facial images with the variations in illumination, pose and expression as well as the occlusion. Through the evaluation based on different datasets, SDU was considered to have the best performance, and it was adopted and implemented into a real-time facial analysis system that contains landmark detection and assessment.

Contents

Acknowledgements.....	3
Abstract	4
Contents	5
List of Figures	8
List of Tables.....	12
List of Abbreviations.....	13
1.Introduction	15
1.1 Background	15
1.2. Aim of the Research	17
1.3. Research Contribution	18
1.4. Thesis Structure.....	19
2. Literature Review.....	20
2.1 Holistic Methods	21
2.1.1 Global Face Shape Model	22
2.1.2 Global Face Appearance Model.....	23
2.1.3 Active Appearance Model Search	24
2.1.4 Extensions of AAM.....	25
2.2 Constrained Local Methods	26
2.2.1 Face Shape Model.....	27
2.2.2 Local Face Appearance Model	28
2.2.3 Joint Optimisation Methods.....	29
2.3 Regression-Based Methods	29
2.3.1 Direct Regression Methods.....	30
2.3.2 Cascaded Regression Methods.....	31
2.3.3 Deep Learning-Based Methods	32

2.4 Summary of the three main categories	35
3. Datasets for facial landmark detection	37
3.1 CMU multi-pose, illumination and expression (Multi-PLE) face dataset	40
3.2 300W dataset	40
3.3 Menpo 2D dataset	41
3.4 Landmark configuration	42
4. Implementation of facial landmark detection algorithms	44
4.1 Deep Alignment Network (DAN) Implementation detail	46
4.1.1 Data pre-processing	46
4.1.2 DAN Model Training	48
Model Structure	48
Feed-forward neural network	50
Connection layers	53
4.1.3 DAN Loss Evaluation	55
4.1.4 DAN Model testing	56
4.2. Coarse-to-fine Deep Convolutional Neural Network (DCNN) Cascade Implementation detail	60
4.2.1. DCNN Model Training	60
Model Structure	60
Deep Convolution Neural Network (DCNN)	61
Facial attributes extraction	63
4.2.2 DCNN Loss evaluation	64
4.2.3 DCNN Model testing	64
4.3 Stacked Dense U-nets (SDU) Implementation detail	65
4.3.1 Data pre-processing	65
4.3.2 Model training of the SDU	65

Model Structure.....	65
Dense U-net.....	67
Residual Block	69
Inside Transformer	70
4.3.3 SDU Loss evaluation	71
4.3.4 Model testing of the SDU	72
5. Experiment	74
5.1 Training details of three algorithms.....	75
5.2 Evaluation Metric for facial landmark detectors	76
5.3 Comparison result of three facial landmark algorithms	79
5.3.1 Test on 300W dataset	79
5.3.2 Test on Menpo Dataset	86
5.3.3 Test on Multi-PLE Dataset.....	94
5.3.4 Summary of the three facial landmark algorithms	102
6. Conclusion and Future Work	103
References	105

List of Figures

Figure 1.1 A illustration of face variation (a) expression, (c) illumination, and (c) occlusion	17
Figure 2.1 Main categories of facial landmark detection Methods.....	20
Figure 2.2 A diagram of the AAM model	22
Figure 2.3 Learned shape variation from the AAM model	23
Figure 2.4 Learned appearance variation from the AAM model	23
Figure 2.5 Example of an original face image and a reconstructed face image	24
Figure 2.6 Illustration of CLM search algorithm.....	27
Figure 2.7 The architecture of a simple convolutional neural network.....	32
Figure 2.8 5 landmarks configurations.	33
Figure 2.9 The structure of the stacked hourglass network.....	34
Figure 2.10 Diagram of 3-D face model with the head pose parameter (roll, yaw, and pitch angles).....	35
Figure 3.1 Example of the Multi-PLE (frontal) datasets	40
Figure 3.2 Example of the LFPW dataset.....	40
Figure 3.3 Example of Helen’s dataset.....	41
Figure 3.4 Example of the AFW dataset.....	41
Figure 3.5 Example of the Menpo-2D (frontal) datasets.....	42
Figure 3.6 The illustration of the Multi-PLE 68 annotated landmarks configuration..	43
Figure 4.1 The general process of the facial landmarking system.....	45
Figure 4.2 Examples of intermediate and final results of data pre-processing	48
Figure 4.3 DAN model structure	49
Figure 4.4 Structure of a feed-forward neural network of a DAN stage	50
Figure 4.5 The diagram of fully connected layer regression	52
Figure 4.6 The diagram of the connection layers.	54
Figure 4.7 Example of input images into the second stage	55

Figure 4.8 The face detector workflow	56
Figure 4.9 Example results of Face detector	57
Figure 4.10 Face detectors' accuracy	58
Figure 4.11 Face detector results	58
Figure 4.12 The image transform	59
Figure 4.13 The prediction landmarks' result of DAN	60
Figure 4.14 The outline of the coarse-to-fine Deep convolution network cascaded .	61
Figure 4.15 Typical network structure of DCNN	61
Figure 4.16 Illustration of facial attributes extraction	63
Figure 4.17 The example of landmark prediction	64
Figure 4.18 A example image and results after data pre-processing.....	65
Figure 4.19 SDU model strcuture	67
Figure 4.20 Scale aggregation topology.....	68
Figure 4.21 Comparison of a regular 3×3 convolution and a 3×3 depth-wise separable convolution	69
Figure 4.22 Different residual-based architecture.....	70
Figure 4.23 The deformable convolution	71
Figure 4.24 Examples of the sampling location in 3×3 standard and deformable convolutions.	71
Figure 4.25 Example landmark prediction of the stacked dense U-net	73
Figure 5.1 The diagram of the performance evaluation.....	74
Figure 5.2 Ten-fold cross-validation workflow.....	75
Figure 5.3 Example of the assessed detection circle	78
Figure 5.4 300W dataset cumulative error distributions	79
Figure 5.5 Comparison of landmark error due to different facial attributes on the 300W dataset	80
Figure 5.6 Comparison of each landmarks error on the 300W dataset	81

Figure 5.7 Comparison of landmark error due to the different facial attributes on the 300W dataset with different expressions.....	82
Figure 5.8 Example of SDU poor landmark localisation	83
Figure 5.9 Comparison of the NME due to different facial attributes on the 300W dataset with different illuminations.....	84
Figure 5.10 Example of DCNN poor landmark localisation	84
Figure 5.11 Comparison of landmark error due to different facial attributes on the 300W dataset with heavy occlusion	85
Figure 5.12 The landmark localisation results on the 300W dataset with heavy occlusion.	85
Figure 5.13 Comparison of landmark error due to different facial attributes on the 300W dataset with frontal face	86
Figure 5.14 Menpo dataset cumulative error distributions.....	87
Figure 5.15 Comparison of landmark error due to different facial attributes of the Menpo dataset	88
Figure 5.16 Comparison of each landmarks error on the Menpo dataset.....	89
Figure 5.17 Comparison of landmarks error due to different facial attributes on the Menpo dataset with different expressions	90
Figure 5.18 Comparison of landmarks error due to different facial attributes on the Menpo dataset with different illuminations.....	91
Figure 5.19 Examples of SDU poor landmark localisation results.....	92
Figure 5.20 Comparison of landmarks error due to different facial attributes on the Menpo dataset with heavy occlusion.....	92
Figure 5.21 The poor landmark localisation results under heavy occlusion on the Menpo dataset with heavy occlusion.....	93
Figure 5.22 Comparison of landmarks error due to different facial attributes on the Menpo dataset with frontal face	94
Figure 5.23 Multi-PLE dataset cumulative error distributions	95
Figure 5.25 Comparison of each landmarks error on the Multi-PLE dataset.....	97

Figure 5.26 Comparison of landmarks error due to different facial attributes on the Multi-PLE dataset with different expressions.....	98
Figure 5.27 Examples of SDU landmark localisation	99
Figure 5.28 Comparison of landmarks error due to different facial attributes on the Multi-PLE dataset with different illuminations	99
Figure 5.29 Comparison of landmarks error due to different facial attributes on the Multi-PLE dataset with heavy occlusions	100
Figure 5.30 Examples of SDU poor landmark localisation	101
Figure 5.31 Comparison of landmarks error due to different facial attributes on the Multi-PLE dataset with frontal face	101

List of Tables

Table 2.1 Comparing three main categories of methods.	21
Table 3.1 Overview of the public 3-D face datasets	38
Table 3.2 Overview of the public 2-D face datasets	39
Table 4.1 Structure of the feed-forward network.....	53
Table 4.2 The coarse level's network size	62
Table 4.3 The refine level's network size	63
Table 5.1 The metrics' performance of the 300W dataset	80
Table 5.2 The statistical results on the 300W dataset with different expressions	82
Table 5.3 The statistical results on the 300W dataset with different illuminations.....	83
Table 5.4 The statistical results on the 300W dataset with heavy occlusion.....	84
Table 5.5 The statistical results on the 300W dataset with frontal face	86
Table 5.6 The metrics' performance of the Menpo dataset.....	87
Table 5.7 The statistical result on the Menpo dataset with different expressions	90
Table 5.8 The statistical results on the Menpo dataset with different illuminations ...	91
Table 5.9 The statistical results on the Menpo dataset with heavy occlusion	92
Table 5.10 The statistical results on the Menpo dataset with frontal face	93
Table 5.11 The metrics' performance of the Multi-PLE dataset	95
Table 5.12 The statistical results on the Multi-PLE dataset with different expressions	98
Table 5.13 The statistical results on the Multi-PLE dataset with different illuminations	99
Table 5.14 The statistical results on the Multi-PLE dataset with heavy occlusions.	100
Table 5.15 The statistical results on the Multi-PLE dataset with frontal face	101

List of Abbreviations

AAM	Active Appearance Model
AFW	Annotated Faces in the Wild
AUC	Area Under the Curve
BoRMaN	Boosted Regression and Graph Models based method
CAB	Channel Aggregation Block
CED	Cumulative Error Distribution
CLM	Constrained Local Method
CNN	Convolutional Neural Network
CSR	Cascade Shape Regression
DAN	Deep alignment network
DCNN	Deep Convolutional Neural Network
GPU	Graphics Processing Units
HOG	Histogram of Oriented Gradients
HPM	Hierarchical, Parallel and Multi-scale
IC	Inverse Compositional
LFPW	Labelled Face Part in the Wild
MSE	Mean Square Error
MTCNN	Multi-Task Cascaded Convolutional Networks
MTurk	Mechanical Turk
Multi-PLE	CMU Multi-Pose, Illumination and Expression
NME	Normalised Mean Error
PCA	Principal Component Analysis
ReLU	Rectified Linear Units
RGB	Red, Green, and Blue

RGB-D	Red, Green, Bule, and Depth
RLMS	Regularised Landmark Mean-Shift
RMSE	Root Mean Squared Error
RNN	Recurrent Neural Network
SAT	Scale Aggregation Topology
SCRFD	Sample and Computation Redistribution for efficient Face Detection
SDU	Stacked Dense U-nets
SIFT	Scale-Invariant Feature Transform
STD	Standard Deviation
SVM	Support Vector Machine
2-D	Two Dimensions
3-D	Three Dimensions

1.Introduction

1.1 Background

While the human face is an important medium that can directly reflect people's emotions, state of consciousness and health, facial Landmarks can be used to extract the features for facial analysis and assessment. The commonly used landmarks are at the eyebrow arcs, ear lobes, eye corners, nose tip, chin, and mouth corners.

With the recent development and enhancement of computer hardware, facial analysis has been receiving an increasing amount of attention for research, where facial landmark plays a vital role, such as expression recognition (Mohammad et al., 2022), face swapping (Ding et al., 2019) and medical auxiliary diagnosis (Ding et al., 2021). Here, three landmark-dependent tasks are detailed below:

- Expression recognition: Human facial expressions play an essential role in social communication because the channel of non-verbal communication and human emotions could support spoken to transform messages. (Michael et al., 2021). Fuzail (2020) utilized facial landmarks to calculate the Euclidean distances between the corner points in each organ, which can generate the input feature vectors of a neural network and then classify the six different emotions. Mohammad (2022) presents a lightweight algorithm to classify different emotions and recognise facial expressions in a real-time system. In this algorithm, the facial landmarks of each face need to be detected and then extract the input feature based on the angles in each attribute.
- Face swapping: This task refers to swapping a face with the face of another person or an animal object between the images or in a video. For example, we can select funny images and swap them with other images, or face swapping can be used to keep people private in the live stream. In OpenCV, face-swapping technology utilises facial landmarks to calculate the affine matrix to warp one face to the other face.

- **Medical Auxiliary Diagnosis:** Traditional medical auxiliary diagnosis refers to building a prior rule from the result of the invasive medical examination to aid doctors in diagnosis. For example, facial stroke is a common disease that can rely on the analysis of facial asymmetry in diagnosis. Compared with the traditional methods, automatic analysis of facial asymmetry provides the possibility of a non-invasive medical assessment to master primary disease indication and investigate the patient's status thoroughly. The key to achieving the automatic analysis of facial asymmetry based on two-dimensional photogrammetry can assess based on the angle and distance between the feature lines (Choi, 2015). Facial landmarks can be used to calculate the Euclidean distances and angles between the feature lines.

From the perspective of human vision, the task of identifying facial landmarks, such as the centre of an eye, is instinctive and natural. However, this simple task becomes challenging in the field of computer vision. The challenge is caused by the variability of facial appearances, which can be divided into two factors (Celiktutan et al., 2013). The challenge details that explain these influence factors are as follows:

The discrepancy of facial appearances can be divided into two parts: the intrinsic factor has a great deal of variability between individuals' facial shape, texture, and colour. The other one is the extrinsic factor which has several confounding factors, such as extreme illumination, exaggerated expression, partial occlusion, and the resolution of the camera. The current facial landmark detection algorithms still do not work well since facial landmarks may sometimes be partially occluded, such as hand movements, extreme head rotations or occlusions of hair. Furthermore, facial expression and illumination conditions are also the main variations, which lead to significant errors in facial detecting accuracy. As shown in Figure 1.1, the images demonstrate the faces' variations, consisting of different expressions, illumination, and occlusion conditions.

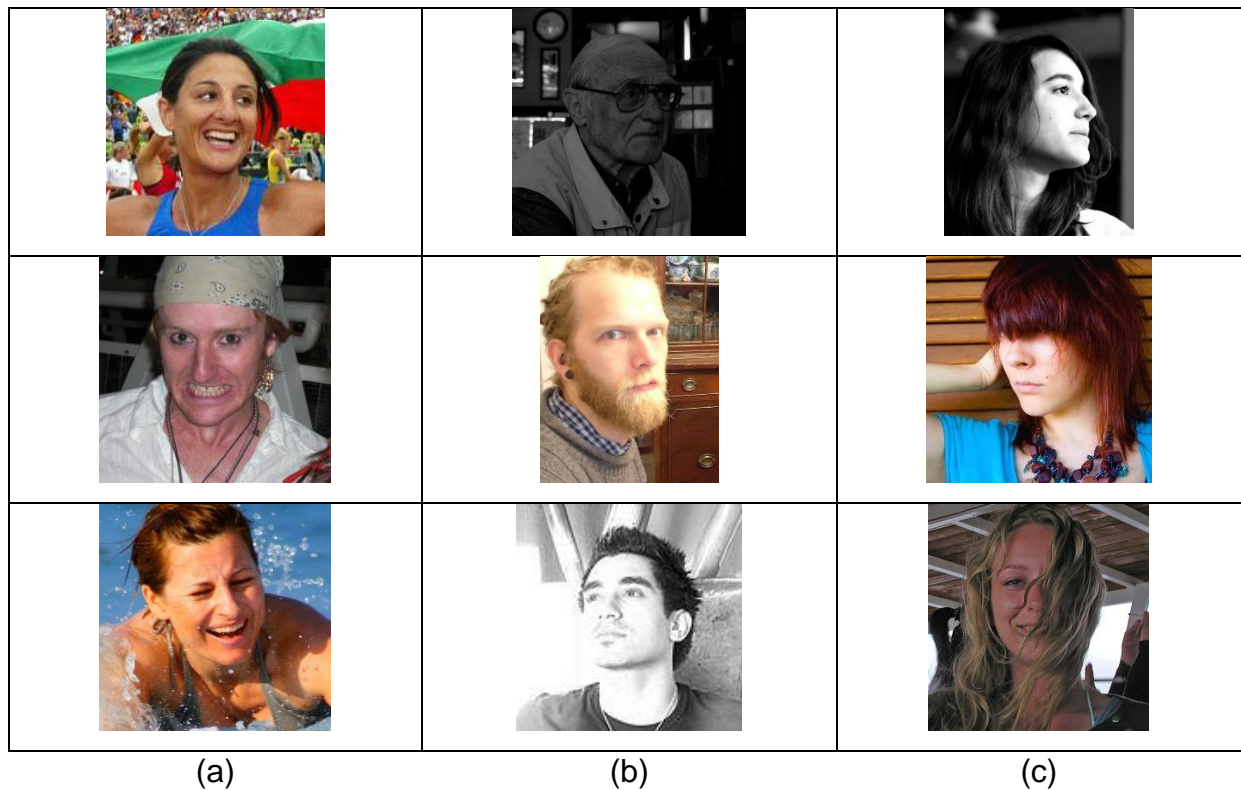


Figure 1.1 A illustration of face variation (a) expression, (c) illumination, and (c) occlusion

1.2. Aim of the Research

This research project aims to conduct a comparative study of the existing image-based algorithms of facial landmark detection and identify appropriate ones with robust performance that could overcome the challenge, including exaggerated expression and partial occlusion.

To be able to achieve the project goal, the following three stages need to be completed. Firstly, the classic facial landmark detection algorithms, publicly available datasets and appropriate landmark configurations should research and review, which can be used for this project. The available public datasets should have intrinsic factors, including various face shapes, textures, and colours, while they also need extrinsic factors, such as different expressions, illumination, and occlusion conditions. Then, according to the existing approaches for facial landmark detection, the widely used approaches are reviewed to investigate how to implement the three classic selected algorithms. Third, the three algorithms are compared based on the intrinsic and extrinsic challenges of the selected datasets.

1.3. Research Contribution

The original contributions of the research study provided by this thesis are summarised as follows:

- A review of the automatic facial landmarking methods can be divided into three categories: holistic methods, constrained local methods, and regression-based methods. The publicly available datasets are selected, including Multi-PIE (Gross et al., 2008), 300W (Sagonas et al., 2013), and Menpo (Deng et al., 2019), while the landmark configuration is called 'Multi-PIE' landmark configuration.
- Implementation of facial landmark detection methods based on three classic deep learning-based algorithms, including Deep Convolutional Neural Network (DCNN) Cascade (Zhou et al., 2013), Deep Alignment Network (DAN) (Kowalski et al., 2016), and Stacked Dense U-nets (SDU) (Guo et al., 2018).
- Comparative study between the three classic methods under publicly available datasets to analyse the intrinsic and extrinsic factors and a discussion of the main trends and current shortcomings in these methods for the utilisation in various applications of face analysis.

1.4. Thesis Structure

Chapter 1: an introduction of the whole project is provided, including the research background, research plan, aim of the research, and research contribution.

Chapter 2: a literature review is presented to discuss the classic facial landmarking methods in the three main categories, including the holistic methods, constrained local methods, and regression-based methods.

Chapter 3: a description of the relevant publicly available two dimensions (2-D) and three dimensions (3-D) facial datasets for facial landmark detection and a selection of the landmarks' configuration.

Chapter 4: implementation of three classic facial landmark detection algorithms, each of which comprises three stages, including data pre-processing, model training, and model testing.

Chapter 5: the training detail of each algorithm is introduced and is followed by the description of the performance metrics for the intrinsic and extrinsic factors. At last, a comparison result of three algorithms in each dataset is presented.

Chapter 6: a conclusion of the research study is stated, and the development of a real-time facial landmarking system can detect facial landmarks.

2. Literature Review

Facial landmarks detection algorithms can be applied to identify the landmark location using facial characteristics. Facial characteristics generally refer to face shape patterns and facial appearance. The face shape patterns are represented by a set of annotated coordinate points. The facial appearance is defined as texture variation, which observes the patterns of pixel colours or intensities across the face image or around the facial landmarks.

In the aspect of facial characteristics, the facial landmark detection algorithm is divided into three main categories of methods, including holistic methods, constrained local methods, and regression-based methods (Wu et al., 2018), as shown in Figure 2.1.

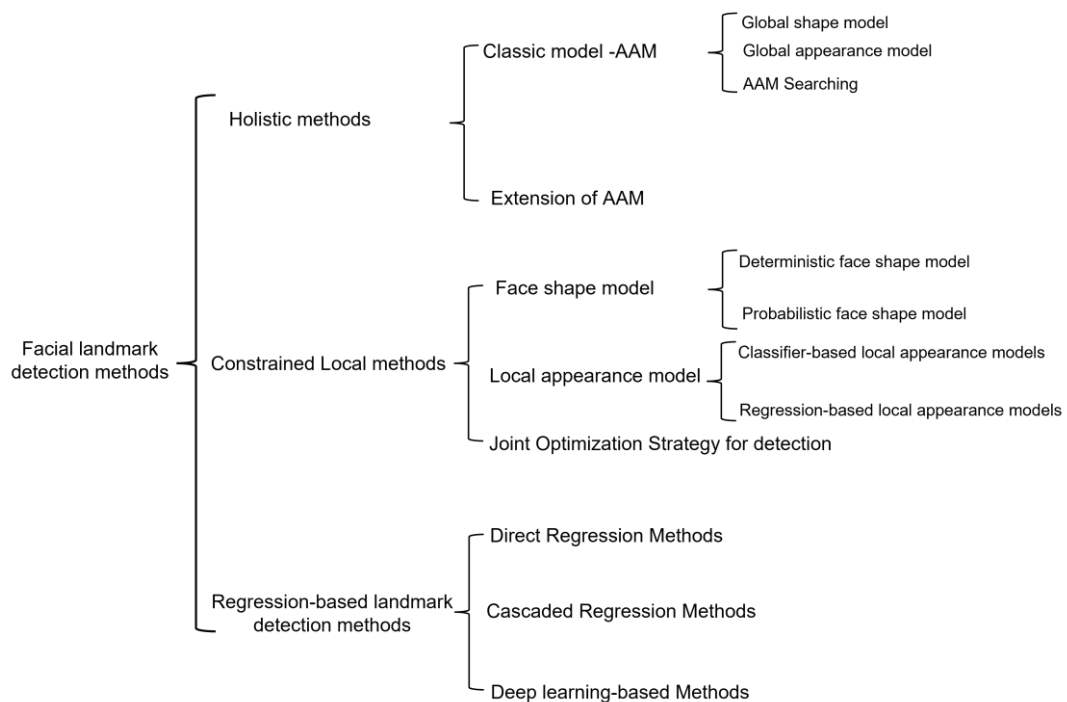


Figure 2.1 Main categories of facial landmark detection Methods

As shown in Table 2.1, there is an explanation based on their shape and appearance why the methods should be divided into three categories and compare their accuracy and the cost time of model inference. The holistic method adopts global shape patterns with explicit parameters and the whole facial appearance. The constrained Local Method (CLM) refers to the explicit holistic facial patterns and explicit local facial appearances. The regression-based method jointly utilises the appearance

information around the whole face region or local image patch and may implicitly embed the facial shape patterns with implicit parameters. In general, the regression-based methods have recently shown better accuracy in landmark detection, which will be described later.

Methods	Shape pattern	Appearance	Accuracy	Speed
Holistic Method	Parameter explicit	Whole face	Poor generalisation/Good	Slow
Constrained Method (CLM)	Local Parameter explicit	Local Patch	Good	Slow/Fast
Regression-based Method	Parameter implicit	Whole face / Local Patch	Good/Very Good	Fast/Very Fast

Table 2.1 Comparing three main categories of methods

2.1 Holistic Methods

The holistic methods are built up by explicit shape patterns and appearance information for facial landmark detection. One of the wide usages and most classic methods was the active appearance model (AAM) (Cootes et al., 2001) in holistic methods. Before understanding other facial landmark detection algorithms, it is essential to be familiar with the traditional holistic method. Furthermore, there is an extended discussion in the last.

Cootes et al. (2001) designed an active appearance model (AAM) to match the facial images, which explicitly utilised the whole facial shape variation and appearance variation. In Figure 2.2, a redrawn workflow of the AAM obtained from Cootes et al. (2001) is presented. Since it was a generative model based on statistics, there were a small number of coefficients to control both face shape and appearance. In processing model construction, AAM conducted a combinational model, including the global shape model and the global appearance model, using Principal Component Analysis (PCA) (Pearson et al., 1901). Then, the AAM search model updated the two models' parameters based on the image reconstruction error. In the detection processing, the position of facial landmarks was identified by matching the achieved

shape and appearance models to the testing images. There are a few steps for AAM to build the global shape model and the global appearance mode.

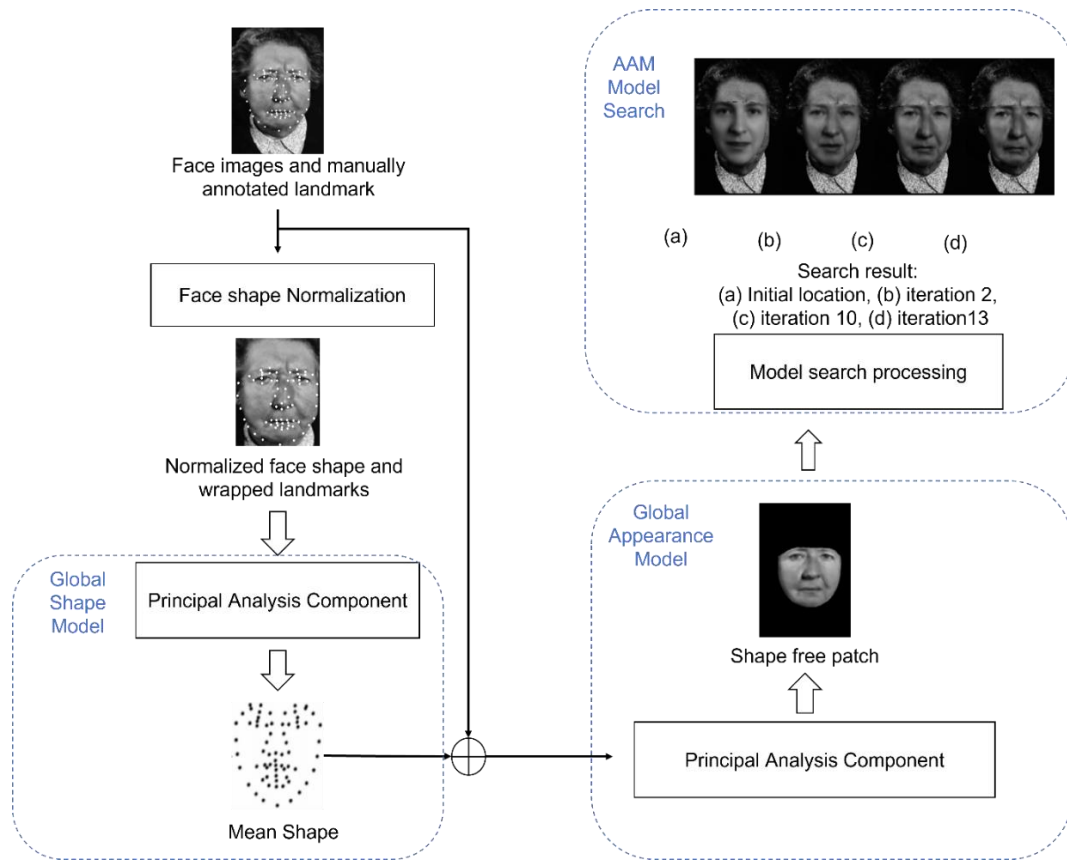


Figure 2.2 A diagram of the AAM model, redraw from Cootes et al. (2001)

2.1.1 Global Face Shape Model

For constructing a global shape model, it was required a set of landmark annotated images for training. In general, people must manually annotate landmarks on the images where corresponding landmarks are defined as the main features. Following image annotation, the next step is to align the set of landmarks by using Procrustes Analysis (Gower, 1975). The affine transformation was applied to generate the normalised face shape \mathbf{x}' based on the small displacements in scale, rotation, and translation. Then, the output is denoted by:

$$\mathbf{x}'_i = c\mathbf{R}_{2d}(\theta)(\mathbf{x}_i) + \mathbf{t} \quad (2.1)$$

where c is the factor of scale, $\mathbf{R}_{2d}(\theta)$ defined as the matrix of rotation, and \mathbf{t} is an in-plane translation.

Given the normalised face shape \mathbf{x}' , PCA was utilised to learn mean shape \mathbf{s}_0 , a set of orthogonal bases \mathbf{P}_s which can extract the shape variations, as shown in Figure 2.3. Based on the mean shape and a set of learned orthogonal bases \mathbf{P}_s , the normalised face shape could be represented by a set of shape coefficients \mathbf{b}_s .

$$\mathbf{x}' = \mathbf{s}_0 + \mathbf{P}_s \cdot \mathbf{b}_s \quad (2.2)$$



Figure 2.3 Learned shape variation from the AAM model, adapted with Cootes et al. (2001)

2.1.2 Global Face Appearance Model

For constructing a global face appearance model, each training image must be wrapped so that the landmarks can match the mean shape. The normalised facial appearance image is denoted as $L_i(W(\mathbf{x}'))$, where $W(\cdot)$ indexes the warping operation. The normalised facial appearance image identifies the image region of the normalised face shape, which is covered by the mean shape. Afterwards, PCA was used again to learn the mean appearance $\bar{\mathbf{g}}$ and a set of appearance orthogonal bases \mathbf{P}_g from the normalised facial appearance image, as shown in Figure 2.4. According to the facial appearance vectors \mathbf{P}_g , the mean appearance, any normalised facial appearance image can be represented by the appearance coefficients \mathbf{b}_g as:

$$L_i(W(\mathbf{x}')) = \bar{\mathbf{g}} + \mathbf{P}_g \cdot \mathbf{b}_g \quad (2.3)$$



Figure 2.4 Learned appearance variation from the AAM model, adapted with Cootes et al. (2001)

2.1.3 Active Appearance Model Search

In the previous section, the two models can obtain the shape coefficients \mathbf{b}_s and appearance coefficients \mathbf{b}_g by applying (2.2) and (2.3), which can be used to represent any example of mean shape and appearance. Furthermore, there is an optimal model that can apply PCA to learn the correlations between the shape coefficients \mathbf{b}_s and appearance coefficients \mathbf{b}_g .

During the process of landmark detection, AAM calculates the shape coefficients \mathbf{b}_s and appearance coefficients \mathbf{b}_g to match the reconstructed image as closely as possible, which can determine the position of landmarks. The matching process is an optimisation problem, which can be formulated by minimising the difference between the normalised face-testing image \mathbf{p}' and the reconstructed image $W(\mathbf{p}')$. The error distance ΔI can be minimised,

$$\Delta I(\mathbf{b}_s, \mathbf{b}_g) = Diff(\mathbf{p}', W(\mathbf{p}')) \quad (2.4)$$

Model coefficients are then updated prediction in an iterative manner, which computes the error distance based on the two current models' coefficients.

$$\mathbf{b}_s^*, \mathbf{b}_g^* = arg \min_{\mathbf{b}_s, \mathbf{b}_g} \Delta I(\mathbf{b}_s, \mathbf{b}_g) \quad (2.5)$$

Certainly, AAM can search for the most suitable shape and appearance coefficients \mathbf{b}_s and \mathbf{b}_g . As shown in Figure 2.5, Edwards et al. (1998) presented the image within the reconstructed face on the left side as the image within the original face is shown on the right side.



Figure 2.5 Example of an original face image and a reconstructed face image, adapted with Edwards et al. (1998)

2.1.4 Extensions of AAM

In the aspects of feature representation, AAM methods had other extensions. It is obvious to know that the AAM model had the lack ability to generalisation since the AAM model was difficult to match unseen face variations, including illumination, occlusion, etc. (Gross et al., 2004; Cross et al., 2005). The pattern of raw pixel intensities or colours that are used as features is one of the main reasons for the limitation. In order to solve this problem, some researchers employ more robust image features. For example, Hu et al. (2003) used wavelet features to construct the facial appearance model instead of raw pixel intensities. Furthermore, instead of the global appearance, the local appearance features were adapted to improve the robustness to illumination and occlusion. Jiao et al. (2003) used the Gabor wavelet with the Gaussian Mixture model to construct the local appearance features, which could search local points faster.

In addition, given the test image, the searching procedure is usually very slow because each iteration needs to calculate both shape and appearance coefficients $\mathbf{b}_s, \mathbf{b}_g$, by using Jacobin and Hessian matrices (Jones et al., 1998). To solve this problem, Baker et al. (2002) presented a popular algorithm, the Inverse Compositional (IC) algorithm. IC (Baker et al., 2002) warped back onto the coordinate of the reconstructed image by minimising the squared error between the input and reconstructed images. Firstly, the algorithm estimates the input image and reconstructed image to compute the warped coefficients by using the gradient descent algorithm. Then, the warped shape coefficients \mathbf{b}_s and appearance coefficients \mathbf{b}_g can jointly update until the error converges. Moreover, Gross et al. (2005) provided that PIC is computationally expensive but more robust.

However, a single AAM model may not give a high performance due to the linear relationship between the face shape and appearance. There were many different methods to solve the problem using ensemble models. One particular direction was to use the ensemble model that includes independent ensemble AAMs and coupled sequential AAMs to improve accuracy. The independent ensemble AAMs employed the different perturbed model coefficients in independent models. Compared with independent ensemble AAMs, the coupled ensemble AAMs generated the perturbed

data in the later training level based on the learned prediction model of the first few levels. For instance, Patrick Sauer and Taylor (2011) presented the sequential regression AAM model. The model utilised a series of AAMs to construct a sequential model for matching in a cascaded manner. In the first few levels, the model paid attention to the large variations (e.g., expression, pose) while those match the small variations in the next level. In this further work, Saragih et al. (2009) and Sauer et al. (2011) used the coupled ensemble model, which formulates the searching problem as an optimisation problem with a stochastic gradient descent solution. It can learn the linear model from the training data in the first few iterations and then update the perturbed model coefficients for the next iteration in a cascaded manner. There are different training data at different levels, which Saragih et al. (2009) and Sauer et al. (2011) applied to be avoided to trap in local minima. All methods mentioned above can improve the accuracy of the AAM method.

2.2 Constrained Local Methods

Compared to the holistic methods, the face characteristics in the CLM methods referred to the global shape pattern and the local appearance, which can be more robust to occlusion and illumination. CLM (Cristinacce and Cootes, 2006; Saragih et al., 2011) predicted the position of landmarks, including two main components. The first component was the face shape model, which captured the global shape pattern. The second component was the local appearance model, which used an independent local appearance around each landmark within the image, followed by an optimisation strategy to update landmark locations. Hence, CLM can utilise local appearance variation to replace global appearance variation, which can overcome the drawback of holistic methods, such as the sensitivity of the lighting and the effect of ambiguity.

For instance, the CLM search algorithms are shown in Figure 2.6 (Cristinacce and Cootes, 2006, p.5). The global face shape model and local appearance model are used to generate a set of CLM templates by the initial points. Then the optimisation strategy is utilised to update the shape parameter until the points converge. The two models were built to train the dataset separately while they jointly predicted landmark locations during landmark detection. Sections 2.2.1, 2.2.2, and 2.2.3 will discuss

each component and how to predict landmark location through the optimisation strategy jointly.

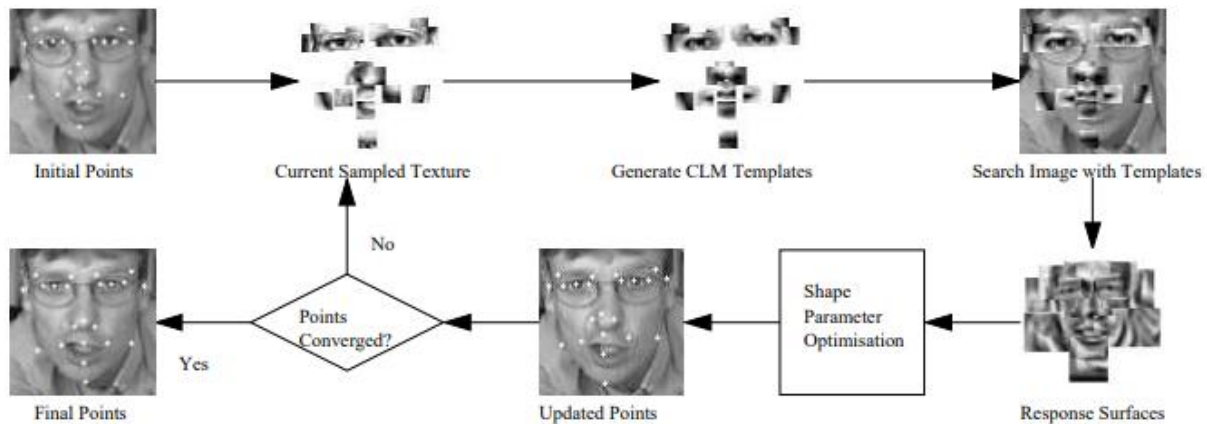


Figure 2.6 Illustration of CLM search algorithm, adapted with (Cristinacce and Cootes, 2006, p.5)

2.2.1 Face Shape Model

The face shape model captures information on global face shapes, which can constrain the landmark location search. There are two directions: the first direction is a deterministic method, and another is a probabilistic method.

In the deterministic view, the face shape model can capture the face shape, which assigns low errors to regular face shapes and penalise irregular face shapes in a global sense. Zhu and Ramanan (2012) utilised a pose-dependent tree structure to build a face-shape model that can capture the local non-linear relationship in which each node is an independent landmark. Hsu et al. (2015) improved the method based on Zhu and Ramanan (2012) to construct two levels tree structure that can use different numbers of landmarks with different resolutions. However, Baltušaitis et al. (2012) do not use 2-D facial shapes, and they embed facial shape patterns into the 3-D deformable model, which could overcome the head pose and expression variations. The method learns the coefficient of the 3-D deformable model and the parameter of head pose separately during training.

In the probabilistic view, the face shape model assigns higher probabilities to feasible face shapes that satisfy the anthropological constraint. Valstar et al. (2010) and Martinez et al. (2013) proposed a generative Boosted Regression and Graph Models based method (BoRMaN) based on the Markov Random Field. Each node

represents the corresponding positions of three points, and the method can model the combined relationships for all landmarks. In Wu et al. (2013) and Wu and Ji (2015), a discriminative deep face shape model constructed based on the Restricted Boltzmann Machine model is proposed. The facial shape can be divided into the head pose and expression-related parts, which can explicitly overcome handle head pose and expression variation. Compared to the holistic facial shape model, it is better to handle facial expressions and pose. However, facial occlusion is still an unsolved issue.

2.2.2 Local Face Appearance Model

The local face appearance model learns local appearance information based on the searched patches from the face shape model and can be divided into two categories: classifier-based local appearance models and regression-based local appearance models.

In the classifier-based local appearance models, they use classifiers to discriminate the positive patches that are located at the centre of landmark locations and negative patches that are far away from the centre of landmark locations. Different features descriptor and classifiers are used. The original CLM (Cristinacce and Cootes 2006) applied the raw image patch to build a classifier, while Zhu and Ramanan (2012) used the Histogram of Oriented Gradients (HOG) feature descriptor. Moreover, Belhumeur et al. (2011) used SIFT (Scale-invariant feature transform) feature descriptor and Support Vector Machine (SVM) classifier to learn the local appearance model, while Cristinacce and Cootes (2007) used the Gentle Boost classifier.

In the process of training, regression-based local appearance models are used to estimate the offset vector, which is the error distance between the position of any pixel and the position of the landmark in the searched patches using regression models. During detection, the regression model can be used for the prediction of offset value at a different location in the patches, which is added to the current landmark's location to calculate landmark prediction. Different features descriptor and regressors are used. Cristinacce and Cootes (2007) employed the Gentle Boost as the regressor. Furthermore, Cootes (2012) extended the method and used the

random forests as the regressor, which was able to demonstrate an increase in performance. Valstar et al. (2010) selected the Adaboost feature descriptor and SVM regressor to learn the regression function.

Both Classifier-based and Regression-based methods have a common drawback: it is unclear which feature and classifier or regressor to employ. In addition, the prediction results of the regression-based local appearance model may have a significant error when the positions of the current landmarks are far away from the target because the model performs a one-step prediction.

2.2.3 Joint Optimisation Methods

Having the face shape model and local appearance model introduced above, CLMs jointly infer landmark locations during landmark detection. One issue with the local appearance model was that it could not be directly analysed and computed using the learned facial feature. For this reason, it would cause the solution to trap into the local optimum. The joint optimisation strategy aims to solve the issue.

In order to tackle the challenge, there are some optimisation methods to find a strategy to combine the prediction result from the local patches of each point with the local face shape model, which can join infer. Saragih et al. (2011) proposed a Regularised landmark mean-shift (RLMS) to determine the probability of each prediction result within a patch. Furthermore, the author compared other optimisation methods, such as the Isotropic Gaussian Model (Cootes et al., 1995) and Gaussian Mixture Model (Gu and Kanade, 2008).

2.3 Regression-Based Methods

The regression methods infer the landmark coordinates using the mapping from the facial appearance to the landmark positions. It differs from the holistic or constrained local methods, which explicitly utilise the pattern of face shape and appearance variation to generate the most probability or determinacy of the synthetic face. Instead of the two methods, regression-based methods implicitly embed the constrained face shape, showing superior performance. Furthermore, the performance has been increased significantly by the cascade manner and the use of

Convolutional Neural Networks (CNN). Hence, the regression-based methods of landmark detection are into three categories: direct regression methods, cascaded regression methods, and deep Learning-based methods.

2.3.1 Direct Regression Methods

The direct regression methods can infer the landmark location in one step, which learns the direct mapping from facial appearances to the landmark location, without any pre-processing step for landmark location. The methods can further be categorised into local and global approaches in terms of shape and appearance. Since the local approaches employ the local patch, the global approaches utilise the global facial appearance.

The local approaches extract different facial regions as the local patches and perform the offset vectors based on the regression models. The vectors need to add the location of the current local patch to present all landmark positions jointly. The local approaches predict all points simultaneously since it differs from the local appearance model in the CLM that predicts each point independently. For instance, Dantone et al. (2012) used conditional regression forests to learn a mapping from the random extracted patches in the face region to update the face shape. Therefore, the performance was impacted by the attribute's prediction. There was a significant drawback with local regression methods: global shape estimation may not convey all feature information from the extracted local patches in the occlusions condition.

The global approaches directly learn the mapping from the global image to the landmark position. Compared with the local approaches, the whole shape and appearance convey more information for landmark estimation. However, it is more challenging to learn facial features information from the global facial appearance to landmark location because of dramatic variation in the global facial appearance. Sun et al. (2013) and Zhang et al. (2014) used deep-learning methods to conflict the problem of learning facial features from the global facial appearance, which we will introduce in Section 2.3.3.

2.3.2 Cascaded Regression Methods

Compared with direct regression methods that build without pre-processing step, Cascaded regression methods convert the initial landmark location (e.g., Mean face shape), and they update the landmark location with different regression functions in different stages. In the sequence of regression functions, the model could learn the facial features from shape-index appearance to the landmark location, where local appearance is sampled from the current landmark. The learned model parameters in the early stage can employ to update the training data in the next stage.

Different shape-index appearances and regression models are used. Cao et al. (2014) applied the pixel intensity of the shape index to facial features, which were defined as the relative position to the current landmarks. In addition, Ren et al. (2014) independently applied the regression forests to learn their binary feature within each landmark. During training, a linear regression function was utilised to convert the binary feature for learning the joint facial features from shape-index appearance to shape updates. Furthermore, Kazemi and Sullivan (2014) used regression trees as the model for face alignment. Among different functions, a parallel method of cascaded linear regression in Asthana et al. (2014) gained superior performance, which only relied on the statistics information from the previous level. From the above methods, Deng et al. (2015) showed the cascade regression method can improve the performance based on multi-component, including multi-view, multi-scale, and multi-component. The method employed six-component deformable models as a facial detector to classify and group the training shape into three categories, including left, right, and front profile views, which reduced the perturbation of each view set.

However, there is an issue with the cascade manner, which must be a fixed number of cascaded stages. We cannot change the cascaded stages or judge the landmark prediction stability for different images in the testing stages. Therefore, it is a case of prediction trapped in local minima, and the iteration continues. In other cases, it is possible that the prediction is near the optimal solution, but we still do not know when to stop the iteration best.

2.3.3 Deep Learning-Based Methods

In last ten years, the use of deep learning methods has gained more popularity in the field of computer vision; what is more, it is a trend to change the traditional statistics methods to deep learning-based methods in the aspect of facial landmark detection or tracking. Another case of deep learning methods becoming a popular tool is many publicly available datasets and a wide range of high-performance and computational hardware such as graphics processing units (GPU). Hence, CNN is the most influential model for facial detection, landmark detection, or recognition, which also follow the principle of direct regression or cascade regression.

A perceptron is a simple basic form to explain the convolution neural network within the convolution operation: $Q = I * W + b$ where I are the input images by corresponding a set of initial weights W and basis b (Goodfellow et al., 2016), as shown in Figure 2.7. Hence, the backpropagation algorithm (Hecht-Nielsen, R., 1989) can be applied to optimise W, b by calculating the error from the given loss function. Thus, the convolutional layer forms a set of automatic feature extractors which can implicitly learn facial features from global or local shapes and appearances within facial images. Compared with the other two methods, CNN can directly predict the location of landmarks based on facial images rather than utilise manually handcrafted features such as HOG and SIFT. In addition, CNN is powerful for improving the model's performance while more data has been added to the training sets.

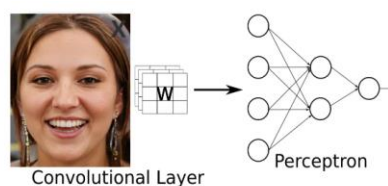


Figure 2.7 The architecture of a simple convolutional neural network

In the early work of deep learning, the cascaded coarse-to-fine structure was commonly used to design facial landmark detection, where several different CNNs levels are linked sequentially to predict the final landmarks. Sun et al. (2013) built up three-level of structures in a cascaded manner that the CNNs of each level predict

five facial landmarks using a facial image, and the configurations are shown in Figure 2.8.



Figure 2.8 5 landmarks configurations.

Sun et al. constructed three levels of CNNs, which produced the approximate position of five facial landmarks. The first level input an image of the bounding box region, which involves the whole facial. Thus, the CNNs of the first level predicted each landmark within the image of corresponding regions. The mean value of each prediction landmark position was calculated since the CNNs predicted each landmark two or three times. Then, the coarse prediction can be redefined in the local image patch as the input of the convolutional stage of the second level and the third level. The CNNs' input of the second level is the patches of the eyes and noses, and the input of the third level is the patches of the noses and mouths. The refined prediction was computed by multiple landmark positions produced from the CNNs in the second and third levels.

Later, there were two directions to improve the design of facial landmark detection models. The first direction was to improve the design of the cascaded manner. In the work of Zhou et al. (2013) and Fan, Zhou (2016), the three-level cascaded CNN model was somewhat similar to the method (Sun et al., 2013), both being produced with local facial patches to train and thus refining each individual landmark locally. In contrast with Sun, Zhou constructed to predict more landmarks employing 68 landmarks instead of 5 in a coarse-to-refine manner. In addition, Zhang et al. (2014) searched the cascaded manner by using the deep auto-encoder model. Compared with training CNNs in a cascaded manner, Trigeorgis et al. (2016) constructed a

deep Recurrent Neural Network (RNN) to build an end-to-end facial landmark detection mode which can mimic the cascaded manner and append time constraints. The second direction was to investigate different methodologies or network designs of facial landmark detection to improve performance. Zhang et al. (2014, 2016) and Ranjan et al. (2016) described multi-task learning for facial landmark detection, in which sharing the same facial features and the relationships of multi-tasks could encourage an increase in the performance of the task. For instance, Zhang et al. (2014) utilised facial attributes jointly, such as head pose, gender, and facial expression, to train a deep CNN, which was shown an advance in accuracy when the model transfer to predict dense landmarks. In the work proposed by Randjan et al. (2016), a similar framework was constructed to implement face detection, landmark detection, pose estimation and gender recognition, which jointly employs coarse and fine feature representations from multiple convolution layers. In addition, Yang et al. (2017) proposed the stacked hourglass convolution network, in which keeping a feature representation copy of output after the convolution layer to avoid losing the appearance information, as shown in Figure. 2.9. The network can reduce the height and width of an image while increasing the depth as well as a traditional CNN. It can also return the original values of the width, height, and depth from the feedforward copy of layer output.

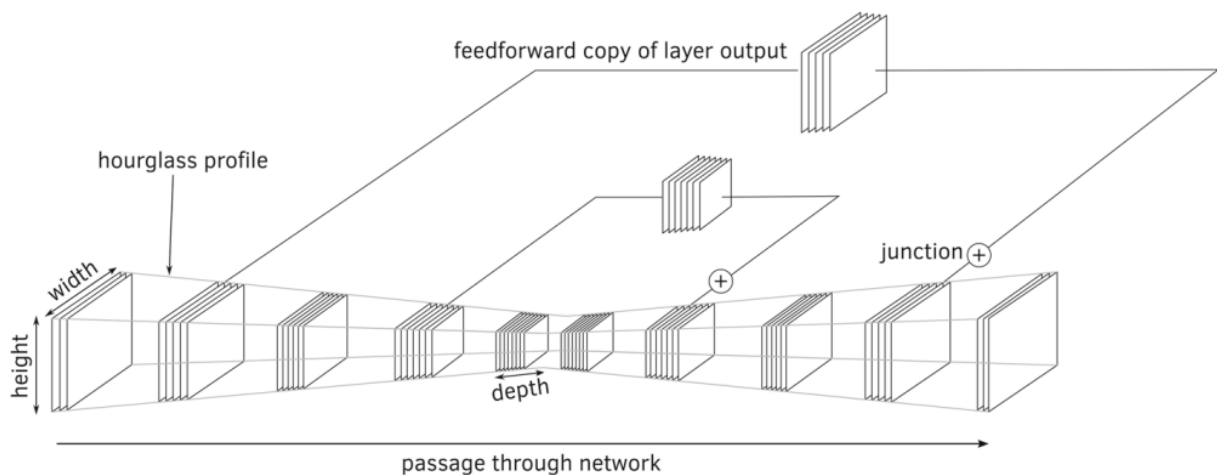


Figure 2.9 The structure of the stacked hourglass network, adapted with Yang et al. (2017)

In the past six years, some work has improved the CNNs' power by applying the facial shape and appearance from the 3-D vision, which produces 3-D shape deformable model coefficients and 3-D head poses. As shown in Figure 2.10, the 3-

D head pose shows more information, such as roll, yaw, and pitch angle; therefore, the computer vision projection model can determine the location of 2-D landmarks with more accuracy. Instead of directly predicting 2-D landmarks in CNN, the 3-D shape deformable model and 3-D head poses are better at handling pose variation and facial expression. What's more, the coefficient parameters and shape constraints are explicitly embedded in the final prediction. For instance, Zhu et al. (2016) constructed a dense 3-D face shape model in a cascaded manner. Hence, a cascaded framework with CNN models was employed to iterate the coefficient of the 3-D facial model and the parameters of the 3-D head pose in each iteration. Then, the 3-D shape is constructed and projected to 2-D as the input of the CNN model to regress the prediction. It can still boost performance by adding more data or utilising multi-task learning (Ranjan et al. 2016).

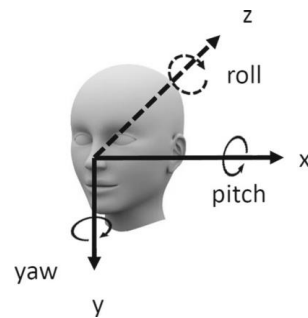


Figure 2.10 Diagram of 3-D face model with the head pose parameter (roll, yaw, and pitch angles)

Compared with different regression-based methods, cascaded regression methods are better than direct ones. What is more, the deep learning-based methods can show an increase in performance when applying the model in a cascaded manner. Furthermore, the regression-based methods do not need to use explicit parameters to build a face shape model, while the face shape patterns implicitly are embedded in the model. The drawback of the regression-based method is that the quality of the initial face region in the image is sensitive to learning facial features from the facial appearance. Therefore, the model could not train the high performance if the initial face region is offset. Sagonas et al. (2016) have researched the drawback.

2.4 Summary of the three main categories

Based on the reviewed paper by Wu et al. (2018), we presented the three main categories of facial landmark detection methods: the holistic methods, the CLM

methods, and the regression-based methods. Furthermore, we explore and provide a more detailed review of the methodologies of deep learning algorithms to enhance the comprehensive understanding of this field.

In terms of utilising features, the global shape patterns using explicit parameters are adopted to construct face shape model coefficients as in the holistic methods and CLM. CLMs have a significant improvement over the holistic methods in that they use the local patches and appearances around the landmarks as facial features. Compared with the global shape pattern, the shape and appearance model using the local patches are more robust under ‘in-the-wild’ conditions, such as facial occlusion and illumination. However, the holistic methods or CLM always match the synthetic face by updating the model coefficient in an iterative manner since the large face offset may be caused by small model coefficient errors.

Compared to the holistic methods and CLMs, the regression-based methods are more promising by learning facial features from shape-indexed features. As the jointly predicting the shape, the regression-based model can embed the implicit face shape pattern constraint and employ the different regression functions in a cascaded manner instead of using the explicit face shape model. As a directly predicting, the landmark locations can be more accurate than using the model coefficients as in the holistic methods and CLMs.

In recent years, deep learning-based methods have shown better performance for detection since the deep learning model applies the holistic or local appearance as the feature information and embeds the implicit global face shape constraint. Some algorithms with advanced design overcome the challenges and limitations in ‘in-the-wild’ conditions. We will describe the implementation of the deep learning-based algorithms in Chapter 4. The first algorithm, called ‘Deep Alignment Network (DAN)’, was inspired by the cascade shape regression (CSR) model (Kowalski et al., 2016). The second algorithm is called ‘Deep-Convolutional-Neural-Network (DCNN)’, a cascaded network in a coarse-to-fine manner (Zhou et al., 2013). The final algorithm is called ‘Stacked Dense U-Nets (SDU)’ based upon Hourglass Networks to design a novel scale aggregation network topology and channel aggregation block. (Guo et al., 2018).

3. Datasets for facial landmark detection

The study reviews the suitable and public dataset, which can be used for landmark detection studies. The appropriate datasets that are crucial for the development of facial landmarking algorithms are required to contain rich features within the face image since algorithms can capture the relationships between the data during the training process. The datasets generally are involved 2-D datasets and 3-D datasets, and the detail (e.g., sizes, subjects, the number of annotations and year) are listed in Table 3.1 and Table 3.2.

In Table 3.1, some 3-D datasets are detailed in size, the number of annotations and the year. Compared with 2-D landmarks, 3D landmarks are defined as the 3-D coordinates of the facial landmarks, which contain the depth information of the 3-D face. What's more, 3-D landmarks are determined as the 2-D projections of the 3-D facial landmark coordinates in the face plane. The BU-3DFE database contains 100 subjects, including 56% female and 44% male, with ages from 18 years to 70 years old, who are almost from the university. Ethnic includes White, Black, East-Asian, Middle east Asian, Indian, and Hispanic Latino. (Yin et al., 2006). The 'BP4D+' (Zheng et al., 2016) is a Multimodal Spontaneous Emotion Corpus (MMSE) containing multi-model datasets. There are 140 subjects, including 58 males and 82 females, with ages ranging from 18 to 66 years old. Ethnic/Racial Ancestries include Black, White, and Asian. AFLW200-3D (Zhu et al., 2016) was constructed by using the first 2000 images of AFLW (Koestinger et al., 2011), which contain the 3D face with the corresponding 68 3-D landmarks. In addition, Menpo-3D (Deng et al., 2019) presents 84 3-D landmarks in order to better 3-D landmark localisation. However, our work only focuses on 2-D images because the 2D datasets are more rather than 3-D datasets.

Name	Sizes	Subjects	Landmarks	Year	Description
BU-3DFE	8000	100	83	2006	Each subject has seven expressions, including happiness, disgust, fear, angry, surprise, neutral and sadness. Each expression includes four levels of intensity.
BP4D+	50000	140	83	2016	Each subject has both 2-D and 3-D video to track spontaneous facial expressions in a diverse group of young adults
AFLW2000-3D	2000	Unknown	68	2016	The first 2000 images of AFLW, annotated with 68-point 3-D facial landmarks
Menpo-3D	11836	Unknown	84	2017	The images of AFLW, LFPW, Helen, IBUG, and 300W(private), annotated with 84-point 3-D facial landmarks

Table 3.1 Overview of the public 3-D face datasets

In Table 3.2, the 2-D datasets can be compromised into two categories: datasets which are collected under controlled conditions and datasets which are collected under ‘in-the-wild’ conditions without any control. The controlled datasets collected several images from a few subjects in the experiment using the camera. One or more control conditions, including different facial expressions, variation in head posture, occlusion, and illumination, can set up the experiment to generate various images, such as XM2VTS (Messer et al., 1999) and Multi-PLE (Gross et al., 2008). The uncontrolled datasets are collected from social media or websites such as facebook.com, flickr.com and google.com, in which the images are called ‘in-the-wild’ images. The ‘in-the-wild’ images provide more reliable and challenging datasets, which contain rich facial feature information, including LFPW (Peter et al., 2011), Helen (Le et al., 2012), AFLW (Koestinger et al., 2011), AFW (Zhu et al., 2012), 300W (private), IBUG (Sagonas et al., 2013), COFW (Burgos-Artizzu et al., 2013), Menpo-2D (Liu et al., 2015), and WFLW (Sagonas et al., 2017). Compared with the controlled dataset, it is easier to build an uncontrolled dataset within a larger number of ‘in-the-wild-face’ images. However, they need to manually annotate the landmark’s location in the images.

Name	Sizes	Subjects	Landmarks	Year	Description
XM2VTS	2360	295	68	1999	Controlled images with expressionless faces captured during a speech in the same light.
Multi-PIE	755,370	337	68	2008	Controlled images that face different facial expressions, occlusions, and illumination conditions
LFPW	1432	Unknown	29	2011	Uncontrolled images were collected from social media, such as Flickr, Google, and Facebook.
Helen	2000	330	68	2012	Uncontrolled images were collected from social media, such as Flickr, Google, and Facebook.
AFLW	21997	Unknown	21	2011	Uncontrolled images were collected from social media, such as Flickr, Google, and Facebook.
AFW	205	468	68	2012	Uncontrolled images were collected from social media, such as Flickr, Google, and Facebook.
IBUG	135	135	68	2013	Uncontrolled images were collected from social media, such as Flickr, Google, and Facebook.
300W (private)	600	Unknown	68	2013	Uncontrolled images contained 300 indoor and outdoor 'in-the-wild' images with a variety of expression, occlusion, and illumination conditions.
COFW	507	Unknown	68	2013	Uncontrolled images with large variations in pose, expression, and occlusion
Menpo-2D	8979	Unknown	Frontal:68 Profile:39	2015	Uncontrolled images with large variations in pose, expression, and occlusion
WFLW	10000	Unknown	98	2017	Uncontrolled images with large variations in pose, expression, and occlusion

Table 3.2 Overview of the public 2-D face datasets

In the next sections, the face datasets we utilise are briefly described, including Multi-PIE (Gross et al., 2008), LFPW (Peter et al., 2011), Helen (Le et al., 2012), AFW (Zhu et al., 2012), and Menpo-2D (Liu et al., 2015).

3.1 CMU multi-pose, illumination, and expression (Multi-PILE) face dataset

CMU multi-pose, illumination, and expression (Multi-PILE) face dataset: Over 750,000 face images contain 337 subjects in the Multi-PILE dataset. (Gross et al., 2008) All images are collected under 15 different poses and 19 different illumination conditions in the four different sessions. The face images can be divided into frontal faces containing 68 annotated landmarks and profile faces containing 39 annotated landmarks since profile faces do not perform all landmarks.



Figure 3.1 Example of the Multi-PILE (frontal) datasets

3.2 300W dataset

The 300W is a face dataset that was originally used for Facial Landmark Tracking in-the-Wild Challenge & Workshop to be held in conjunction with International Conference on Computer Vision (ICCV) 2015. It consists of three sub-sets, including LFPW, Helen and AFW datasets (Ren et al., 2014) (Xiao et al., 2016) (Zhu et al., 2016) (Kowalski et al., 2016).

Labelled face part in the wild (LFPW) dataset: Peter et al. (2011) downloaded 1432 face images from the web. The 'in-the-wild' face images contain different variety of appearances such as exaggerated facial expressions, age, make-up, and imaging environment condition. Moreover, there are also face-cropped images from movie senses and manipulated photos.



Figure 3.2 Example of the LFPW dataset

Helen dataset: Le et al. (2012) collected a large number of candidate photos, which search on Flickr. The keyword used 'portrait' and was augmented with different titles, such as 'outdoor', 'boy', 'family', 'studio', and 'wedding' etc. What's more, they use different languages to repeat the queries in order to avoid cultural bias. The photos were filtered by hand to remove false positives that represent more than one face in the single images, profile views, and low-quality images. The dataset consists of 2330 high-resolution images with various appearance variations, including pose, illumination, and occlusion.



Figure 3.3 Example of Helen's dataset

Annotated faces in the wild (AFW) dataset: Zhu. et al. (2012) collected another 'in-the-wild' face dataset consisting of 205 images, including one or more faces in a single image, and a total has 468 faces.

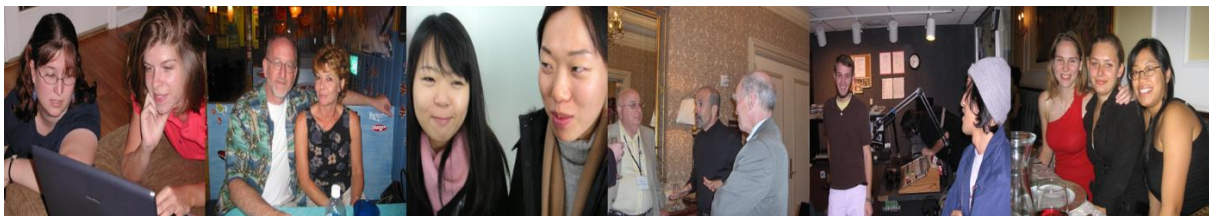


Figure 3.4 Example of the AFW dataset

3.3 Menpo 2D dataset

Menpo 2D dataset: Deng et al. (2019) collect semi-frontal and profile face image datasets under uncontrolled conditions, respectively. Then, the semi-frontal subset consists of 5658 images from AFLW (Koestinfger et al., 2011) and FDDB (Jain and Learned-Miller, 2010). Menpo becomes a challenging dataset because four factors in the dataset, including pose, illumination, occlusion, and expression, significantly influence the local facial appearance and further affect the local information for facial landmark detection models. Here are the frontal face examples of the Menpo dataset, as shown in Figure 3.5.

- Occlusion can frequently miss some facial attributes or happens on the facial contour, such as food on the mouth, and self-occlusion within some facial regions, such as almost half of the facial contour is missing in a single face image. Heavy occlusion can take challenges in estimating the position of the face since the facial attributes are missing or changed.
- The exaggerated expression can locally change some facial attributes; for example, when a person is happy, the mouth shape will be affected by the expression. Therefore, it is challenging for facial landmark detection within the face under exaggerated expressions.
- Different Illumination conditions have a significant influence on facial appearances because the changing patterns of intensity can even be ultimately missing some texture information at the facial attributes.



Figure 3.5 Example of the Menpo-2D (frontal) datasets

3.4 Landmark configuration

Since datasets with a uniform and prominent annotation configuration are convenient for assessing the experiment results, the standard landmark configuration is presented to annotate landmark locations in the appropriate datasets. The manually annotated landmarks represent the position of landmarks, which can call ground truth.

Sagonas et al. (2013) provide a reliable landmark annotation schema in the first 300W face 'in-the-wild' challenge called 'Multi-PLE 68pts'. This schema gives an opportunity of comparing the different landmark detection algorithms while employing the many datasets within the Multi-PLE 68 landmarks configuration. The Multi-PLE

annotated landmark configuration is demonstrated in Figure 3.6, which is utilised as the standard landmark configuration in our selected data.

There are two different ways to build the landmark of the Multi-PLE configuration. One way is to use Amazon Mechanical Turk (MTurk) system, which can manually identify the landmarks' location on single images. During this work, each annotator used the 'turkmarker' tool to locate the position of each landmark, and then all landmarks were determined, followed by the Multi-PLE 68 landmarks configuration. Finally, all results should be correctly recorded. However, there is hardly complete agreement between the annotators. Thus, three annotators generally are employed to carry out this work and take the mean positions of ground truth landmarks. The other way is to use an automatic landmark detector that presents a more consistent and prominent than three human annotators (Belhumeur et al., 2011). Here, our selected datasets use the automatic face detector (RetinaFace) to annotate the landmarks of Multi-PLE configurations, which achieves state-of-the-art performance on the WiderFace dataset.

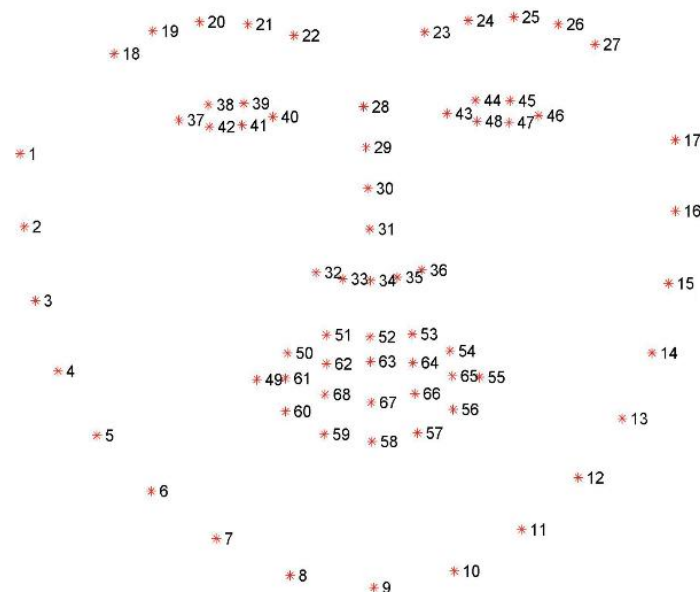


Figure 3.6 The illustration of the Multi-PLE 68 annotated landmarks configuration, adapted with Sagonas et al. (2013)

4. Implementation of facial landmark detection algorithms

Since deep learning-based facial detection has demonstrated very promising results in recent years, three widely used ones have been implemented in this research project in order to identify the appropriate detection methods which can cope the challenging conditions, such as changes in head pose, facial expression, occlusion and illumination. The methods include deep alignment network (DAN), deep convolutional neural network (DCNN) cascade and stacked dense U-net (SDU). While DAN uses the global information of facial images to predict the corresponding landmarks (Kowalski et al., 2016), DCNN is designed to predict landmarks in a coarse-to-fine manner (Zhou et al., 2013). Both DAN and DCNN are based on the fully connected layer of the convolutional network to regress the landmark positions. SDU uses the fully convolutional neural network to predict a set of heatmaps corresponding to the landmarks that need to be extracted. The heatmaps reflect the probability of landmark position at each pixel (Guo et al., 2018).

A generic processing structure of three methods is illustrated in Figure 4.1, which consists of three stages, namely data pre-processing, model training and model testing. Data pre-processing is the first stage of a facial landmark detection algorithm that extracts and transforms the facial images and corresponding landmarks in the training dataset to a standard form as well as eliminates the irrelevant background (Chen et al., 2016). Model training is the core of the detection, which determines the model parameters through a set of operations, such as convolution, max pooling, activation function, etc., based on the designed network structure and features (Sarker 2021). Finally, model testing evaluates the performance of the training network using different datasets, which could simulate the challenging conditions in the real world. The remaining chapter will describe each implemented method's processing stage in detail.

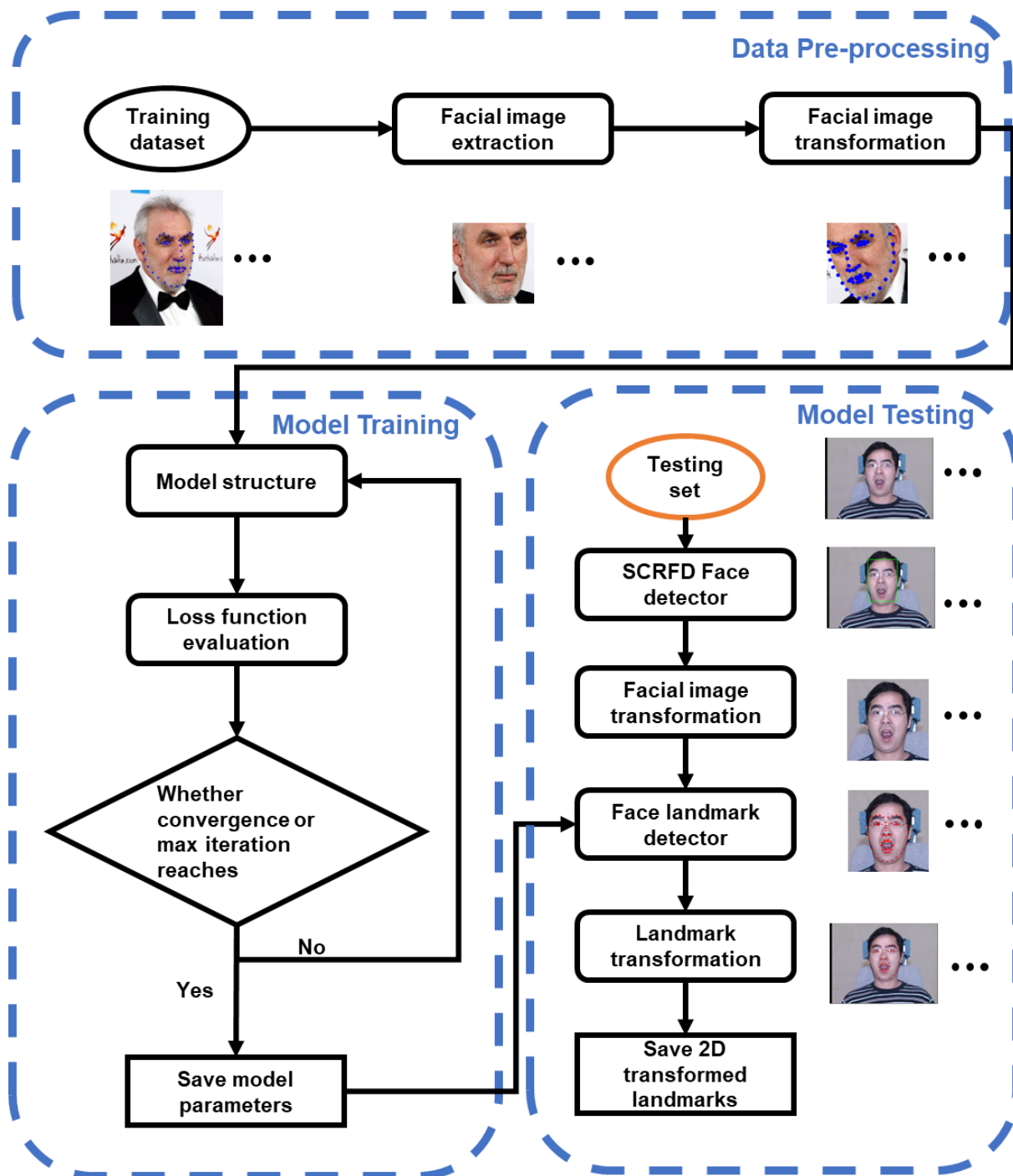


Figure 4.1 The general process of the facial landmarking system

4.1 Deep Alignment Network (DAN) Implementation detail

Kowalski et al. (2016) proposed DAN for facial landmark detection, which is based on convolutional neural networks and can predict the positions of 68 landmarks together on human faces in a cascaded manner.

4.1.1 Data pre-processing

The aim of the data pre-processing is to remove the background, extract the facial image and transform both facial images and corresponding landmarks into a standard form for the training dataset.

The first step is bounding box estimation using the provided landmarks in the training dataset. With the bounding box, a region of the face is located and extracted to a new image with the background removed. The upper-left and lower-right corners can be determined using the extreme values by a set of \mathbf{x} landmarks, $\mathbf{x} = \{\mathbf{x}_i, i \in [1, 68]\}$, where \mathbf{x}_i is the coordinate of the i^{th} landmark. The width and height of a face bounding box are calculated as follows by

$$Width_{box} = \mathbf{x}_{max} - \mathbf{x}_{min} \quad (4.1)$$

$$Height_{box} = \mathbf{y}_{max} - \mathbf{y}_{min} \quad (4.2)$$

where $(\mathbf{x}_{min}, \mathbf{y}_{min})$ is the coordinate at the upper-left corner, $(\mathbf{x}_{max}, \mathbf{y}_{max})$ is the coordinate at the lower-right corner, $Width_{box}$ and $Height_{box}$ are the weight and height of a face bounding box separately. The coordinates of landmarks are then normalised to the range of (0,1) by using (4.3).

$$\bar{\mathbf{x}}_i = \left(\frac{(\mathbf{x}_i - \mathbf{x}_{min})}{Width_{box}}, \frac{(\mathbf{y}_i - \mathbf{y}_{min})}{Height_{box}} \right) \quad (4.3)$$

where $(\mathbf{x}_i, \mathbf{y}_i)$ is the coordinate of the i^{th} landmark, and $\bar{\mathbf{x}}_i$ are the corresponding normalised landmarks.

The normalised landmarks $\bar{\mathbf{x}}$ are augmented with a random combination of rotation, scaling, or horizontal flip and followed by a multiplication of pre-defined facial image sizes as:

$$\mathbf{x}_i' = s(\mathbf{cR}_{2D}(\theta) \cdot \bar{\mathbf{x}}_i) \quad (4.4)$$

where \mathbf{x}' are the transformed landmarks of an image, i is the landmark's index, c is the scaling factor that the range is (0.8, 1.2), $\mathbf{R}_{2D(\theta)} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}$ is a 2x2 rotation matrix, that the range of θ is $(-20^\circ, +20^\circ)$, a horizontal flip represented by $\begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$ transformation matrix is also randomly applied, and s is the factor of the input image size that is set as 112. The augmentation is to increase the number of images in the training dataset since the networks heavily rely on big data to reduce overfitting. Overfitting refers to perfectly model the training data when a network learns a function with very high variance. (Shorten, C and Khoshfotaar, T.M., 2016)

Having the original and transformed landmarks, a similarity transformation between them is estimated as

$$\mathbf{x}'_i = \mathbf{M} \cdot \mathbf{x}_i \quad (4.5)$$

where \mathbf{x}' are the transformed landmarks, the coordinate represented by $\begin{bmatrix} \mathbf{x}'_i \\ \mathbf{y}'_i \end{bmatrix}$, $\mathbf{M} = \begin{bmatrix} a_{11} & b_{12} & c_{13} \\ a_{21} & b_{22} & c_{23} \end{bmatrix}$ is a 2x3 transformation matrix, that a and b represent the factors of rotation and scaling, c represents the translation factor, and \mathbf{x} is the original landmarks, with the coordinate is $\begin{bmatrix} \mathbf{x}_i \\ \mathbf{y}_i \\ 1 \end{bmatrix}$.

Once the transformation matrix \mathbf{M} is obtained, the extracted facial image is then transformed in order to align its region of the face in the image with the transformed landmarks as Figure 4.2 shows the examples of original images with the landmark, the extracted facial image and the transformed facial images with the corresponding landmarks.










The original images with landmark	The extracted facial image	The transformed facial images with the transformed landmarks
		
		
		

Figure 4.2 Examples of intermediate and final results of data pre-processing

In DAN, all transformed facial images are standardised before feeding into the network, and the following equation shows the function:

$$\bar{\mathbf{I}} = \frac{\mathbf{I}' - \mathbf{I}'_{\text{avg}}}{\mathbf{I}'_{\text{std}}} \quad (4.6)$$

where $\bar{\mathbf{I}}$ is the offset value of a transformed facial image, \mathbf{I}'_{avg} and \mathbf{I}'_{std} are the average pixel value and the standard deviation of the image, respectively.

4.1.2 DAN Model Training

Model Structure

Having the standardised form of facial images and the corresponding landmarks in the training dataset, a DAN model can be constructed through the model training, which consists of two stages, each of which has a feed-forward neural network (Simonyan and Zisserman, 2014), as shown in Figure 4.3. The offset value of all landmark positions $\Delta \mathbf{x}_1$ is predicted by the first network, while the mean shape \mathbf{x}_0 is

computed, which is the average of all transformed landmarks of the training dataset. Then, the predicted landmarks \mathbf{x}_1 of the first DAN stage define as the following:

$$\mathbf{x}_1 = \mathbf{x}_0 + \Delta\mathbf{x}_1 \quad (4.7)$$

In the following, connection layers apply the predicted landmarks to generate the input to the network of the second stage, including the input image $\mathbf{M}_2(\bar{\mathbf{I}})$ which uses the transform matrix \mathbf{M}_2 to align the input image with the predicted landmark, the landmark heatmap \mathbf{E}_2 , and the feature image \mathbf{L}_2 . In addition, a transform matrix \mathbf{M}_2 also used to produce the mean shape $\mathbf{M}_2(\mathbf{x}_1)$ of the second stage and its inverse \mathbf{M}_2^{-1} .

Through the second network, the final offset value of the predicted landmarks $\Delta\mathbf{x}_2$ are predicted, the predicted landmarks should transform back to match the original image by using an inverse similarity transform. The landmark output is defined as the following (Kowalski et al., 2016):

$$\mathbf{x}_2 = \mathbf{M}_2^{-1}(\mathbf{M}_2(\mathbf{x}_1) + \Delta\mathbf{x}_2) \quad (4.8)$$

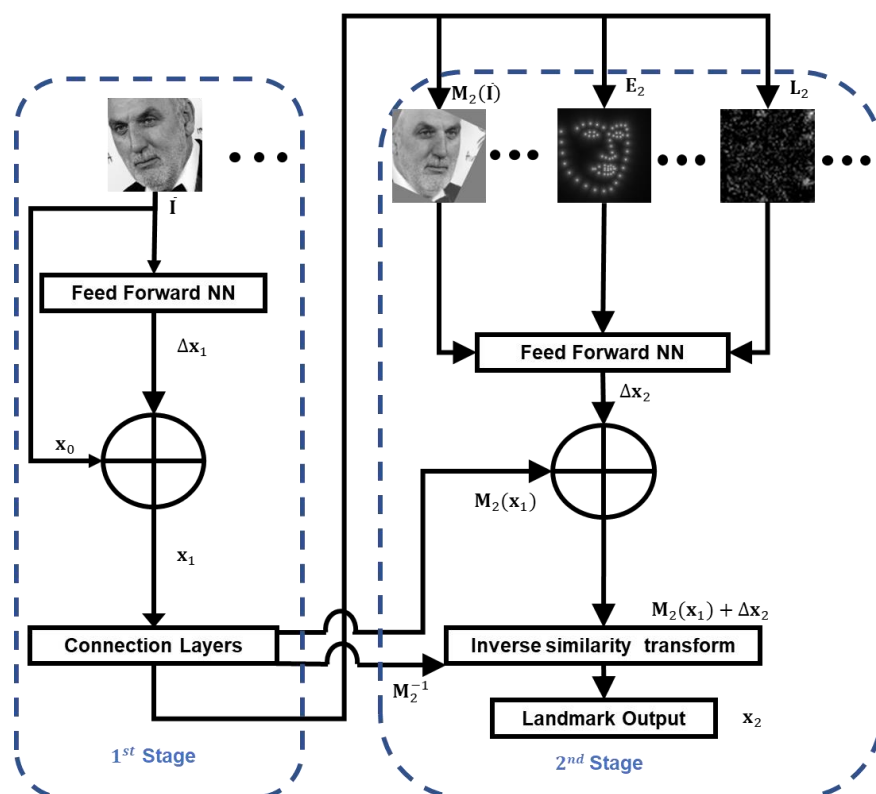


Figure 4.3 DAN model structure

Feed-forward neural network

In Figure 4.4, a structure of a feed-forward neural network contains ten layers, including four sets of convolutional layers, followed by max-pooling and two fully connected layers. Each convolutional layer (grey box) utilises the advantage of batch normalisation (Ioffe and Szegedy, 2015) and Rectified Linear Units (ReLU) (Krizhevsky et al., 2012) as the activation functions.

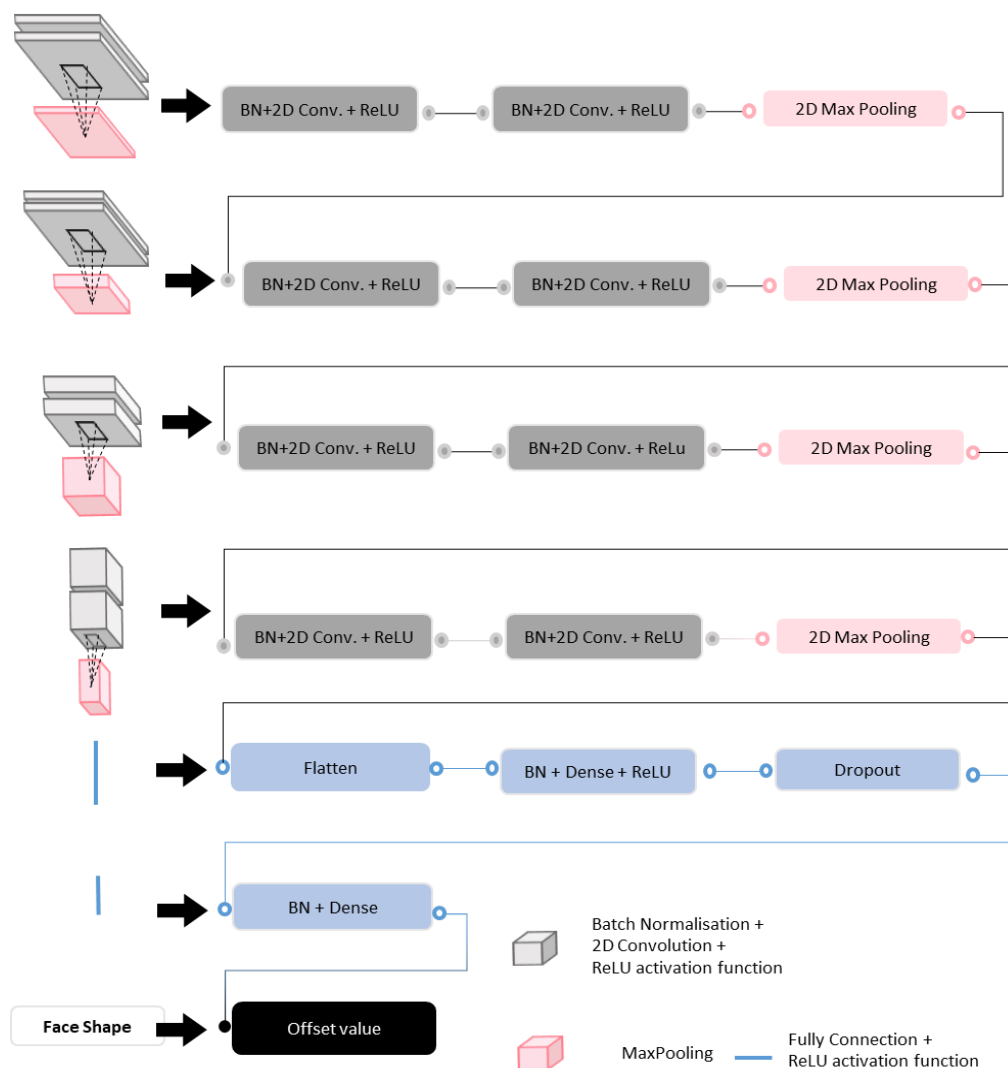


Figure 4.4 Structure of a feed-forward neural network of a DAN stage

The batch normalisation layer standardises the input images of each mini-batch to the 2-D convolutional layer, which can accelerate the training processing of the neural network. The batch normalisation layer (Ioffe and Szegedy, 2015) applies a transformation that maintains the distribution of output mean values close to 0 and the output standard deviation close to 1. The batch normalisation transform is present in the following equation:

$$\mathbf{Y}_i = \lambda \left(\frac{\bar{\mathbf{I}}_i - \frac{1}{m} \sum_{i=1}^m \bar{\mathbf{I}}_i}{\sqrt{\frac{1}{m} \sum_{i=1}^m \left(\bar{\mathbf{I}}_i - \frac{1}{m} \sum_{i=1}^m \bar{\mathbf{I}}_i \right)^2} + \epsilon} \right) + \beta \quad (4.9)$$

where \mathbf{Y}_i is the output value, λ and β are the parameters of scale and shift to be learned during training processing, m is the number of input images of a mini-batch, and ϵ is a small constant added to the mini-batch variance for numerical stability.

The 2-D convolutional layer sample the feature map from the output image of the batch normalisation layer, as seen in Figure 4.4 above. Each 2-D convolutional layer follows an activation function (ReLU), which can bring non-linearity into the neural network (Krizhevsky et al., 2012). The convolutional operation can be equal to (Ma and Lu, 2017):

$$g(\mathbf{X}_{i,j,k}^t) = \sum_{m=0}^{w_1} \sum_{n=0}^{w_2} \mathbf{Y}_{i+m,j+n,l}^{t-1} * \mathbf{W}_{m,n,k}^t + \mathbf{b}_k \quad (4.10)$$

$$g(\mathbf{X}) = \begin{cases} \mathbf{X}, & \text{if } \mathbf{X} > 0 \\ 0, & \text{other wise} \end{cases} \quad (4.11)$$

where \mathbf{Y}^{t-1} is the input feature map, \mathbf{X} is the output feature map, g is an activation function, \mathbf{W} is the kernel, i and j represent the height and width of the feature map, l is the input channel of the feature map, \mathbf{b} is the bias, m is the row number of the kernel, n is the column number of the kernel, k is the number of kernels or the output channel of the feature map.

The pooling layer utilises the max-pooling operation, called down-sampling. The max-pooling operation uses kernels to take the maximum value within the feature map of the kernel size, in which the kernel size is 2×2 , and the stride is 1 pixel (Simonyan and Zisserman, 2014). The max-pooling operation can take the most important information and send it to the following convolutional layers. The height and width of the feature map also reduce to improve memory efficiency since the original image size would not be memory efficient for the end step. The operation of the output feature map (Zhou et al., 2013) is shown below:

$$\mathbf{I}_{i,j,k}^t = \max_{i*s \leq m \leq i*s+f, i*s \leq n \leq i*s+f} (\mathbf{X}_{m,n,k}^t) \quad (4.12)$$

where \mathbf{I} is the output feature map, \mathbf{X} is the input feature map, i, j are the height and width of the output feature map, m, n are the height and width of the input feature map, k is the channel of the feature map, s is the stride size, and the max-pooling layer's kernel has same height and width that equal to f .

Then, the fully connected layer's structure illustrates in Figure 4.5, which are two levels of dense layers to regress the prediction of the offset value, while there is a dropout layer between two dense layers. In the flatten layer, the output feature maps of the final max-pooling layer flatten to one dimension feature vector as each node represents a scalar value in a vector. The operation of the dense layer defines as the following (Ma and Lu, 2017):

$$g(\mathbf{X}_{i \times 1}^{n+1}) = \mathbf{W}_{i \times j} * \mathbf{X}_{1 \times j}^n + b_{i \times 1} \quad (4.13)$$

where \mathbf{X} is one dimension feature vector, \mathbf{W}, b are the parameters of the weight and bias that the model learned, n is the number of the level, j is the number of the input vector, and i is the number of the output vector.

In the following of the first dense layer, the drop layer randomly ignores a set of nodes, as shown in Figure 4.5. The operation of the dropout layer is to randomly set a part of input nodes to 0 with a frequency of rate during training, which generally helps to prevent overfitting (Nitish et al., 2014). Since the other input nodes not to set 0 are scaled up by $1/(1-\text{rate})$, the dropout layer's output can keep the same as the sum over all inputs. Having the drop layer's output vector, the second dense layer regresses the offset value of the predicted landmarks' positions.

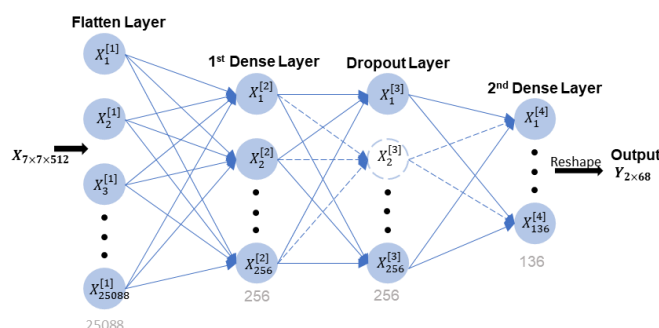


Figure 4.5 The diagram of fully connected layer regression

Table 4.1 outlines the input and output feature map size of each layer, kernel size, and convolution stride. The feature maps describe as height \times width \times channel. However, the second feed-forward neural network has three input images, the input feature map size of the first convolution layer is $112 \times 112 \times 3$.

Layer	Input feature map size	Output feature map size	Num of kernel	Kernel size	Stride	Padding
Conv.1_1	$112 \times 112 \times 1$	$112 \times 112 \times 64$	64	3×3	1	1
Conv.1_2	$112 \times 112 \times 64$	$112 \times 112 \times 64$	64	3×3	1	1
Max Pooling_1	$112 \times 112 \times 64$	$56 \times 56 \times 64$	1	2×2	2	1
Conv.2_1	$56 \times 56 \times 64$	$56 \times 56 \times 128$	128	3×3	1	1
Conv.2_2	$56 \times 56 \times 128$	$56 \times 56 \times 128$	128	3×3	1	1
Max Pooling_2	$56 \times 56 \times 128$	$28 \times 28 \times 128$	1	2×2	2	1
Conv.3_1	$28 \times 28 \times 128$	$28 \times 28 \times 256$	256	3×3	1	1
Conv.3_2	$28 \times 28 \times 256$	$28 \times 28 \times 256$	256	3×3	1	1
Max Pooling_3	$28 \times 28 \times 256$	$14 \times 14 \times 256$	1	2×2	2	1
Conv.4_1	$14 \times 14 \times 256$	$14 \times 14 \times 512$	512	3×3	1	1
Conv.4_2	$14 \times 14 \times 512$	$14 \times 14 \times 512$	512	3×3	1	1
Max Pooling_4	$14 \times 14 \times 512$	$7 \times 7 \times 512$	1	2×2	2	1
1st dense	1×25088	1×256		-		
2nd dense	1×256	$1 \times 1 \times 136$		-		

Table 4.1 Structure of the feed-forward network

Connection layers

The connection layers can produce the five outputs for the second stage, as illustrated in Figure 4.6. The connection layers include five layers, consisting of transform estimation, image transform, landmark transform, heatmap generation and feature generation. Given the predicted landmarks \mathbf{x}_1 and the mean shape \mathbf{x}_0 , the transform estimation is responsible for calculating the similarity transformations matrix \mathbf{M}_2 and the inverse \mathbf{M}_2^{-1} , using (4.5). Based on the similarity transformation matrix \mathbf{M}_2 , image transform is applied to warp the input image of the first network $\bar{\mathbf{I}}$, while landmark transform is used to transform the predicted landmarks \mathbf{x}_1 that can

be close to the mean shape \mathbf{x}_0 . The heatmap generation uses the mean shape $\mathbf{M}_2(\mathbf{x}_1)$ to calculate landmark heatmap \mathbf{E}_2 , using the following equation (Kowalski et al., 2016):

$$\mathbf{E}_2(x, y) = \frac{1}{1 + \min_{x_i \in \mathbf{M}_2(\mathbf{x}_1)} \|(x, y) - x_i\|} \quad (4.14)$$

where \mathbf{E} is the heatmap image, x_i is the i^{th} landmark of $\mathbf{M}_2(\mathbf{x}_1)$, and x, y is the coordinate of each pixel on the heatmap. The landmark heatmap is an image with the highest pixel intensity at the landmarks' positions, and it can infer the landmark estimation based on the entire image. Then, the feature generation is generated a feature image \mathbf{F}_2 from the first dense layer of the first network. The feature image can transfer any information learned by the first stage to the second stage.

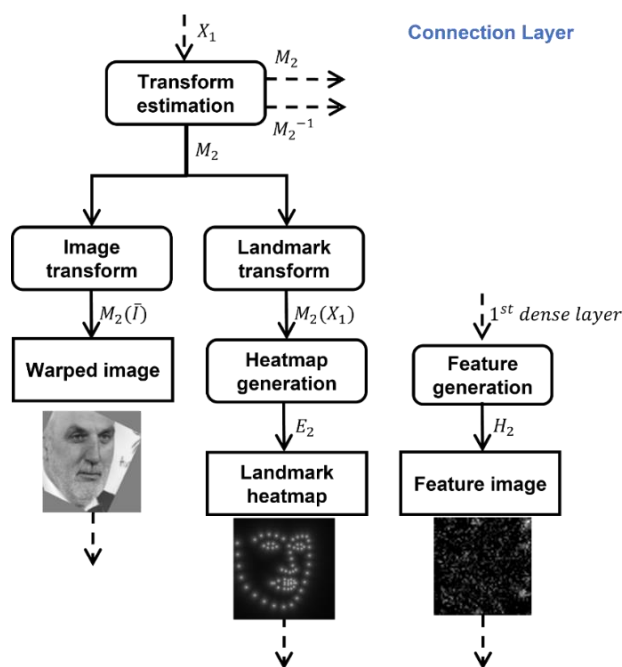


Figure 4.6 The diagram of the connection layers

The examples of three kinds of images are illustrated in Figure 4.7, including the warped image $\mathbf{M}_2(\bar{\mathbf{I}})$, the landmark heatmap \mathbf{E}_2 , and the feature image \mathbf{F}_2 . These three kinds of images are concatenated as the input into the feed-forward neural network of the second stage, while these images need to reshape to the size of 112×112 , which can keep the same input size of networks in the two stages. Furthermore, the inverse similarity transformation matrix \mathbf{M}_2^{-1} and the mean shape

$\mathbf{M}_2(\mathbf{x}_1)$ are utilised in (4.8), which can transform the output of the second stage back to match the original image.










Warped image	Landmark heatmap	Feature image
		
		
		

Figure 4.7 Example of input images into the second stage

4.1.3 DAN Loss Evaluation

The loss evaluation of DAN uses the loss function to evaluate the training model parameters. The optima model parameters will be saved when the error converges, and the max iteration reaches. In the model training of DAN, the loss function is employed to calculate the error between the 68 predicted landmarks and the 68 annotated landmarks or called ground truth. The loss function minimises the landmark location error normalised by the distance between the eye centres (Kowalski et al., 2016):

$$\min_{\mathbf{x}} \frac{\| \mathbf{M}_2^{-1}(\mathbf{M}_2(\mathbf{x}_1) + \Delta\mathbf{x}_2) - \mathbf{x}^{gt} \|}{d_{pupils}} \quad (4.15)$$

where \mathbf{x}^{gt} are the landmark ground truth, \mathbf{M}_2^{-1} is the inverse matrix, $\mathbf{M}_2(\mathbf{x}_1)$ is the mean shape from the first stage, $\Delta\mathbf{x}_2$ is the output offset value of 68 landmark coordinates, and d_{pupils} is the normalised factor, which is the distance between the eye centre.

4.1.4 DAN Model testing

The section provides the process of the model testing of DAN by jointly using the face detector and facial landmark detector, as shown in Figure 4.1. In the original author's implementation, the model testing of DAN uses the data pre-processing stage to transform the original image with the corresponding landmarks as input into the facial landmark detector (Kowalski et al., 2016). However, the data pre-processing needs an image with the landmark ground truth, which is not allowed in the test stage. Therefore, the face detector is used instead of the data pre-processing stage. The face detector estimates the face bounding box. Then, the face bounding box can transform the input image into the facial landmark detector for landmark estimation.

Figure 4.8 uses the face detector to estimate the face bounding box step by step. The original images must resize to a fixable image size through the pre-trained model, and the input image size must be set as 224×224 using bilinear interpolation. To maintain the same information as the original image, the height and width of an image need to keep the same ratio, and the rest of the resized image is padded by 0. Hence, the face bounding box estimate by the face detector, which is the pre-trained model, was applied to predict the face bounding box.

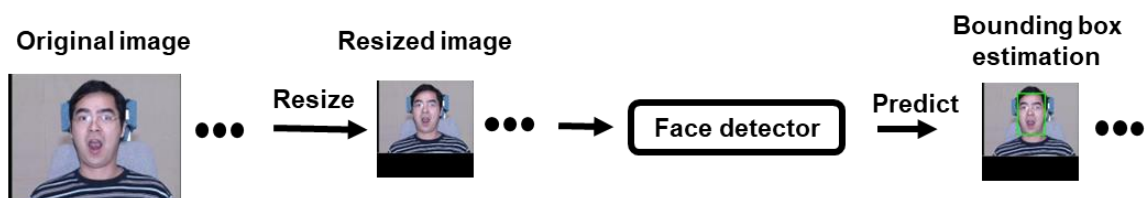


Figure 4.8 The face detector workflow

The face detectors use the pre-trained models of Multi-Task Cascaded Convolutional Networks (MTCNN) (Zhang et al., 2016) and Sample and Computation Redistribution for efficient Face Detection. (SCRFD) (Guo et al., 2021) to present a comparison result. The results show the accuracy of each face detector and the number of false positives and missing using the Multi-PLD dataset, 300W challenge dataset, and Menpo dataset. A face was detected correctly if the predicted face bounding box returned by a face detector and then the ground truth of the bounding box overlapped by at least 50%. According to the bounding box estimation in the data pre-processing, the height and width of the facial region can be determined by

using (4.1) and (4.2). In Figure 4.9, each dataset lists an image with its correct face bounding box and the poor result of each face detector. In the correct detection, the red box is the ground truth of the bounding box, the green box is the prediction of SCRFD, and the blue box is the prediction of MTCNN. In the poor results of each face detector, the red box is the ground truth of the bounding box, and the green box is the prediction box.

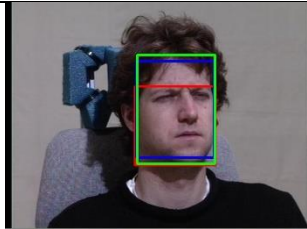
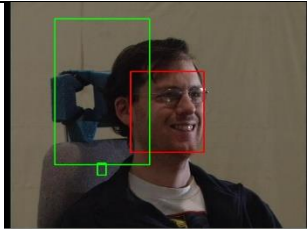
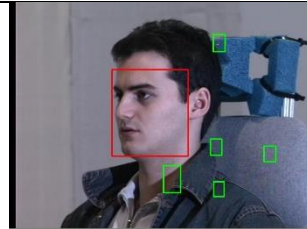
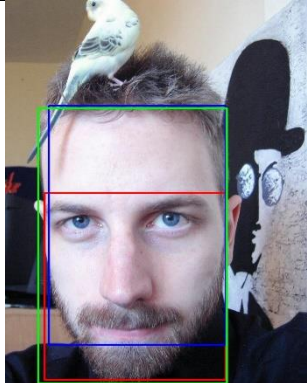
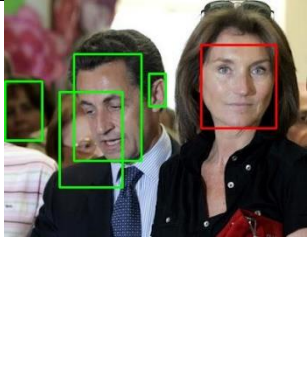
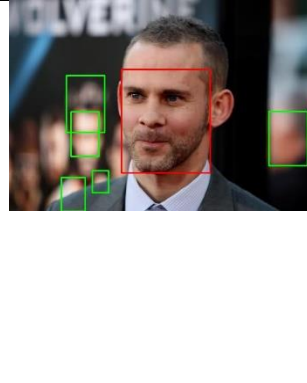
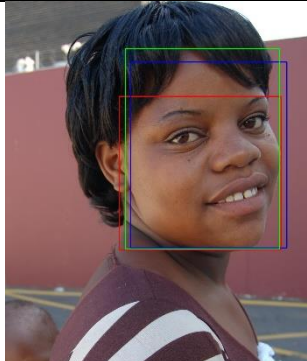
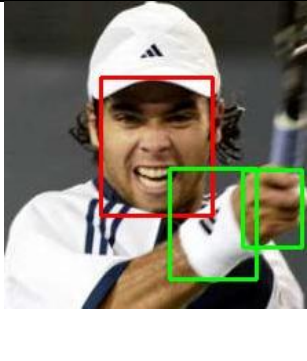
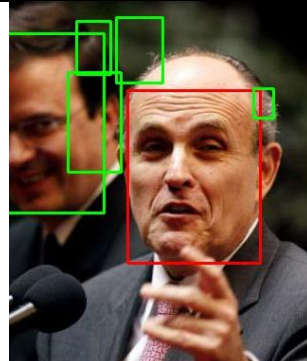
Dataset	Correct Detection	Poor Detection of MTCNN	Poor Detection of SCRFD
Multi- PLE			
300W			
Menpo			

Figure 4.9 Example results of Face detector.

Figure 4.10 shows the accuracy of face detectors, and N represents the number of images in the dataset. The accuracy of each face detector in a dataset is computed as the number of correct predictions divided by the total image numbers of a dataset.

It could obtain that the face detector of the SCRFD outperformed the face detector of the MTCNN in all datasets.

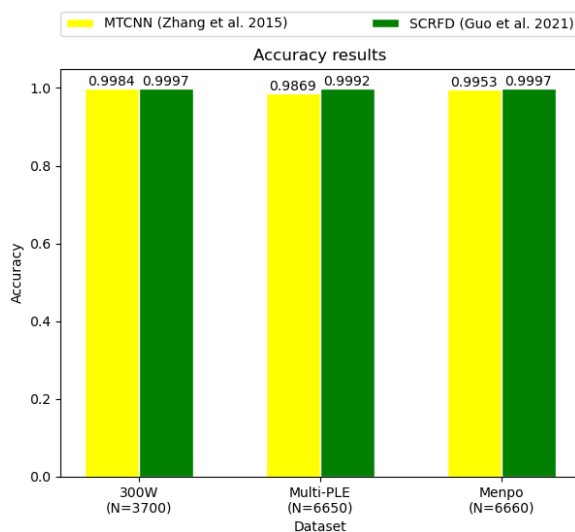


Figure 4.10 Face detectors' accuracy

Since each image in datasets only consists of one face's landmark configuration, the additional boxes in a single image, not the face with the ground truth, are determined as false positives. The comparison result of false positives shows in Figure 4.11(a). It still exists some images that a face detector cannot return a bounding box denoted as missing images. Figure 4.12(b) shows the result of missed images. In conclusion, the face detector of the SCRFD outperformed the face detector of the MTCNN with higher accuracy, fewer false positives, and fewer missed images.

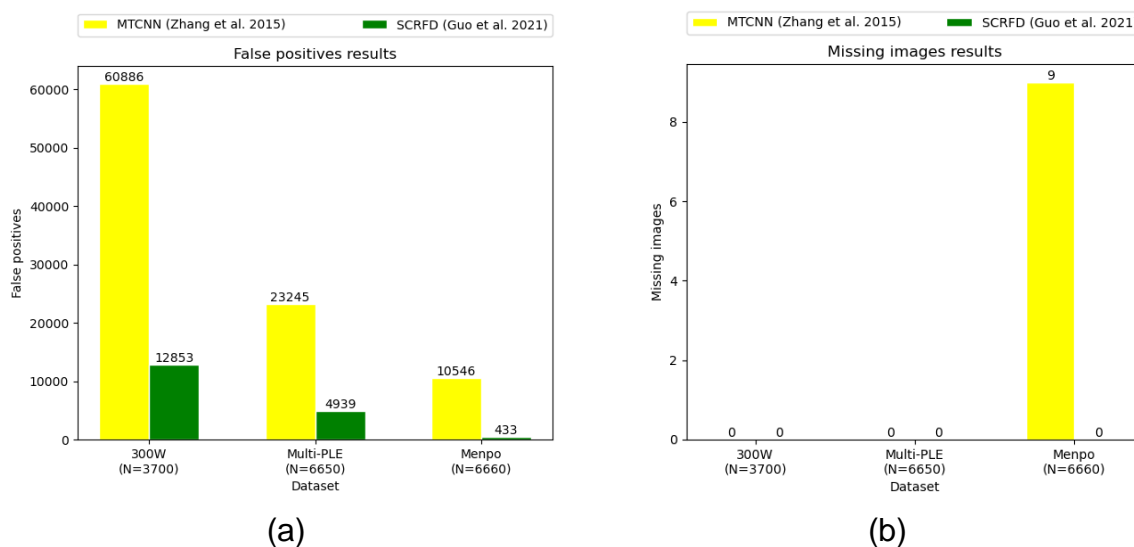


Figure 4.11 Face detector results (a) false positives, (b) missing images

With the face bounding box estimated by the SCRFD, the similarity transform uses to sample the box region of the original image as the input of the facial landmark detector. The transformation matrix is defined as (Jaderberg et al., 2015):

$$\begin{bmatrix} scale & 0 & t_x \\ 0 & scale & t_y \end{bmatrix} \quad (4.16)$$

$$scale = \frac{S}{\max(w, h)} \quad (4.17)$$

$$t_x = \frac{S}{2} - w'_{centre} \quad (4.18)$$

$$t_y = \frac{S}{2} - h'_{centre} \quad (4.19)$$

where $scale$ is the scaling ratio between the original image and transformed image, w, h are the width and height of the original image, S represents the width of the transformed image as the same as the height, which is 112, t_x and t_y are the translation scalar at the x-axis and the y-axis, and $(w'_{centre}, h'_{centre})$ is the centre coordinate of the face bounding box in the transformed image. Hence, the original images transform to the transformed images using (4.6), as shown in Figure 4.12.

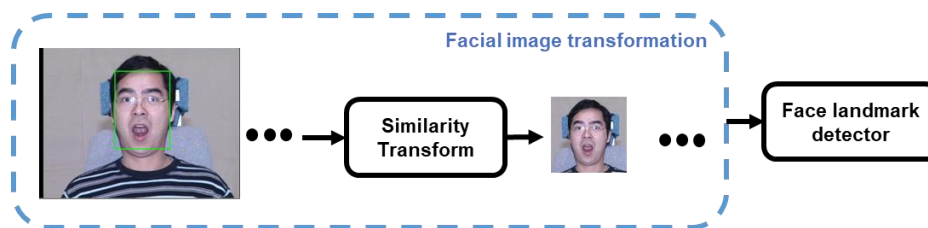


Figure 4.12 The image transform

The Red, Green, and Blue (RGB) colour-resized images transform the greyscale-resized images. Then, the facial landmark detector uses the saved model parameters and transformed images to infer the predicted landmarks' positions. Furthermore, the 68 predicted landmarks can transform to match the original image using the inverse similarity transformation matrix. The transformed landmarks can be used to compare the performance with other facial landmark detection algorithms. As shown in Figure 4.13, the left is the predicted landmarks on the transformed image, and the right is the original image with the transformed landmarks.

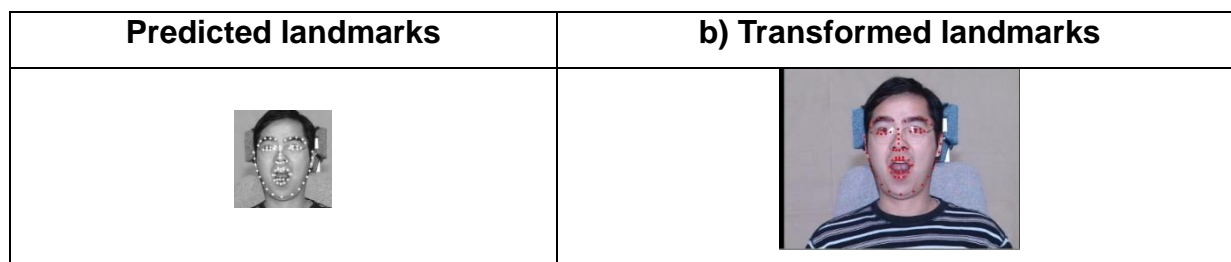


Figure 4.13 The prediction landmarks' result of DAN

4.2. Coarse-to-fine Deep Convolutional Neural Network (DCNN) Cascade Implementation detail

The coarse-to-fine Deep Convolutional Neural Network (DCNN) cascade (Zhou et al., 2013) proposed a four-level convolutional network cascade to handle the challenge of 68 landmarks' positions in a coarse-to-fine manner. The coarse-to-fine DCNN cascade has three stages, which are the same as DAN. Since the data pre-processing of DCNN is exactly the same as DAN, the following section will focus on DCNN model training and testing.

4.2.1. DCNN Model Training

Model Structure

Having the normalised form of facial images and the corresponding landmarks in the training dataset, the model builds a two-level network, which refines a set of landmarks on the local patches provided by the previous level. As shown in Figure 4.14, the framework gives a brief demonstration, including the coarse and the refine level.

In the coarse level, 68 landmarks are separated into inner points and contour points: the inner points denote the 51 landmarks of eyes, brows, mouth, and nose as inner, and the contour points denote the other 17 landmarks. (Zhou et al., 2013). The coarse level's network utilises two individual deep convolutional neural networks (DCNN) to predict the inner points and contour points' position separately. Once the inner points at the coarse level have been predicted, the region of four facial attributes, including eyes, brows, mouth, and nose, are extracted to the local patches. These local patches of each attribute are applied as the inputs of the four individual networks in the refine level. These networks have the same network architecture, called 'Deep convolutional neural network (DCNN)', which was inspired by the work

of Sun et al. (2013). The final 68 predicted landmarks combine with the inner points prediction at the refine level and the contour points prediction at the coarse level.

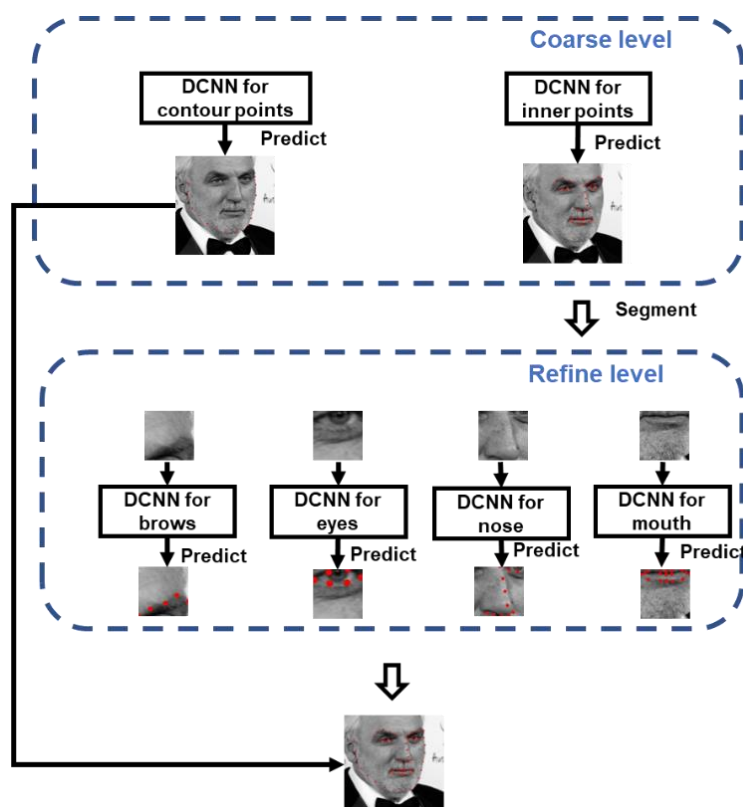


Figure 4.14 The outline of the coarse-to-fine Deep convolution network cascaded.

Deep Convolution Neural Network (DCNN)

A deep convolutional neural network (DCNN) denotes a basic network of the model, and the architecture (Zhou et al., 2013) is illustrated in Figure 4.15. The network applies the three convolutional layers, each following a max-pooling layer and a fully connected layer taken in the end. An unshared convolutional layer only uses in the network of the refine level, and each face attribute uses an individual DCNN to predict the landmark position in each face attribute. The structure is adopted by Zhou et al. (2013).

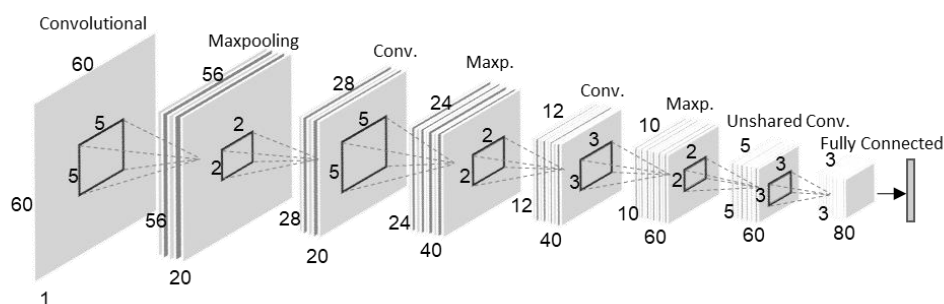


Figure 4.15 Typical network structure of DCNN, redrawn from Zhou et al. (2013)

Each convolution layer employs a set of square convolution kernels to the multichannel input feature maps. Then the convolution operation with an activation function is computed by (Zhou et al., 2013):

$$\mathbf{X}_{i,j,k}^t = \left| \tanh \left(\sum_{x=0}^{h_{t-1}} \sum_{y=0}^{w_{t-1}} \sum_{z=0}^{c_{t-1}} \mathbf{I}_{i-x,j-y,z}^{t-1} \cdot \mathbf{W}_{x,y,z,k}^t + \mathbf{b}_k \right) \right| \quad (4.24)$$

where \mathbf{I}^{t-1} is denoted as the output feature maps of the previous convolution layer, t is the t^{th} convolutional layer, h and w are the height and width of the input feature maps, c is the number of the channel of the feature map, \mathbf{W} is weight, \mathbf{b} is bias, and \mathbf{X} is the output feature maps. For $i = h - s + 1$ and $j = w - s + 1$, i and j represent the size of each region of feature maps. Thus, each region is converged by the kernel with $s \times s$. After the convolution operator, the hyper-tangent and absolute value functions are employed to the t^{th} output feature map, which brings non-linearity to the model.

After the convolution layer, the Max-pooling layer and the fully connected layer are employed, which is the same as DAN. Finally, the 68 predicted landmarks are produced by one fully connected layer. After describing DCNN architecture, the networks' sizes of the coarse level and refine level are illustrated below.

At the coarse level, Table 4.2 summarises the network size with **C1** and **C2** (Zhou et al., 2013). Table 4.2 shows the input images' resolution, kernel size and the number of channels. **C1** and **C2** constructed to predict the contour points and the inner points separately.

Network	C1	C2
Input	120 × 120 × 1	120 × 120 × 1
Conv.1	5 × 5 × 20	5 × 5 × 20
Max Pooling_1	2 × 2	2 × 2
Conv.2	5 × 5 × 40	5 × 5 × 40
Max Pooling_2	2 × 2	2 × 2
Conv.3	3 × 3 × 60	3 × 3 × 60
1st dense	1 × 34	1 × 102
Output	2 × 17	2 × 51

Table 4.2 The coarse level's network size

As seen in Table 4.3, the refine level constructs four networks with the same input resolution, kernel size and the number of channels (Zhou et al., 2013). The first network *R1* predicts a left brow that has 5 landmarks, while the network reuses to predict the images of the right brow that also have 5 landmarks. The second network *R2* predicts a nose that has 9 landmarks. The third network *R3* predicts an eye that has 6 landmarks, and the network reuses to predict the images of the right eye that also have 6 landmarks. The last one *R4* predicts a mouth that has 20 landmarks.

Network	<i>R1</i>	<i>R2</i>	<i>R3</i>	<i>R4</i>
Input	$40 \times 40 \times 1$	$40 \times 40 \times 1$	$40 \times 40 \times 1$	$40 \times 40 \times 1$
Conv.1	$5 \times 5 \times 20$	$5 \times 5 \times 20$	$5 \times 5 \times 20$	$5 \times 5 \times 20$
Max Pooling_1	2×2	2×2	2×2	2×2
Conv.2	$3 \times 3 \times 40$	$3 \times 3 \times 40$	$3 \times 3 \times 40$	$3 \times 3 \times 40$
Max Pooling_2	2×2	2×2	2×2	2×2
Conv.3	$3 \times 3 \times 60$	$3 \times 3 \times 60$	$3 \times 3 \times 60$	$3 \times 3 \times 60$
1st dense	1×10	1×18	1×12	1×40
Output	2×5	2×9	2×6	2×20

Table 4.3 The refine level's network size.

Facial attributes extraction

Since the networks in the refine level train each face attribute separately, the facial attributes must be extracted and resized from the whole image to the local patch. As seen in Figure 4.16, the coordinates of the 51 predicted inner points provided by coarse level are applied to extract each facial component. In addition, each component needs to be extracted as a square to keep the facial appearance. The length of the square selects the longer length between the width and height of the components. Finally, the bilinear interpolation methods can use to resize the facial components as the input of the refine level.

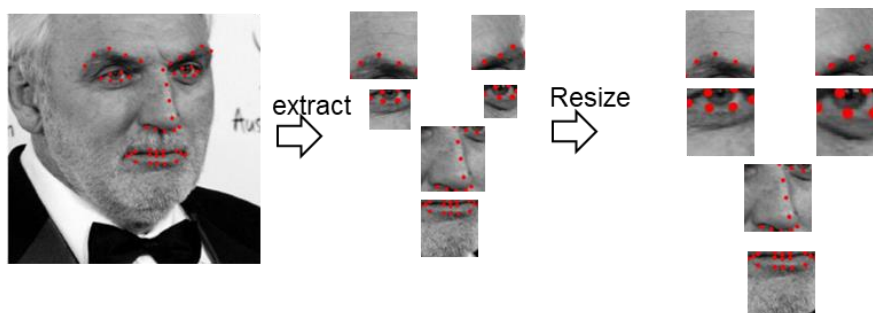


Figure 4.16 Illustration of facial attributes extraction

4.2.2 DCNN Loss evaluation

The loss evaluation of DCNN uses the mean square error (MSE) loss function (Sammut and Webb, 2010) to evaluate the training model parameters. The optima model parameters will be saved, and the saved way is the same as DAN. The number of landmark outputs is different in the refined-level networks, so the distance between the eye centre cannot be used to normalise the distance between the predicted landmarks' positions and the corresponding landmark annotations. Therefore, the MSE loss is applied to adjust the parameter of each network.

$$L(\mathbf{Y}, \hat{\mathbf{Y}}) = \frac{1}{N} \sum_{i=0}^N (\mathbf{Y}_i - \hat{\mathbf{Y}}_i)^2 \quad (4.25)$$

where $\hat{\mathbf{Y}}$ is the predicted landmark, \mathbf{Y} is the corresponding ground truth, N is the number of landmarks, i is the index of the landmarks.

4.2.3 DCNN Model testing

In this section, the test stage of DCNN jointly uses the SDCRFD face detector and facial landmark detector to output the 68 predicted landmarks. The face detector uses the same pre-trained model and image processing step as DAN, while the facial landmark detector applies DCNN to predict the position of landmarks.

In Figure 4.17, the landmark prediction of the facial landmark detector illustrates. The facial landmark detector uses two different trained networks of the coarse level to predict 17 contour points and 51 inner points, as seen in Figure 4.18(a). The predicted landmarks over each face attribute used four different trained networks' parameters of the refine level, as illustrated in Figure 4.18(b). Finally, the transformed landmarks are shown in the original images in Figure 4.18(c).

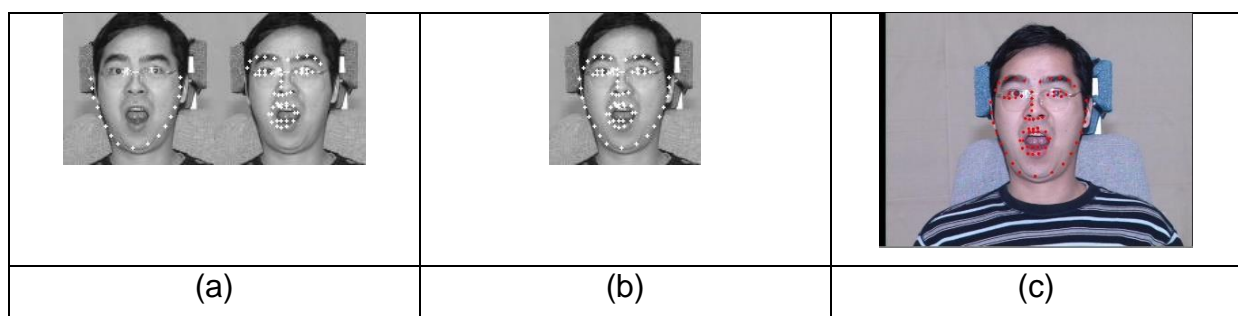


Figure 4.17 The example of landmark prediction (a) Coarse level, (b) Refine level, and (c) transformed landmarks

4.3 Stacked Dense U-nets (SDU) Implementation detail

Stacked Dense U-nets (Guo et al., 2018) is a facial landmark detection algorithm that applies fully convolutional neural networks for heatmap prediction. The network is designed to present a set of landmark heatmaps, which show the probability of landmarks' position at each pixel.

4.3.1 Data pre-processing

Since the data pre-processing has been detailed in Section 4.1.1, the pre-processing stage of the SDU follows the data pre-processing steps to transform the training set's image as the network input. Due to the novel loss function (Guo et al., 2018), the input of the SDU needs two kinds of transformed images from a single image in train sets. As seen in Figure 4.18, the face-cropped image is the first kind of transformed image, which extracts the facial region and resizes it to $128 \times 128 \times 3$. The other kind of transformed image uses data augmentation by random rotation, scaling, or flip, and the size also is $128 \times 128 \times 3$.

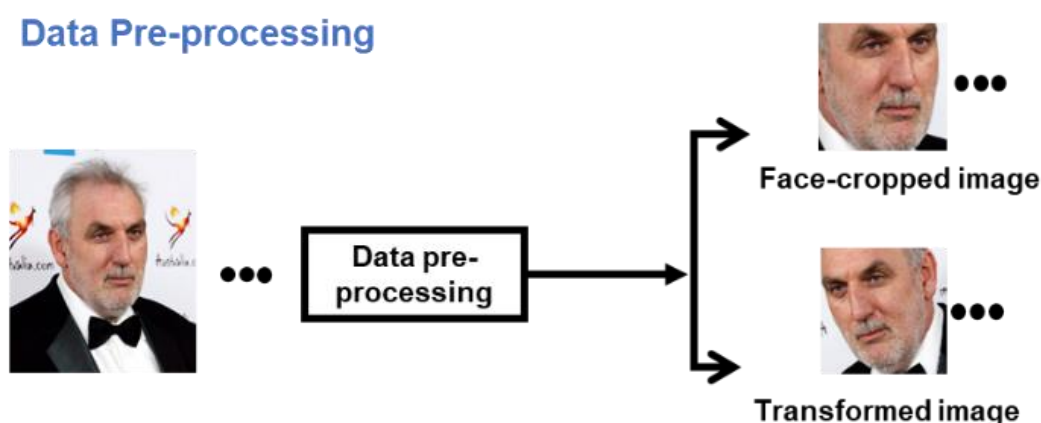


Figure 4.18 A example image and results after data pre-processing

4.3.2 Model training of the SDU

Model Structure

Having the two kinds of images as the input of the network, Stacked Dense U-nets (SDU) is comprised of the initial phase, the network backbone, the inside transformer, and the loss evaluation, as illustrated in Figure 4.19. The figure is inspired based on the interpretation in this research.

The initial phase is used to decrease the scale of the feature maps from 128x128 to 256x256 while increasing the channel number from 3 to 256. Moreover, the initial phase uses two kinds of residual blocks, including Hierarchical, Parallel and Multi-scale (HPM) (Adrian et al., 2017) and Channel Aggregation Block (CAB) (Guo et al., 2018). In the following, a max-pooling operator is utilised to reduce the scale of the feature maps.

Two dense U-nets are stacked as the network backbone, each of which follows an inside transformer. A dense U-net proposed a Scale Aggregation Topology (SAT) and a CAB for network design (Guo et al., 2017). SAT is built upon a topology which is symmetric in scale, based on the topology of the U-net (Ronneberger et al., 2015) and Hourglass (Alejandro et al., 2016). For the same insight in the network topology, a CAB is built upon a residual block for the SAT, which is symmetric in the channel.

Inside transformer employs the deformable convolution (Dai et al., 2017) to generate the landmark heatmap by learning additional offsets, which can replace the global explicit parametric transformation, such as the connection layer in DAN. Afterwards, the input of the second dense U-net stack the output feature maps from the initial phase, the landmark heatmaps and the feature maps of the inside transformer, which three different inputs add element-wise. The 68 heatmaps are the output of the first dense U-net using the deformable convolution, and then they increase the number of channels from 68 to 256. The second input is the feature maps before the output of the landmark heatmap. They both resample by the 1x1 convolution.

Finally, the 68 predicted landmark heatmaps and the ground truth are shown to minimise the novel loss function.

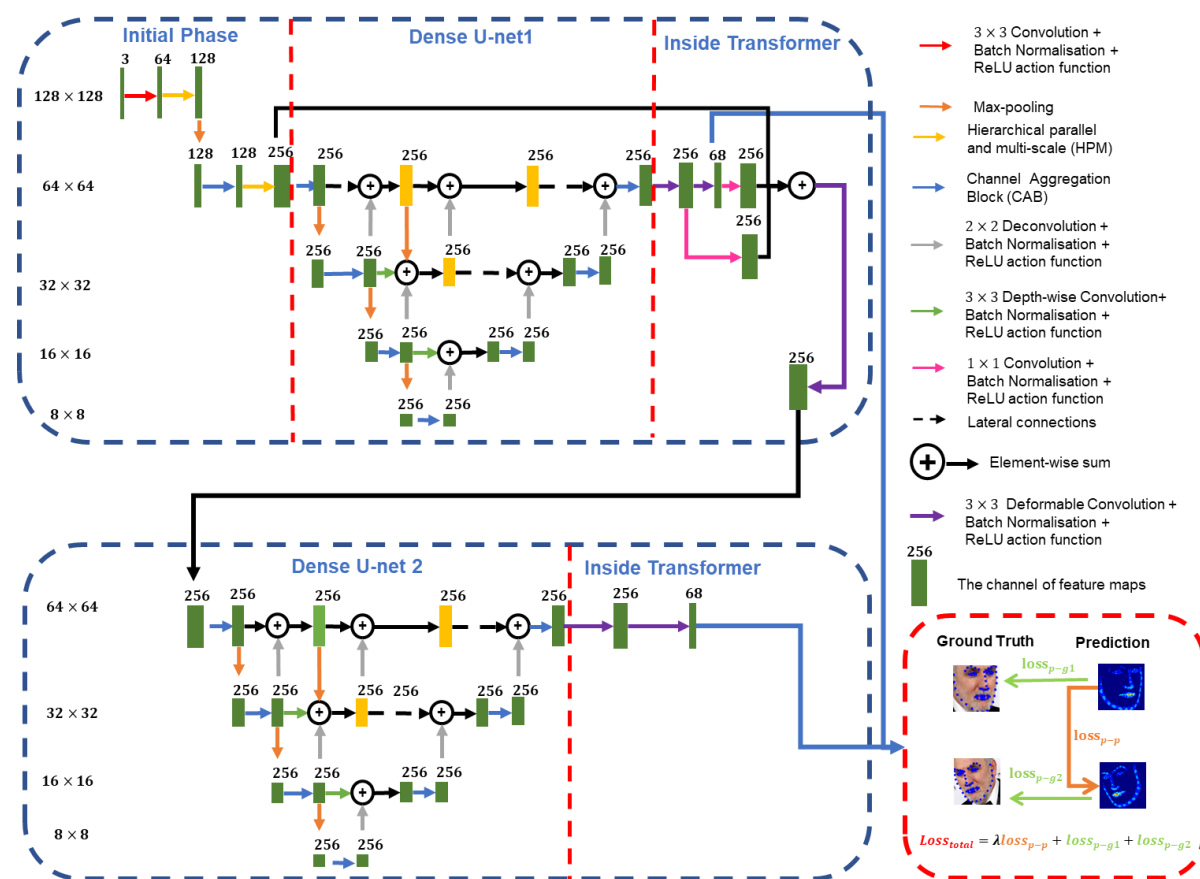


Figure 4.19 SDU model structure

Dense U-net

Having the feature map from the initial phase, SAT illustrates the bottom-up and top-down processing to capture the local and global features at four level scales, as shown in Figure 4.20. In the bottom-up processing, the down-sampling steps reduce the scale of feature maps from high to low, which repeatedly uses the max-pooling operator. There are three aggregation nodes, and the feature maps from different scales can be merged their spatial information using lateral connections. During each down-sampling step, the network branches off the feature maps, which are combined into the corresponding up-sampling aggregation nodes. Moreover, the aggregation node also adds down-sampling input for the aggregation nodes. At each scale, a dense U-net uses lateral connections to preserve the spatial information of the feature map (Yellow box). The up-sampling step expands the scale of feature maps by using a deconvolution operator. The deconvolution operator can identify as the reverse of convolution in mathematics (Zeiler et al., 2010). In addition, the

architecture is to use depth-wise separable convolutions (Andrew et al., 2017), which can improve the models' capacity.

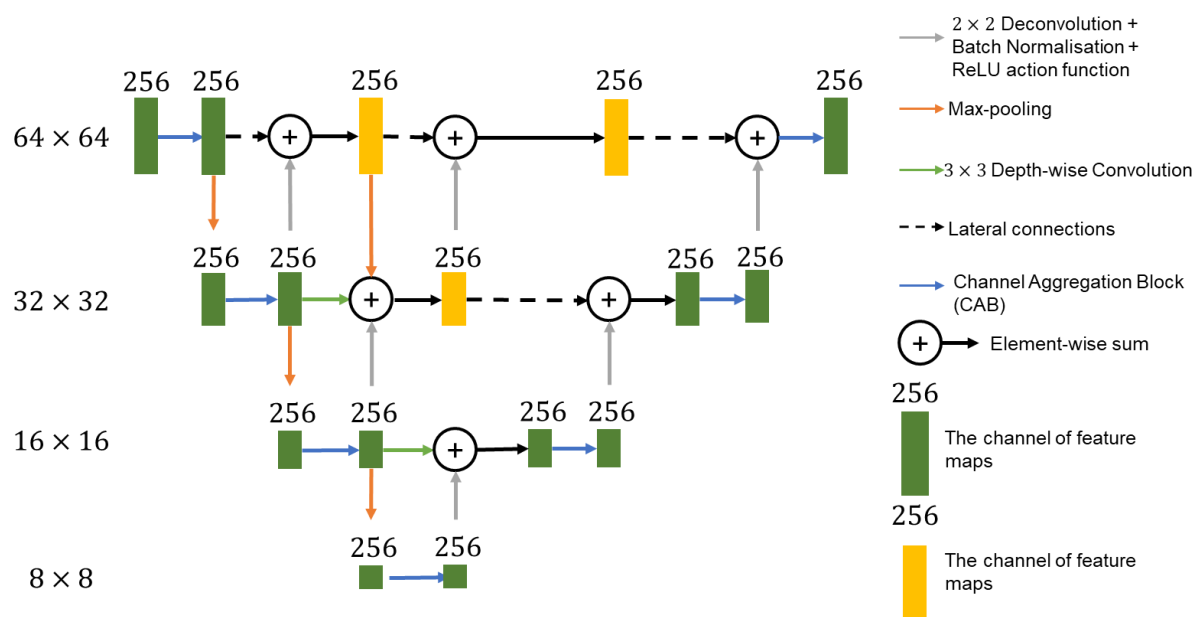


Figure 4.20 Scale aggregation topology

As seen in Figure 4.21, an output shape of a 3x3 depth-wise separable convolution is the same as a regular 3x3 convolution except for a different operation. The top architecture is a regular 3x3 convolution layer with a batch normalisation layer and ReLU action function, and the channel of input shape is 128. The kernel of the regular convolution has four dimensions: the number of kernels, the number of channels, and the kernel size. The regular convolution applies this kernel to multiply values over 3x3 spatial pixels over all the channels by 64 times. Then, the channel of the output shape is 64. The bottom architecture is a depth-wise convolution layer and a point-wise convolution layer, each following a batch normalisation layer and ReLU activation function. The depth-wise convolution convolves with a 3x3 kernel (2-D) for each channel. After that, the point-wise convolution uses 64 of 1x1 kernels to stack the output of each channel together. Finally, the channel of the output shape is 64, which is the same as a regular convolution.

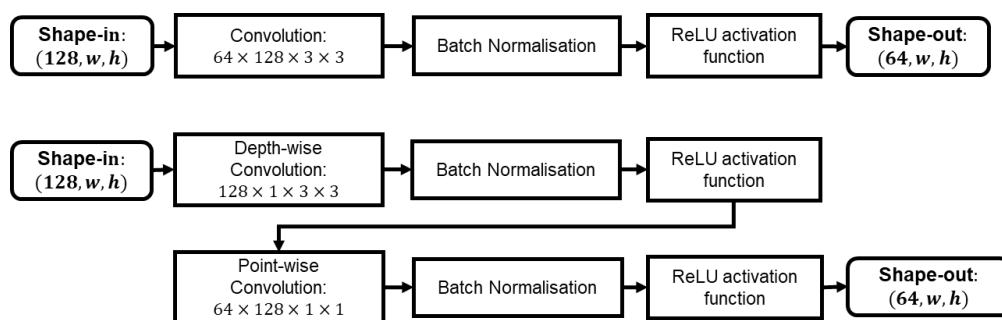


Figure 4.21 Comparison of a regular 3×3 convolution (Top) and a 3×3 depth-wise separable convolution (Bottom)

Residual Block

The Hierarchical, Parallel, and Multi-scale (HPM) residual-based architecture is explored in Figure 4.22(a). This residual block is composed of 4 convolutional layers, and the input feature map branches off two paths. Firstly, the 1x1 convolution operator is applied to a lateral connection to retain the original information from the input shape, while it also increases the number of channels from 128 to 256 to match the output channels' number. Secondly, three 3x3 convolutional operators extract spatial information from different channels, and it needs to match the input feature map's channels. The first 3x3 convolutional layer keeps the same channel number which is 128. The second 3x3 convolutional layer reduces the channels from 128 to 64. The third 3x3 convolutional layer also keeps the same channel number which is 64. Then, three different number channels are concatenated together to match the channel number of the original input shape. Finally, the two paths' feature map can add element-wise, based on the same number of channels, width and height.

Meanwhile, an architecture of Channel Aggregation Block (CAB) is introduced in Figure 4.22(b). This residual block is a symmetric structure in the channel and has four different sizes of the channel in this block. The 3x3 convolution operator is used to decrease the number of channels. Before each 3x3 convolutional layer, it separates two connections path. The first one is 'self-concatenation' used to retain the information of the channel with the current size and concatenate itself again to match the channel number of the previous layer. The other one uses a 3x3 depth-wise separated convolutional layer to extract the information of the channel with the current size. When increasing the channel number, the two different connections should add element-wise to the main path based on the same number of channels, width and height.

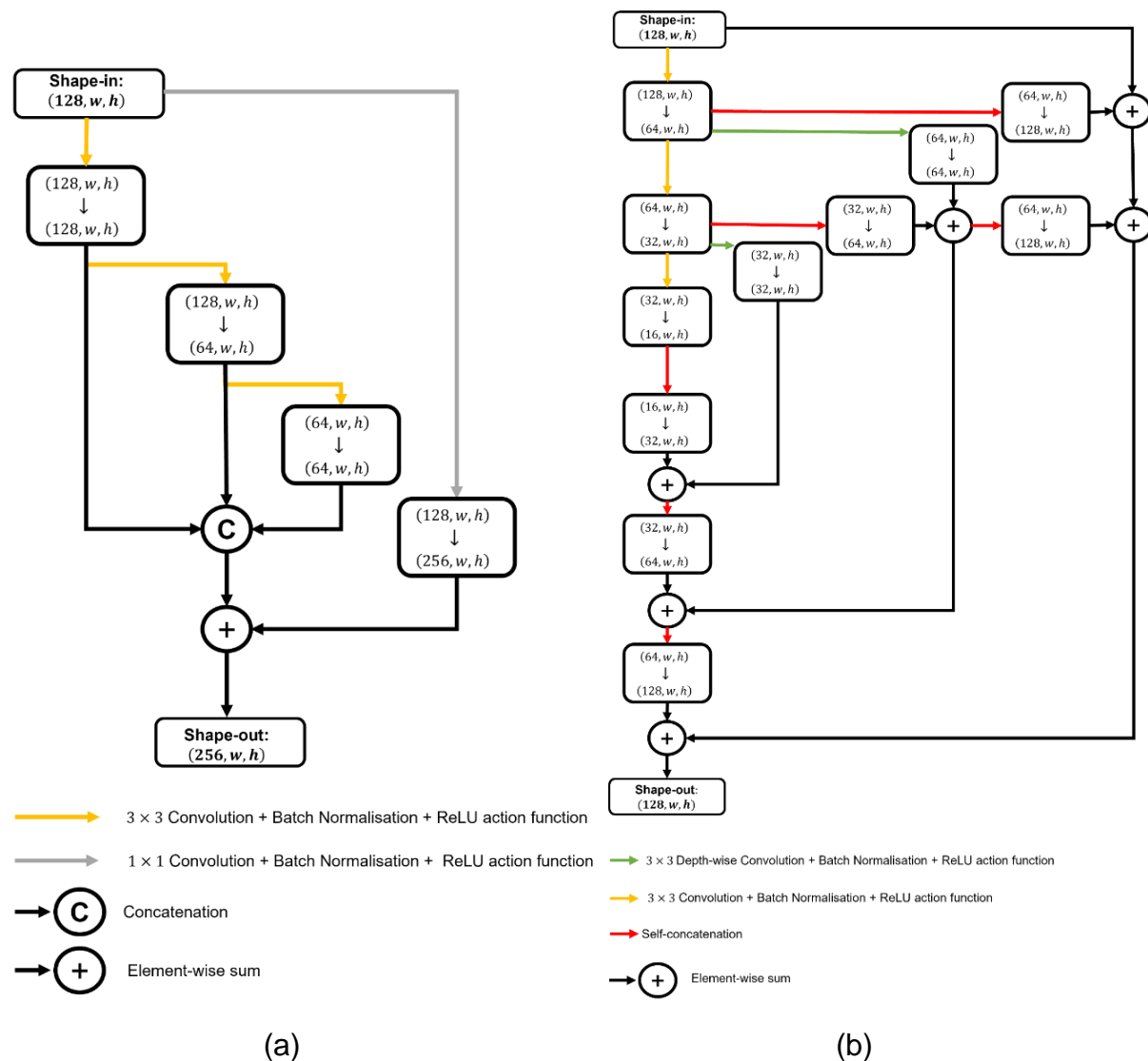


Figure 4.22 Different residual-based architecture. (a) HPM, (b) CAB

Inside Transformer

The feature maps are applied to deformable convolution (Dai et al., 2017) to learn addition offsets. Compared to the standard convolution, the deformable convolution adds 2-D offsets during the sampling location from the offset field, as illustrated in Figure 4.23 (Dai et al., 2017). The offsets are learned from the input feature maps by applying an additional convolutional layer. Then, the output feature maps can be deformed in a local, dense, and adaptive manner.

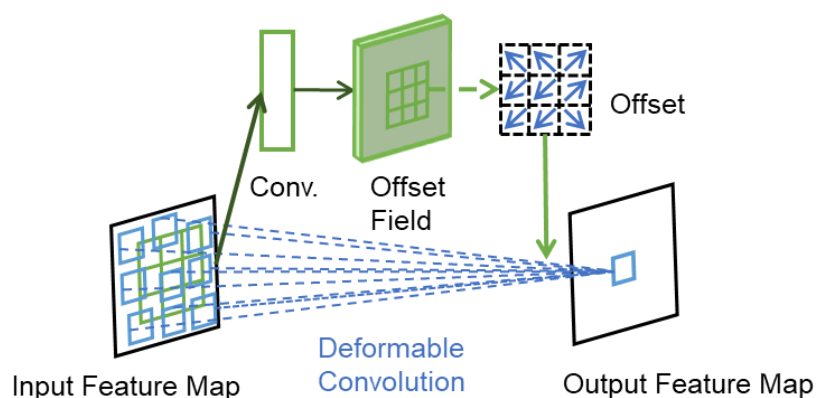
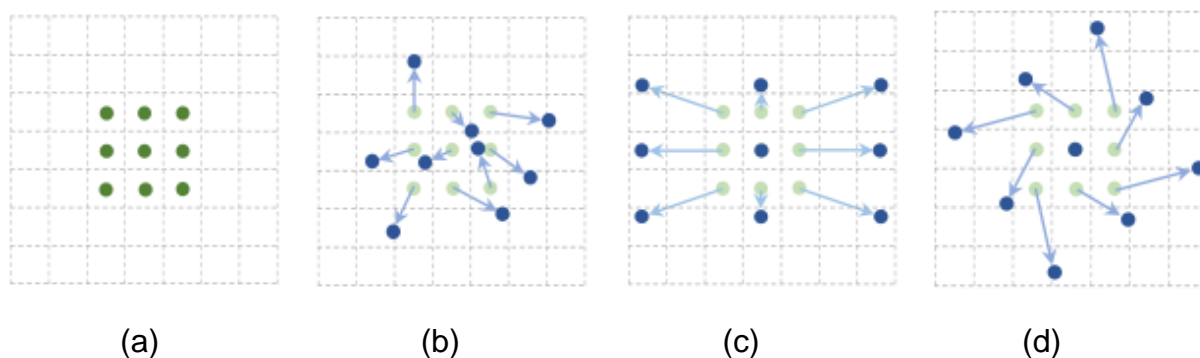


Figure 4.23 The deformable convolution

The deformable convolution can implicitly remove the discrepancy of rigid transformation on the input face image, such as scale, rotation, and translation. The examples in Figure 4.24 (Dai et al., 2017) illustrate the sampling location in the 3×3 standard and deformable convolutions. Figure 4.24(a) shows the regular sampling location in the standard convolution, where are the green points. Figure 4.24(b) shows the deformed irregular sampling location (blue points) and augmented offsets (blue arrows) in deformable convolution. Moreover, Figure 4.24(c) and (d) are the special cases of (b), demonstrating that the deformable convolution can be utilised for different spatial transformations, such as scale and rotation.

Figure 4.24 Examples of the sampling location in 3×3 standard and deformable convolutions.

4.3.3 SDU Loss evaluation

Loss evaluation of the SDU is to use the loss function to evaluate the training model parameters. The optima model parameters will be saved, and how to save is the

same as DAN. The network encourages the coherent loss to output the coherent landmarks when there are rotation, scale, and flip transformations (Guo et. 2018).

The loss function consists of two parts. The first part is the 68 landmark heatmaps difference between the two predictions: the face-cropped image with the corresponding transformed image. The other is the discrepancy of 68 landmark heatmaps between the predictions and annotations. Thus, the end-to-end model minimises the coherent loss using the following equation:

$$\mathbf{loss}_{total} = \frac{1}{\mathbf{n}} \sum_{\mathbf{n}=1}^{\mathbf{n}} \left(\gamma \frac{\|\mathbf{H}_{\mathbf{n}}(\mathbf{M}\mathbf{I}) - \mathbf{M}\mathbf{H}_{\mathbf{n}}(\mathbf{I})\|_2^2}{\mathbf{loss}_{p-p}} + \frac{\|\mathbf{H}_{\mathbf{n}}(\mathbf{I}) - \mathbf{G}_{\mathbf{n}}(\mathbf{I})\|_2^2}{\mathbf{loss}_{p-g_1}} + \frac{\|\mathbf{H}_{\mathbf{n}}(\mathbf{M}\mathbf{I}) - \mathbf{M}\mathbf{G}_{\mathbf{n}}(\mathbf{I})\|_2^2}{\mathbf{loss}_{p-g_2}} \right) \quad (4.26)$$

where \mathbf{I} is the input image, \mathbf{n} is the number of landmarks, $\mathbf{G}_{\mathbf{n}}$ is the ground truth of the \mathbf{n}^{th} landmark, $\mathbf{H}_{\mathbf{n}}$ is the predicted landmark heatmap of the \mathbf{n}^{th} landmark, \mathbf{M} is the matrix of similarity transformation, and γ is the tunable parameter to balance two losses: \mathbf{loss}_{p-p} is the difference between the two predicted landmark heatmaps before and after transformation, \mathbf{loss}_{p-g_1} or \mathbf{loss}_{p-g_2} is the difference between the prediction and ground truth.

4.3.4 Model testing of the SDU

In this section, the test stage of SDU jointly uses the SCRFD face detector and facial landmark detector. The face detector uses the same pre-trained model and image processing as DAN, while the facial landmark detector uses the SDU model to infer the landmark locations.

After that, the facial landmark detector also uses the trained model parameters and infers the 68 predicted landmark heatmaps. As seen in Figure 4.24, each heatmap represents a single landmark, and the landmark position is the pixel location with the highest intensity. Then, the 68 landmark coordinates need to transform back to match the original image using the inverse similarity transformation matrix and (4.5).

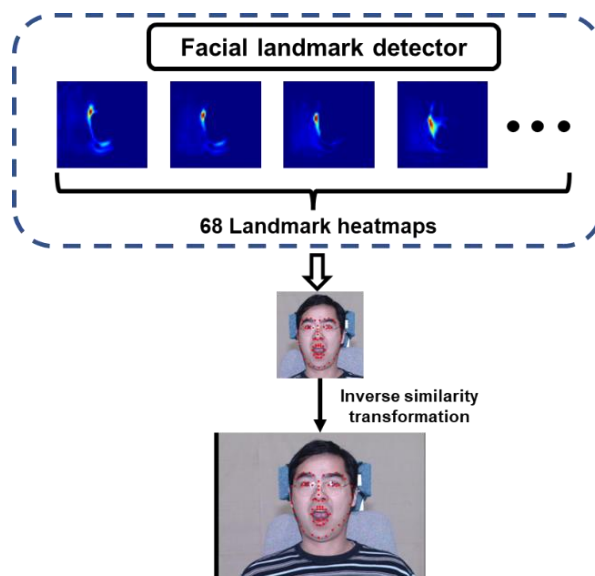


Figure 4.25 Example landmark prediction of the stacked dense U-net

For a more detailed comparison between different methods, Chapter 5 will discuss each method's performance using the evaluation metrics.

5. Experiment

After introducing the datasets and algorithms, this chapter provides a detailed comparison of automatic facial landmark detection.

The workflow of the performance evaluation is shown in Figure 5.1. The model testing stage uses the original image of the testing set and model parameters to present the predicted landmarks, which are shown as the red points on the example image. The model parameters estimate in the model training, and the images for model training are not included in the model testing. The example image's ground truth is blue points, which are utilised to compare with the predicted landmarks. Therefore, the testing set can perform the comparative results of three methods using different metrics.

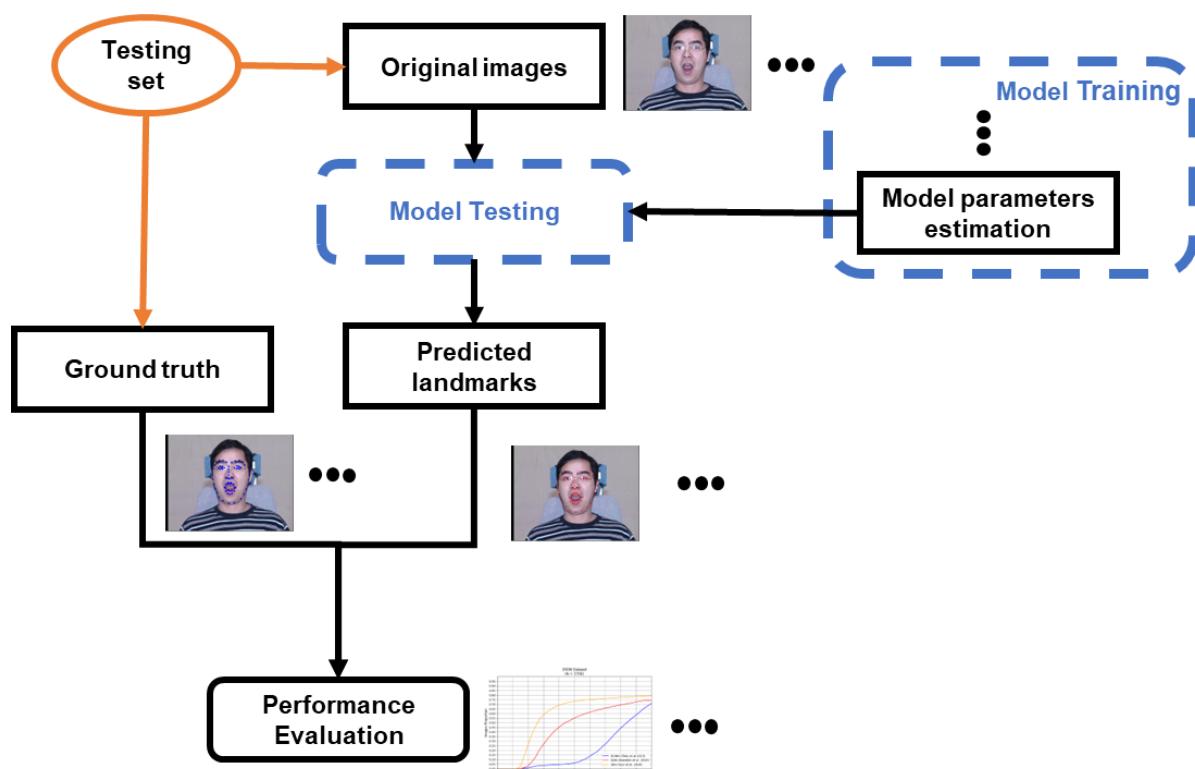


Figure 5.1 The diagram of the performance evaluation

Hence, the training detail of each algorithm will introduce in Section 5.1. Then, the evaluation metrics used for the facial landmark detector will present in Section 5.2. Finally, Section 5.3 will analyse the comparative result of each facial landmark algorithm.

5.1 Training details of three algorithms

The ten-fold cross-validation, as proposed by Kohavi (2001), was utilised for training, and testing the dataset in each algorithm. As shown in Figure 5.2. the method randomly divides a dataset into ten subsets, each containing an equal number of images. As an illustration, the first nine folds of the ten folds constitute the training and validation sets utilised for model training, with a 7:3 split. The remaining fold is used as the test set for model evaluation. Subsequently, it is recommended to perform ten repeats of the cross-validation procedure to ensure that each fold is utilised once as the test set. The result is that the ten outcomes are summed up and then averaged to produce an overall result.

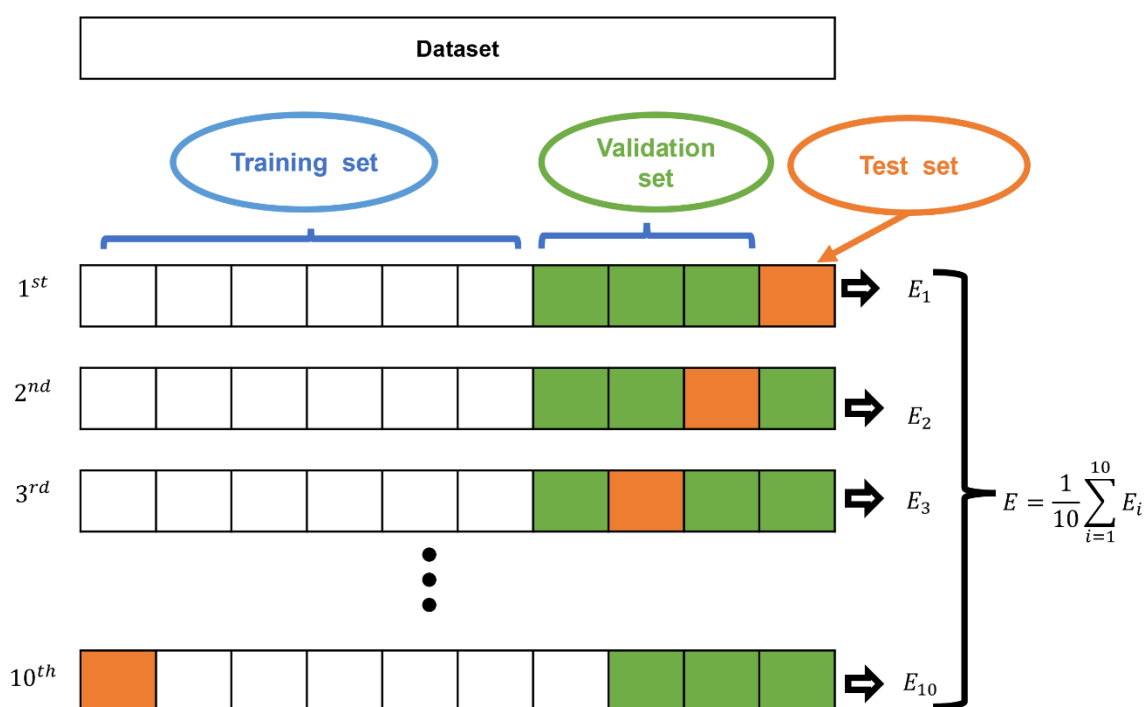


Figure 5.2 Ten-fold cross-validation workflow

Therefore, we trained ten models of each algorithm, each of which used a single dataset. Each optima model parameter will save when the error converges, and the max iteration reaches. To evaluate three different algorithms, we perform experiments on the three datasets consisting of the Multi-PLE dataset, 300W dataset, and Menpo dataset, as we introduced in Chapter 3.

The networks of DCNN were trained using MATLAB with stochastic gradient descent, with an initial learning rate of 0.0001 and a batch size of 64 (Zhou et al., 2013). The

networks of coarse level trained 100 epochs, and the networks of refine level trained 50 epochs. The cost time of the whole DCNN model training stage was 4 hours. And DAN and SDU are implemented using Python. DAN's networks were trained using TensorFlow 1.14.0 framework with Adam optimiser, an initial learning rate of 0.0001, and a batch size of 16 (Kowalski et al., 2016). The first-level network trained 15 epochs, and the second-level network trained 45 epochs. DAN's model training stage costs 6 hours. SDU was trained using MXNet framework with Adam optimiser, with an initial learning rate of 0.0002 and a batch size of 16 (Guo et al., 2018). The learning rate drops by 0.2 after 12 epochs and 18 epochs. The network trained 24 epochs, which cost 4 hours. All models trained on a GeForce GTX 2080ti.

5.2 Evaluation Metric for facial landmark detectors

Evaluation Metric can measure the performance of the facial landmark detection algorithm by comparing the difference between the landmark prediction and ground truths. Hence, we refer to three different evaluation metrics to measure the accuracy of landmarks, including Root Mean Squared Error (RSME), Normalised Mean Error (NME), and Cumulative Error Distribution (CED) (Sagonas et al., 2013).

The most straightforward metric is a root mean squared error (RMSE) to assess the performance of facial landmark localisation or facial landmark detection based on each transformed landmark position and the location of the corresponding landmark annotation. The metric computes the average Euclidean distance between the coordinates on each landmark basis. The higher average distance value can provide a lower landmark prediction accuracy because a landmark's position is far from the corresponding landmark annotations' location. The metric is shown as follows:

$$RMSE = \frac{1}{N} \sum_{i=1}^N \sqrt{(\mathbf{x}_i^p - \mathbf{x}_i^t)^2 + (\mathbf{y}_i^p - \mathbf{y}_i^t)^2} \quad (5.1)$$

where N is the number of the total landmarks, i is the i^{th} number of the landmark, $(\mathbf{x}^p, \mathbf{y}^p)$ the coordinate of the transformed landmark, and $(\mathbf{x}^t, \mathbf{y}^t)$ is the coordinate of the corresponding ground truth.

However, the metric cannot handle the errors caused by the individual face shape discrepancy. A better and more reliable way for the facial landmark detection assessment can be employed in terms of the normalised point-to-point root mean square error (NME) (Deng et al., 2019). The typical normalisation factor is the distance between two defined ground truth landmarks. The factor allows the performance evaluation independent of the discrepancy in individuals' face shape or the zoom factor of a camera. The metric is given as:

$$NME = \frac{1}{N} \sum_{i=1}^N \frac{\sqrt{(x_i^p - x_i^t)^2 + (y_i^p - y_i^t)^2}}{d_{norm}} \times 100 \quad (5.2)$$

$$d_{norm} = \sqrt{(x_{le}^t - x_{re}^t)^2 + (y_{le}^t - y_{re}^t)^2} \quad (5.3)$$

where (x_{le}^t, y_{le}^t) is the minimum coordinate of the ground truth, (x_{re}^t, y_{re}^t) is the maximum coordinate of the ground truth. Bounding box diagonal normalisation can express as the bounding box diagonal distance, which is the distance between the diagonal of the bounding box. Bounding box diagonal distance determine as the normalisation factor because other factors cannot give reliable evaluation metrics (Deng et al., 2019). For instance, the inter-ocular or inter-pupil distance becomes very small since some of the 2-D images in datasets are profile views.

The unit of NME refer to it as a percentage or a fraction of a specific length. The normalized mean error (NME) in facial landmark detection is reported as a percentage or fraction of the bounding box diagonal distance. It represents the average distance between the predicted facial landmarks and the ground truth landmarks, normalized by the distance between the diagonal of the bounding box.

The cumulative error distribution (CED) is employed to summarise the performance of the different models in the same dataset. The plot can illustrate the proportion of images comparing NME with less than a threshold β . For example, as illustrated in Figure 5.3, detection circles around the left eye's outer corner have been assessed where the radius equals the threshold β . The detection circles define the range with the threshold β , such as 10% (blue), 20% (red), and 30% (green). If the value of NME is below or equal to the current threshold, the position of the transformed landmark should exist in the detection circle.

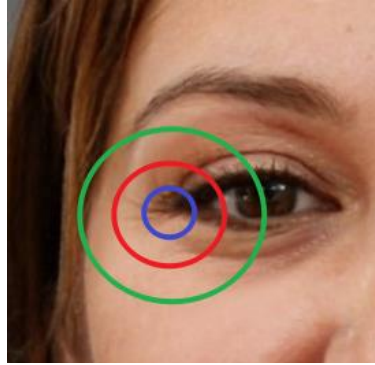


Figure 5.3 Example of the assessed detection circle

Furthermore, we calculate the Area Under the Curve (AUC), and the threshold is set as 0.1. We also calculate the failure rate of a dataset using each algorithm. Failure images can identify as any images with a threshold higher than 0.1. For the detailed evaluation, we separate 68 landmarks based on facial attributes. 'Contour' means the average NME of 17 landmarks at the facial contour. 'Brow' means the average NME of 10 landmarks at two brows. 'nose' means the average NME of 9 landmarks at the nose. 'Eye' means the average NME of 12 landmarks at two eyes. 'Mouth' means the average NME of 20 landmarks at the Mouth.

In addition, the 2-D controlled datasets and 2-D 'in-the-wild' datasets can be divided into different conditions, consisting of different expressions, illumination, heavy occlusion, and frontal face, as mentioned in Chapter 3. In order to compare the performance of three algorithms under these conditions, we sample 15 example images under each condition on three datasets to provide mean error and Standard Deviation (Std). The mean error is the average NME of all transformed landmarks. Then, the standard deviation is the square root of the average squared deviations between each landmark's NME and the mean error. Furthermore, the average of the NME is applied to analyse the difference between the transformed landmarks at different facial attributes. The following section could detail the performance of each algorithm by using these evaluation metrics.

5.3 Comparison result of three facial landmark algorithms

5.3.1 Test on 300W dataset

The cumulative error distributions of each method on the 300W dataset are provided in Figure 5.4, in which apply the x-axis assigned as the NME value of the current threshold, and the y-axis denotes the probability of correctly identified images. Compared with each algorithm's curve, SDU has the best performance, that above 99% of images can correctly identify at 0.04. DAN is the runner-up on the 300W dataset because the curve begins to increase after SDU. DCNN performed the worst since the curve started rising above 0.045, and only above 60% of the images can identify at 0.1.

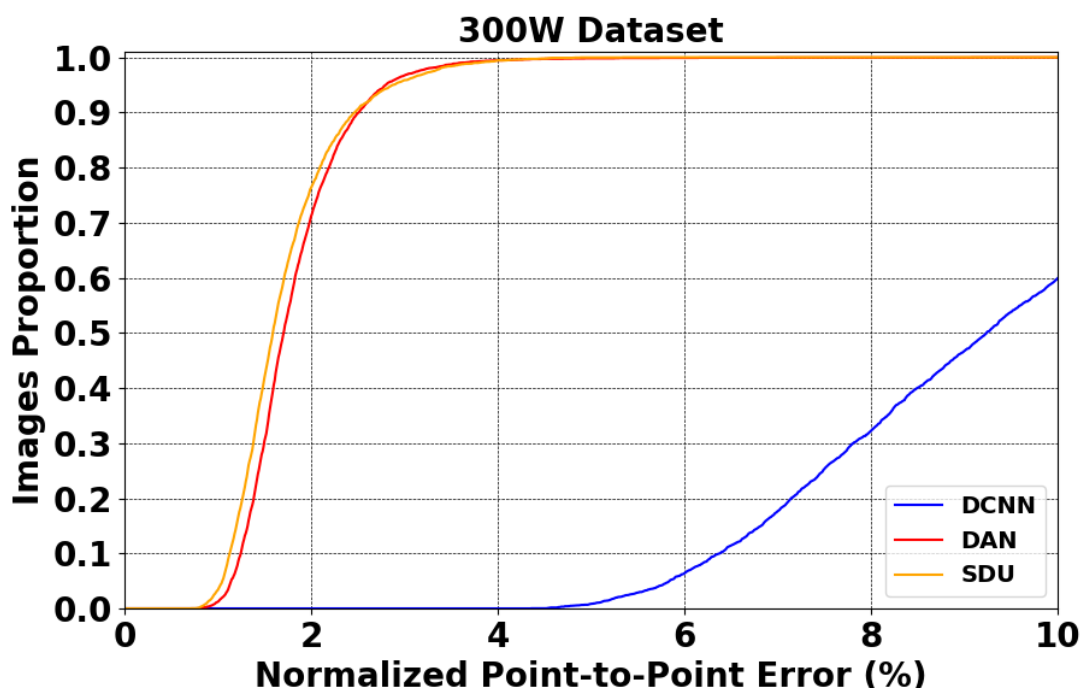


Figure 5.4 300W dataset cumulative error distributions

Furthermore, the metrics' performance of each algorithm on the 300W dataset is summarised in Table 5.1, that RMSE and NME use an average of all landmarks. SDU presents the least value of NME and the highest value of AUC compared to all evaluation metrics, which explains that SDU has the best performance among the three algorithms on the 300W dataset. DCNN shows the lowest performance based on these values of evaluation metrics.

Algorithms	RMSE	NME	AUC _(0.1)	Failure rate
DCNN	96.700	0.098	0.193	0.401
DAN	72.094	0.018	0.818	0.000
SDU	66.802	0.017	0.828	0.000

Table 5.1 The metrics' performance of the 300W dataset

In Figure 5.5, a comparison based on the facial attributes provides using box plots, which can visualise the NME distribution of each attribute on the 300W dataset. In each box, the green line is the median of the NME to split the original dataset into two subsets. Half the values are greater than or equal to the median, and the other half are less than the median. Then, a box's upper black edge line is the upper quartile, the median of the upper data subset. 25% of NME are above the upper quartile, and the range is called 'the upper whisker'. A box's lower black edge line is the lower quartile, the median of the lower data subset. 25% of the NME is less than the upper quartile. The middle box represents the middle 50% of the NME for the face attribute, called the 'interquartile range'. The upper black line is the maximum value, and the lower black line is the minimum value. Compared to the medians of the face contour's box, the median line of DAN is lower than SDU, and DAN's interquartile range is smaller than SDU. DAN show better performance on the face contour.

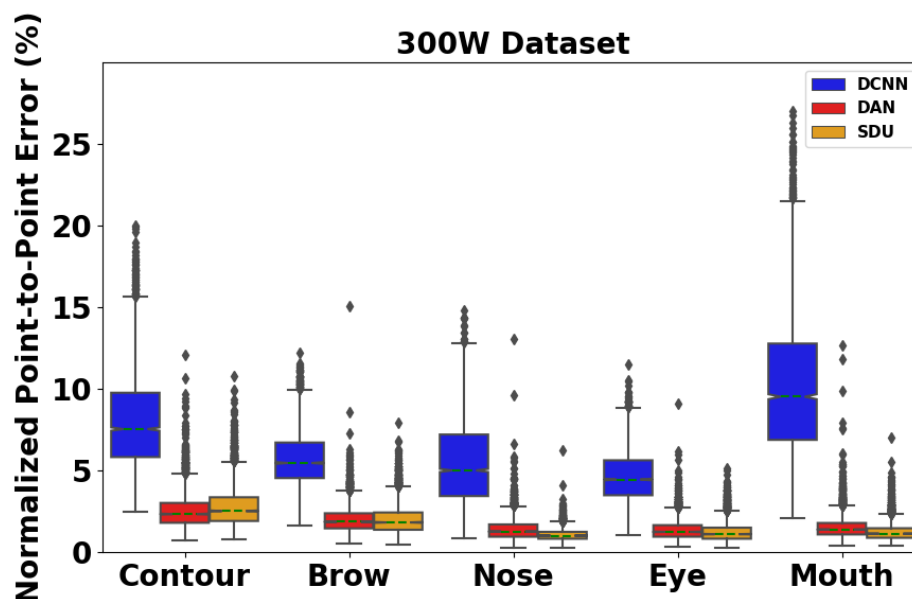


Figure 5.5 Comparison of landmark error due to different facial attributes on the 300W dataset

Furthermore, a more detailed comparison based on each landmark illustrates in Figure 5.6. Each bar represents an average NME of a landmark on the 300W dataset. Compared with DAN and SDU, SDU shows higher error from the 1st to the 22nd landmark, which are the landmark indexes of face contour and brows. Then, each landmark's NME of DCNN is the biggest.

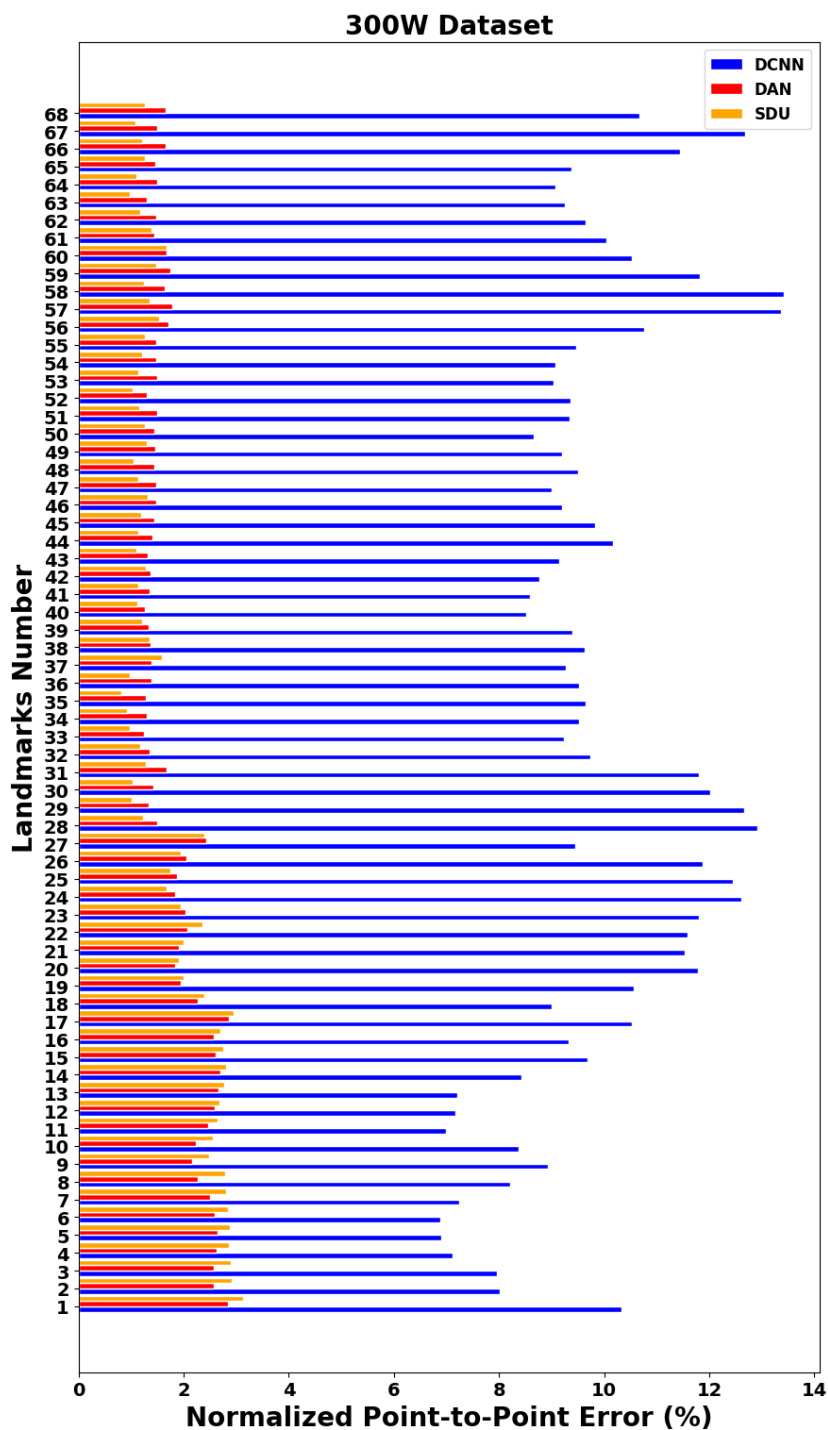


Figure 5.6 Comparison of each landmarks error on the 300W dataset

The reason is that DAN utilises a mean face shape, which can constraint the landmark would not have highly biased. Another reason is that the heatmap predictions of SDU are too sensitive to the colour information, which ignores the face contour need to satisfy the anthropological constraint.

In the following, three algorithms first tested 15 images with different expressions on the 300W dataset. DAN has the most outstanding performance in the image with different expressions on the 300W dataset, while SDU has the highest mean value among the three algorithms, as shown in Table 5.2.

Algorithms	Mean	Std
DCNN	0.02908	0.00014
DAN	0.02449	0.00009
SDU	0.05303	0.00010

Table 5.2 The statistical results on the 300W dataset with different expressions

For more detail, the predicted landmarks at different facial attributes calculate the mean value of the NME to compare the three algorithms, as shown in Figure 5.7. SDU has the highest average NME at different facial attributes, while DAN shows the best performance.

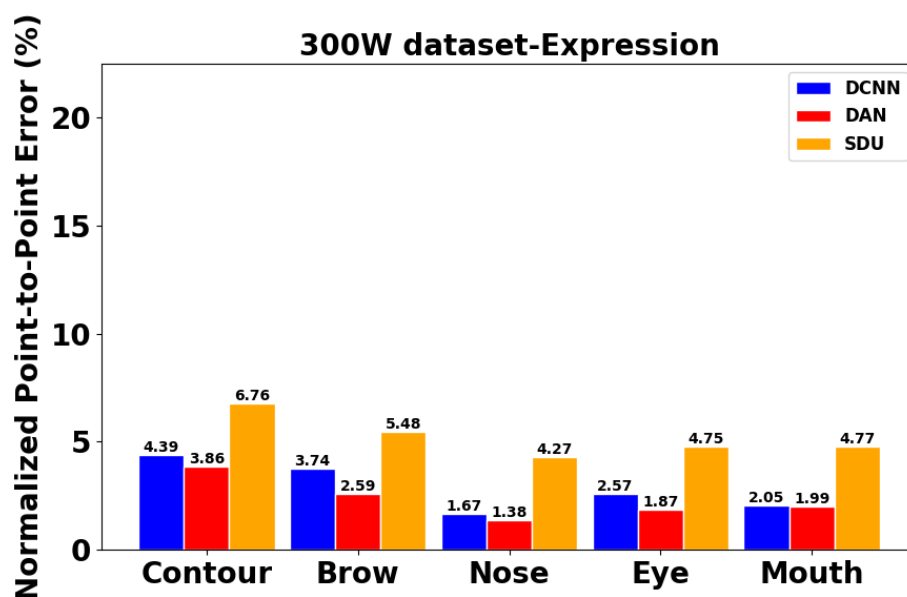


Figure 5.7 Comparison of landmark error due to the different facial attributes on the 300W dataset with different expressions

To explain the reason for the poor performance of SDU, the poor landmark localisation results of SDU are shown in Figure 5.8. There always be some landmarks that deviate significantly from the face contour and mouth. Because the landmark heatmaps of SDU predict the probability of landmarks' presence at each input image pixel, the face with an exaggerated expression will miss face contour sometimes.



Figure 5.8 Example of SDU poor landmark localisation

Secondly, three algorithms tested 15 images with different illumination conditions. DCNN presents the lowest performance, and the mean value of the NME is 0.03336. SDU performs best in that the mean value of the NME is 0.01772, as illustrated in Table 5.3.

Algorithms	Mean	Std
DCNN	0.03336	0.00006
DAN	0.02168	0.00006
SDU	0.01772	0.00008

Table 5.3 The statistical results on the 300W dataset with different illuminations

The NME at different facial attributes of the 15 images under different illuminations on the 300W dataset is shown in Figure 5.9, and DCNN can be observed, which provides the lowest performance.

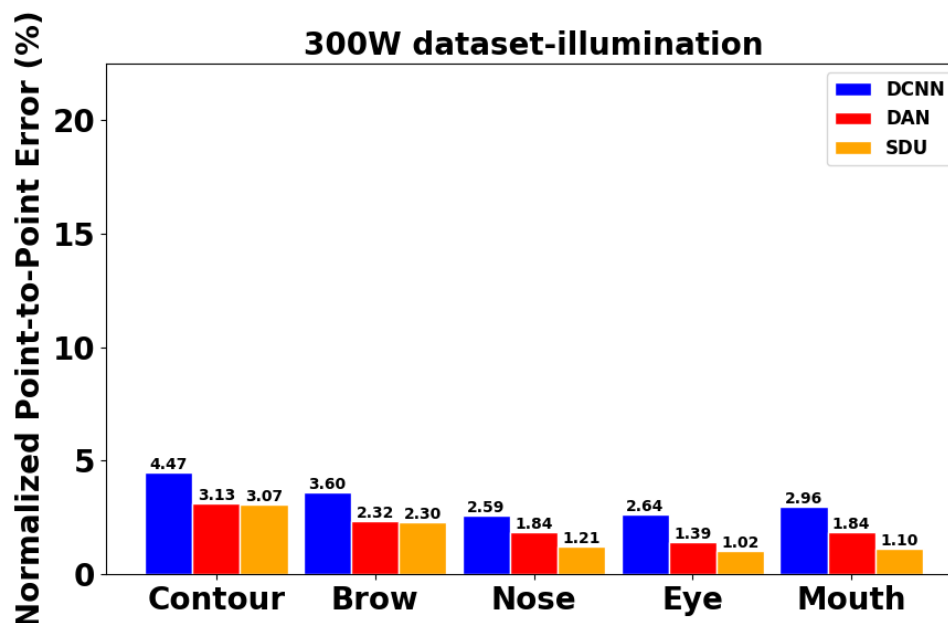


Figure 5.9 Comparison of the NME due to different facial attributes on the 300W dataset with different illuminations

In order to find out the limits of DCNN, example images with the predicted landmark localisation results are demonstrated shown in Figure 5.10. In DCNN, the facial attributes are employed to provide the final predicted landmarks. If the predicted landmark of the coarse level cannot be correctly identified, it will influence the extracted region of each facial attribute.

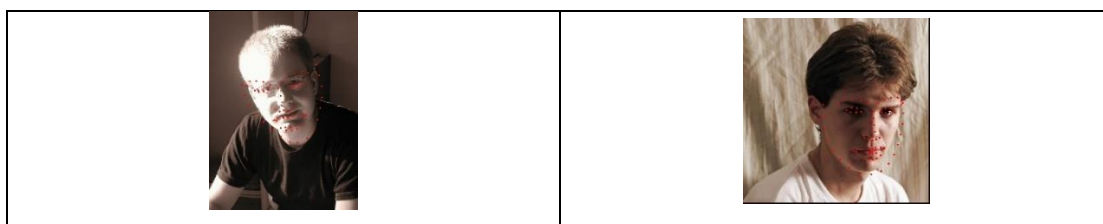


Figure 5.10 Example of DCNN poor landmark localisation

Thirdly, the tested images are under occlusion conditions, and SDU still has the least mean NME is 0.02193 among the three algorithms in Table 5.4.

Algorithms	Mean	Std
DCNN	0.05413	0.00009
DAN	0.03751	0.00006
SDU	0.02193	0.00009

Table 5.4 The statistical results on the 300W dataset with heavy occlusion

The NME values at different facial attributes of the 15 images with occlusion conditions on the 300W dataset are shown in Figure 5.11. However, SDU has the best performance at different facial attributes of the three algorithms, while DCNN has the highest value of the NME.

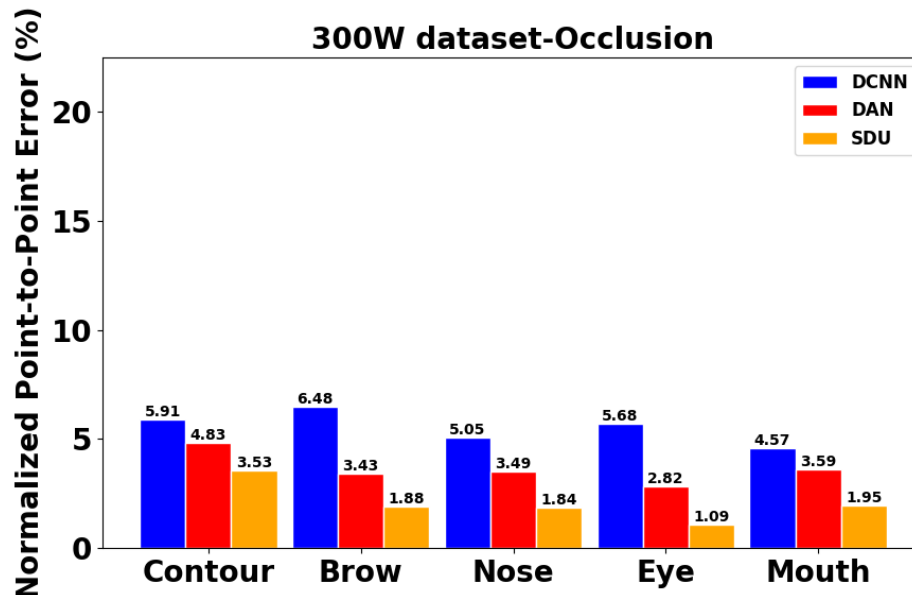


Figure 5.11 Comparison of landmark error due to different facial attributes on the 300W dataset with heavy occlusion

An example image with heavy occlusions uses three algorithms to test the landmark localisation results, as the image with the landmark localisation results of DCNN, DAN and SDU are demonstrated in Figures 5.12(a), (b), and (c). The predicted landmarks of DCNN and DAN have a high bias which means these two algorithms do not have a great ability to handle the face with heavy occlusion.

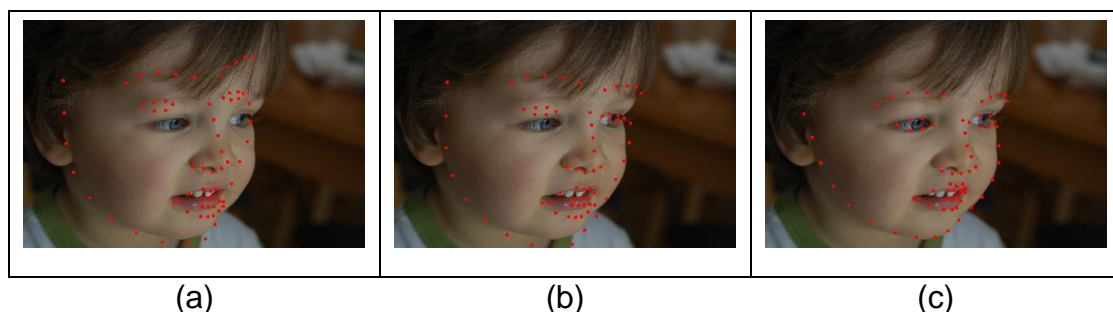


Figure 5.12 The landmark localisation results on the 300W dataset with heavy occlusion. (a) DCNN, (b) DAN, (c) SDU

Finally, 15 frontal-face images from the 300W dataset were sampled and tested the predicted landmarks of three algorithms. SDU has the least mean error in Table 5.5.

Algorithms	Mean	Std
DCNN	0.17179	0.00008
DAN	0.15883	0.00007
SDU	0.15104	0.00006

Table 5.5 The statistical results on the 300W dataset with frontal face

The NME values at different facial attributes for the 15 frontal-face images on the 300W dataset are shown in Figure 5.13. SDU is the best-performing algorithm in all face attribute categories.

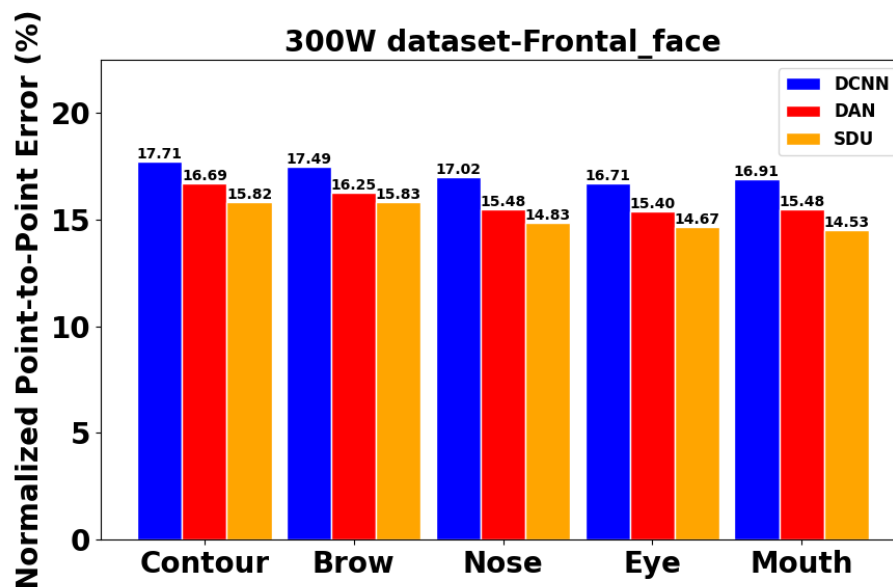


Figure 5.13 Comparison of landmark error due to different facial attributes on the 300W dataset with frontal face

5.3.2 Test on Menpo Dataset

The same experiments were repeated on the Menpo dataset. As illustrated in Figure 5.14, each algorithm provides the cumulative error distributions. The curves can observe that SDU illustrates the best performance that above 97% of images can identify at 0.04, while DAN reaches 95%. The curves' gap between DAN and SDU becomes significant because SDU can learn more rich feature information from the Menpo dataset, which is the most challenging dataset among the three tested datasets. The results of the metrics on the Menpo dataset are outlined in Table 5.6.

SDU still outperform overall algorithms, while DCNN shows the worst performance. DCNN's landmark output of the coarse level has a poor prediction, which will cause a higher biased on the predicted landmark of the refine level.

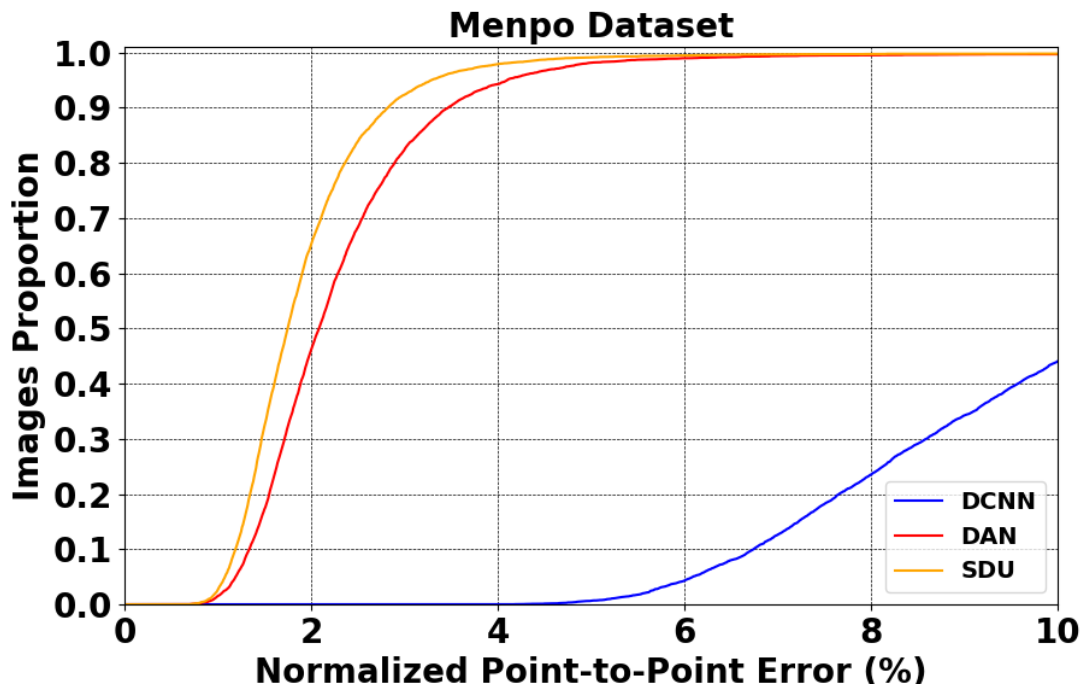


Figure 5.14 Menpo dataset cumulative error distributions

Algorithms	RMSE	NME	AUC _(0.1)	Failure rate
DCNN	29.466	0.116	0.197	0.560
DAN	11.429	0.023	0.769	0.003
SDU	7.254	0.020	0.807	0.002

Table 5.6 The metrics' performance of the Menpo dataset

The comparison performance on the Menpo dataset based on different facial attributes is shown in Figure 5.15. The median of each SDU's face attribute box is the lowest, which explains that the SDU is the best performer. The median and range of interquartile of each DAN's face attribute box are smaller than SDU.

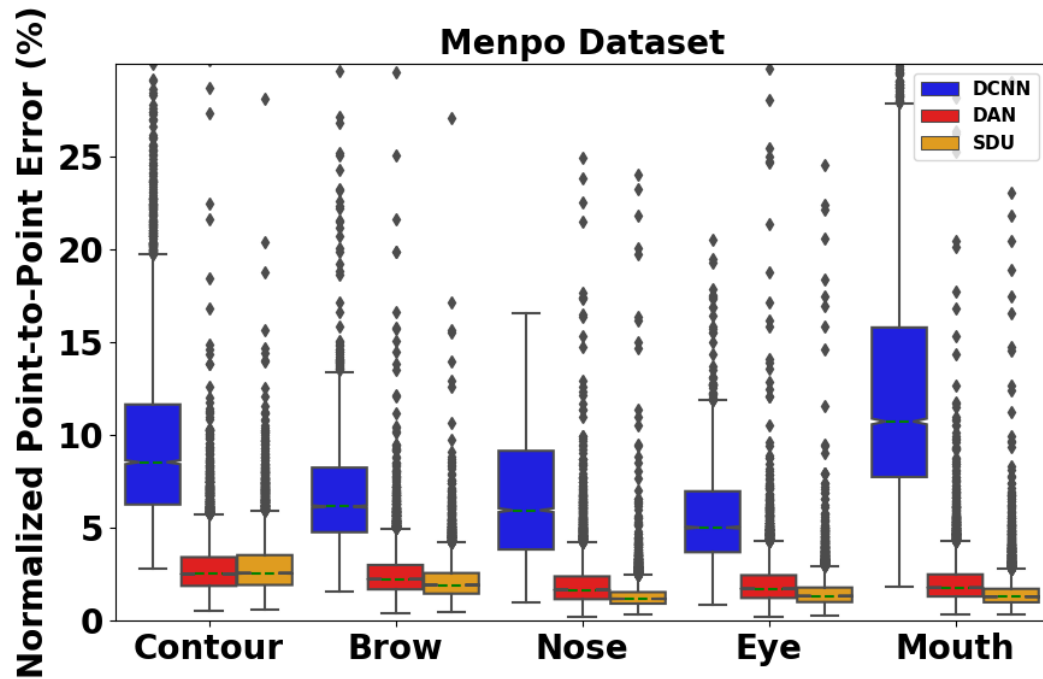


Figure 5.15 Comparison of landmark error due to different facial attributes of the Menpo dataset

The NME values for each landmark on the Menpo dataset are illustrated in Figure 5.16. SDU shows higher error from the 1st landmark to the 10th landmark, which are the landmark indexes of the left side face contour, while each landmark's NME of DCNN is still the biggest. The reason is the same that we tested in the 300W dataset.

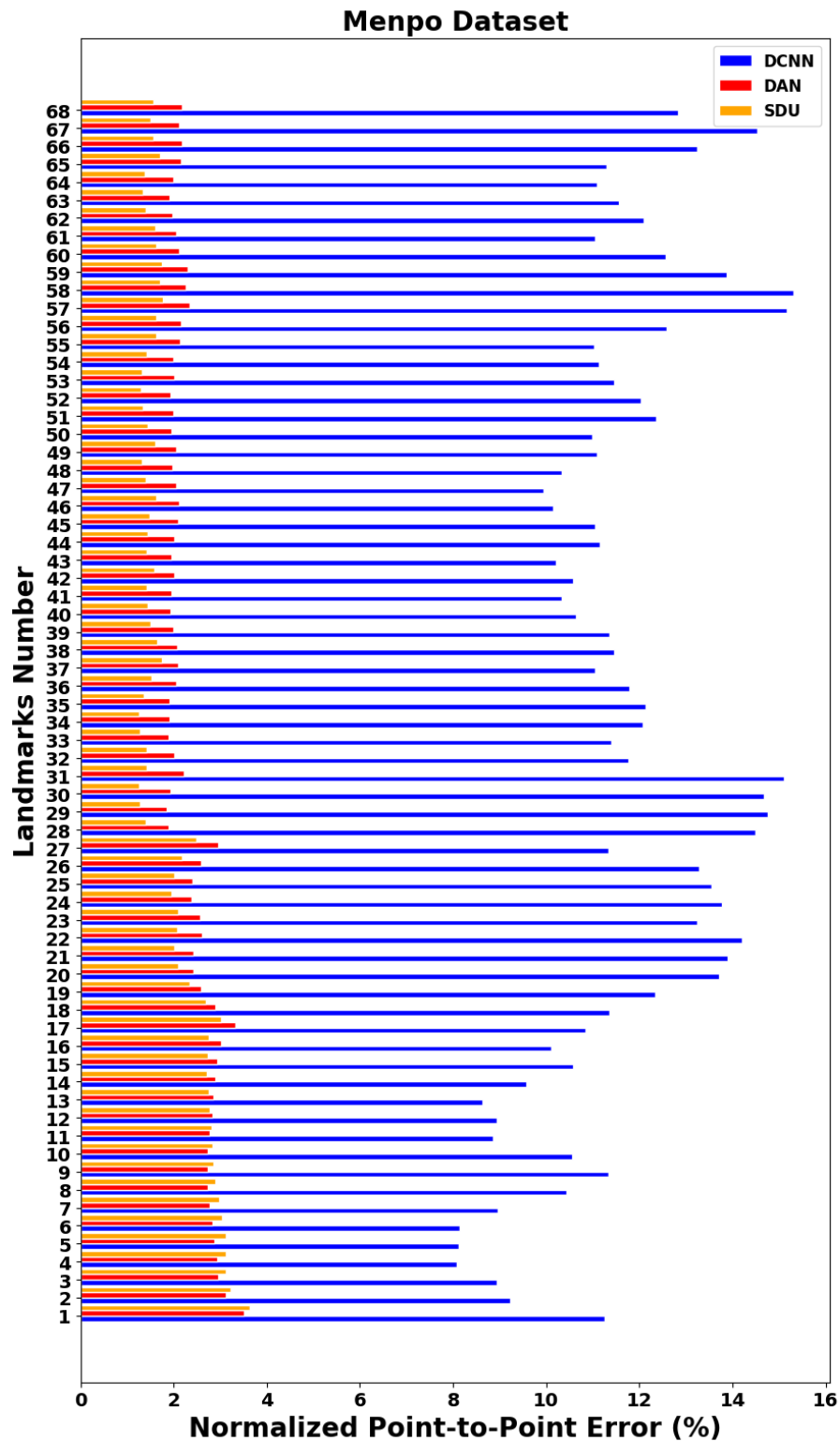


Figure 5.16 Comparison of each landmarks error on the Menpo dataset

Similar to the experiment on the 300W dataset, the performance of the three algorithms under the different conditions was evaluated as each condition uses 15 example images on the Menpo dataset.

Firstly, three algorithms test 15 images with different expressions on the Menpo dataset. As shown in Table 5.7, SDU has the best performance among the three algorithms.

Algorithms	Mean	Std
DCNN	0.03751	0.00011
DAN	0.02254	0.00003
SDU	0.01820	0.00009

Table 5.7 The statistical result on the Menpo dataset with different expressions

The NME values at different facial attributes of the 15 images with different expressions on the Menpo dataset are shown in Figure 5.17. DCNN has a significantly poor performance of the landmarks at the mouth again, and SDU is still worse than DAN at the face contour.

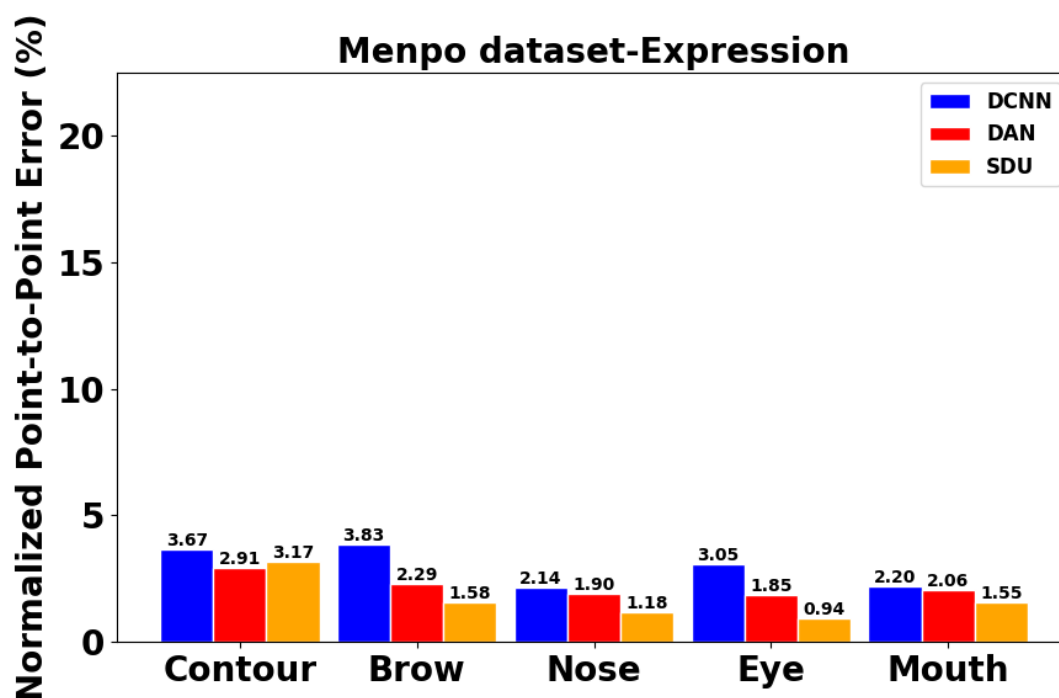


Figure 5.17 Comparison of landmarks error due to different facial attributes on the Menpo dataset with different expressions

Secondly, three algorithms test 15 images under different illumination conditions. As shown in Table 5.8, DAN is the best-performing algorithm in the images of the Menpo dataset under different illumination conditions.

Algorithms	Mean	Std
DCNN	0.04191	0.00014
DAN	0.02691	0.00007
SDU	0.02785	0.00021

Table 5.8 The statistical results on the Menpo dataset with different illuminations

Then, as shown in Figure 5.18, DCNN has the highest NME on almost all facial attributes, and SDU has a poorer performance than DAN, where the landmarks are located at the contour, brow, and eye.

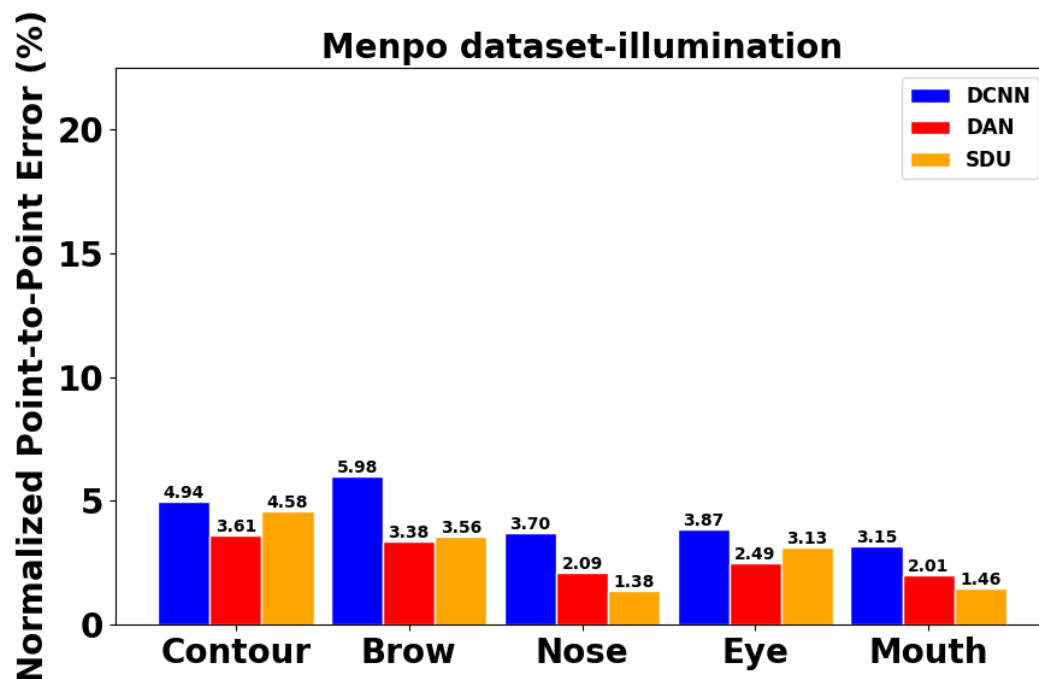


Figure 5.18 Comparison of landmarks error due to different facial attributes on the Menpo dataset with different illuminations

SDU still has poor landmark heatmaps when the images are under different illumination conditions, as shown in Figure 5.19. SDU has a highly biased predicted landmark when some facial attributes of the image are missing in an extremely dark environment. That can explain why landmark heatmap prediction heavily relies on the image's colour information.

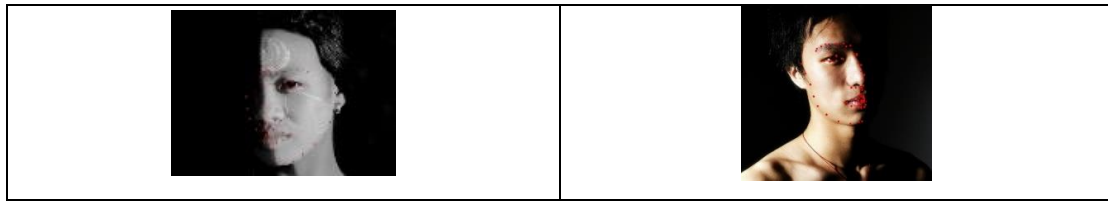


Figure 5.19 Examples of SDU poor landmark localisation results

In the following, three algorithms tested 15 images with heavy occlusion conditions. In Table 5.9, SDU has the best performance among the three algorithms, and the mean value of the NME is 0.03833.

Algorithms	Mean	Std
DCNN	0.07422	0.00049
DAN	0.04411	0.00006
SDU	0.03833	0.00047

Table 5.9 The statistical results on the Menpo dataset with heavy occlusion

The NME values at different facial attributes of the 15 images with heavy occlusion on the Menpo dataset are shown in Figure 5.20. SDU performs less than DAN, where the landmarks are at the face contour and brow.

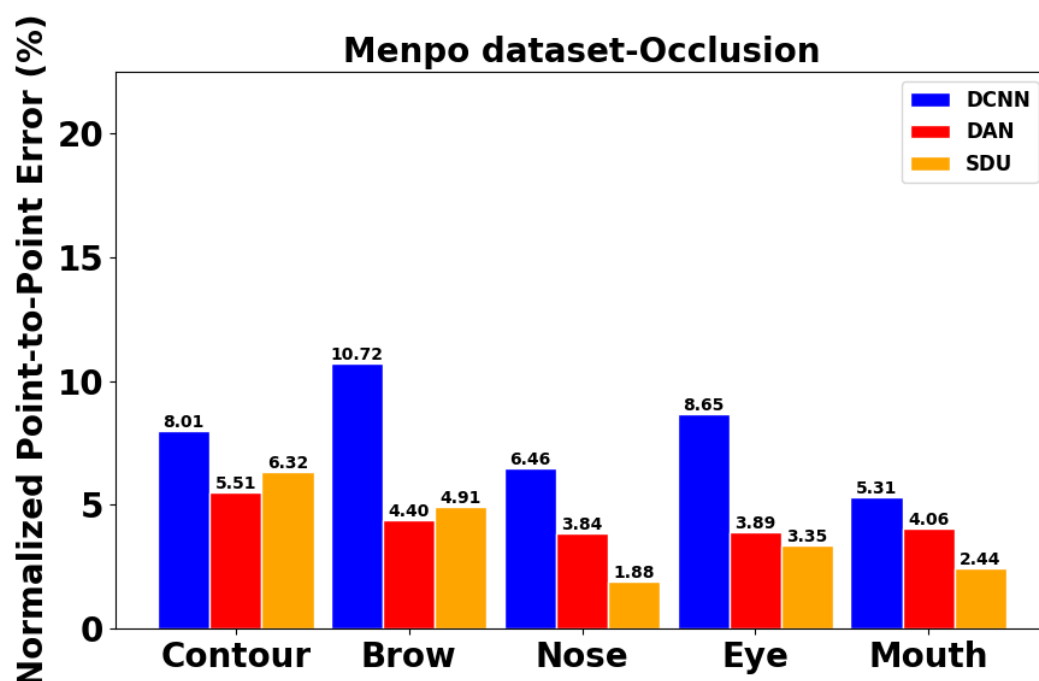


Figure 5.20 Comparison of landmarks error due to different facial attributes on the Menpo dataset with heavy occlusion

The example images with the 68 predicted landmark localisation are shown in Figure 5.21. Not only can SDU not recognise the situation where half of the face occluded by itself, but also DCNN and DAN perform poorly in self-occlusion. Hence, SDU performs poorly in predicting the image with the missing face attribute.

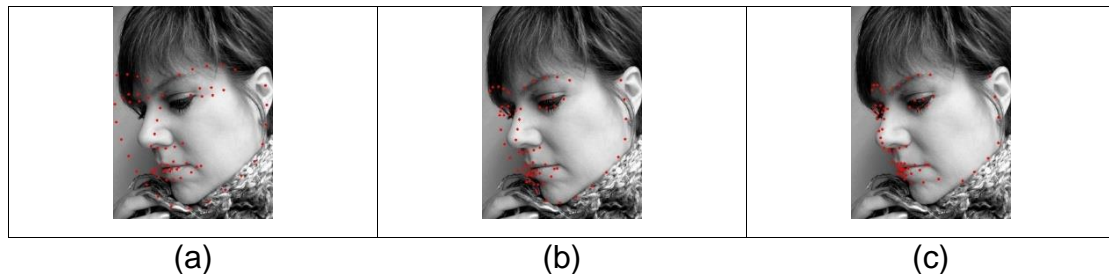


Figure 5.21 The poor landmark localisation results under heavy occlusion on the Menpo dataset with heavy occlusion (a) DCNN, (b) DAN, (c) SDU

Finally, the three algorithms tested the 15 frontal-face images on the Menpo dataset. The statistics results are shown in Table 5.10. SDU still performs best among the three algorithms, and the mean value of the NME is 0.014. DAN is the runner-up, and the mean value of the NME is 0.02212.

Algorithms	Mean	Std
DCNN	0.02573	0.0009
DAN	0.02212	0.0003
SDU	0.01400	0.0004

Table 5.10 The statistical results on the Menpo dataset with frontal face

The NME values of the predicted landmarks on each face attribute are demonstrated in Figure 5.22. It is similar to the test on the 300W dataset that SDU is the best performing in all face attribute categories on the frontal-face images of the Menpo dataset. In comparison, DAN considers the runner-up, which presents better performance than DCNN in all face attribute categories.

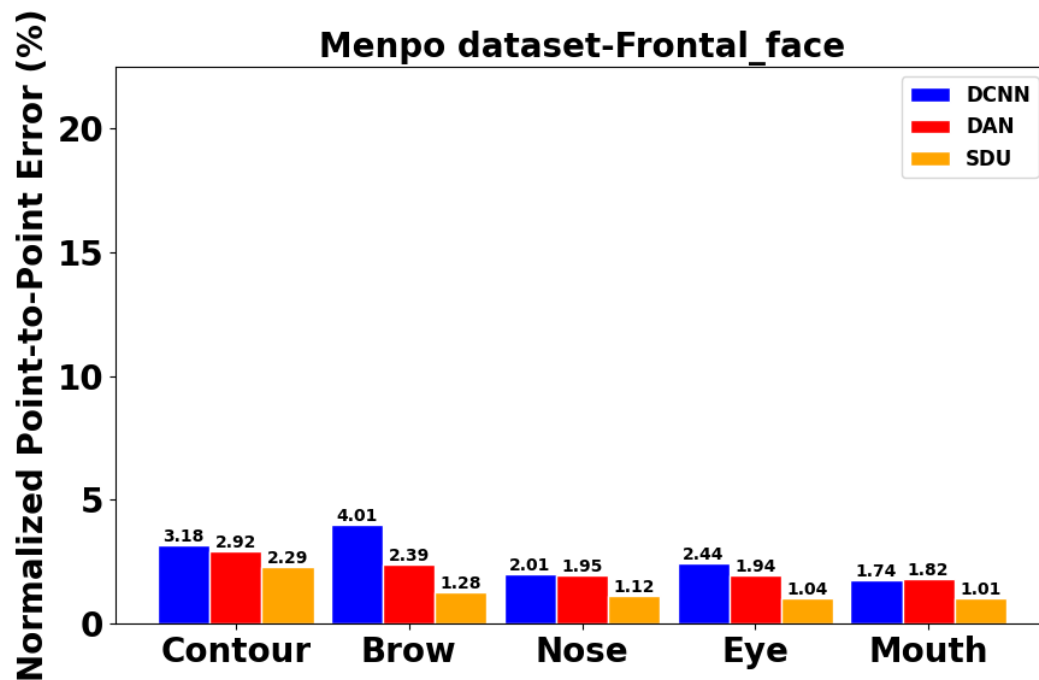


Figure 5.22 Comparison of landmarks error due to different facial attributes on the Menpo dataset with frontal face

5.3.3 Test on Multi-PLE Dataset

Again, the same experiments were repeated on the Multi-PLE dataset. The cumulative error distributions on the Multi-PLE dataset are reported in Figure 5.23. The curves can observe that SDU is not the best performer among all and that above 97% of images can correctly identify at 0.04. DAN is better than SDU, reaching above 99% at 0.04. Because DAN can constrain the landmarks' position based on the initial mean face shape, another reason is that the Multi-PLE dataset has a set of heavy self-occlusion images, which is an excellent challenge for SDU. Furthermore, DCNN always shows the lowest performance at the threshold of NME 0.1 because of the coarse-to-refine level structure. The metrics of each method on the Multi-PLE dataset performs in Table 5.11. DAN outperform overall algorithms, while DCNN still shows the worst performance.

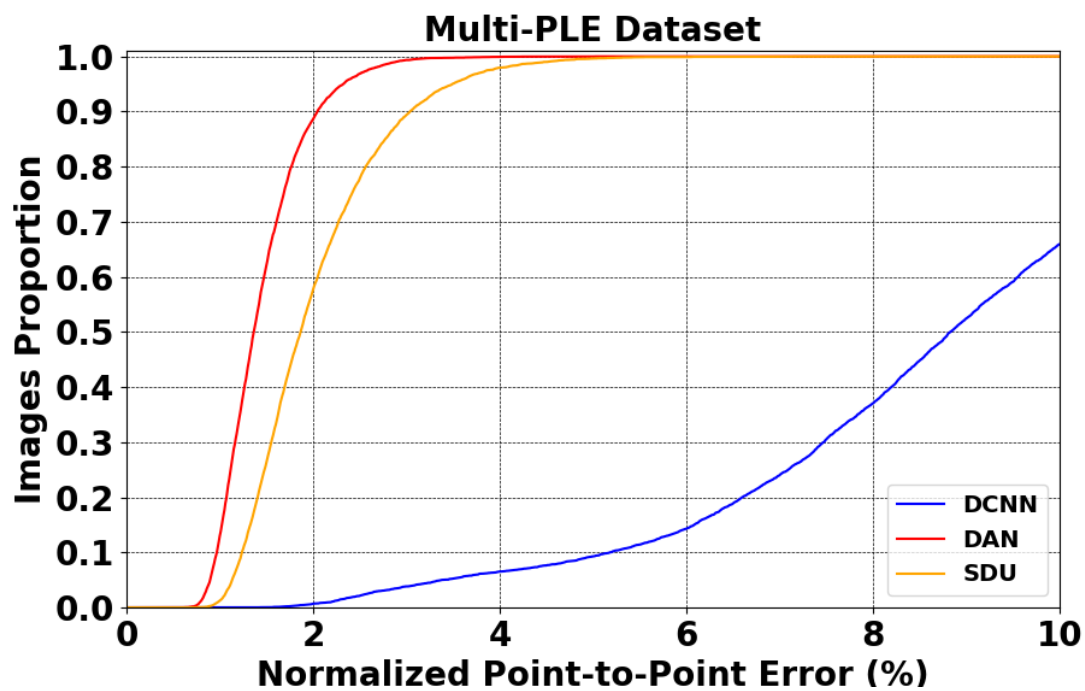


Figure 5.23 Multi-PLE dataset cumulative error distributions

Algorithms	RMSE	NME	AUC _(0.1)	Failure rate
DCNN	24.708	0.114	0.180	0.341
DAN	3.695	0.015	0.855	0.000
SDU	5.171	0.020	0.797	0.000

Table 5.11 The metrics' performance of the Multi-PLE dataset

In Figure 5.24, the comparison was performed on the Multi-PLE dataset based on different facial attributes. SDU has a bigger median of the face contour box than DAN because of the heavy self-occlusion image on the Multi-PLE dataset.

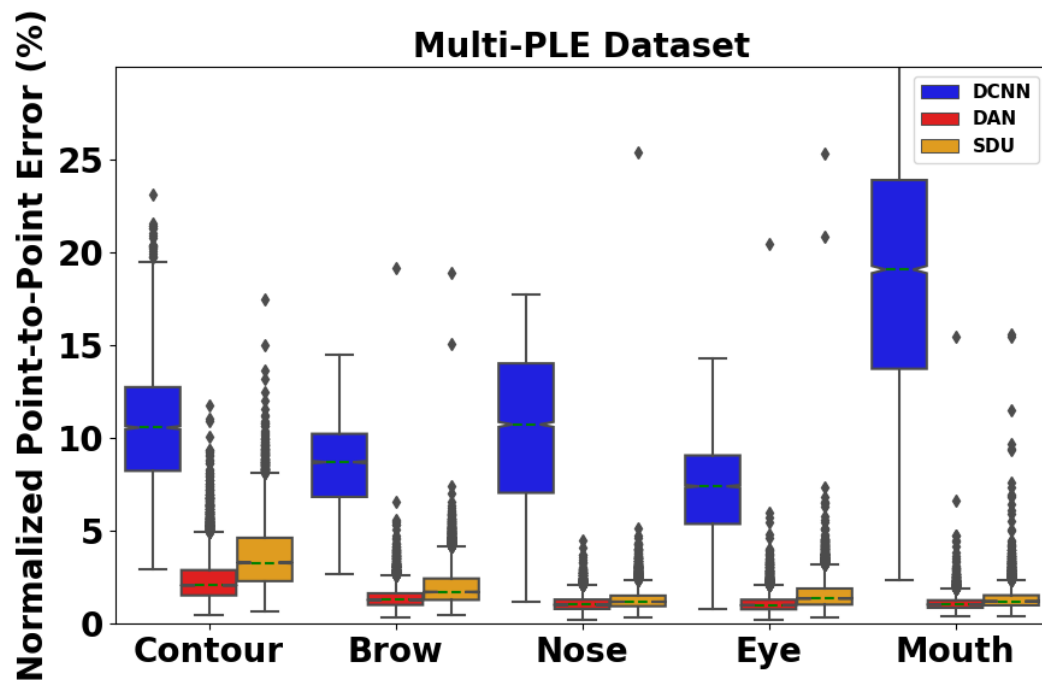


Figure 5.24 Comparison of landmarks error due to different facial attributes on the Multi-PLE dataset

In Figure 5.25, compared with DAN and SDU, SDU still has a bigger value from the 1st landmark to the 28th landmark and the 66th landmark to the 68th landmark because the self-occlusion images can cause highly biased of the predicted landmark located at the face contour, brows, and mouth.

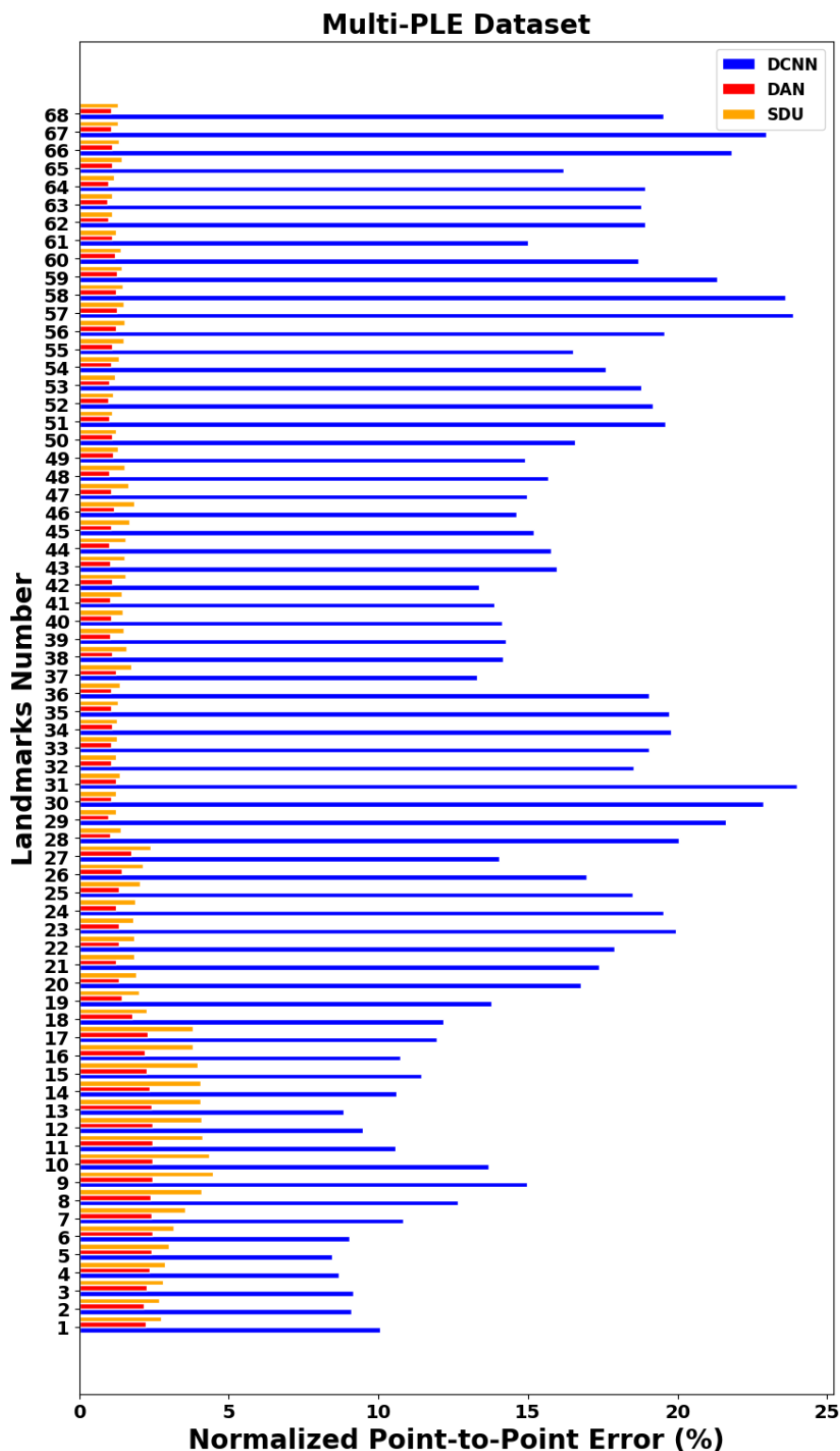


Figure 5.25 Comparison of each landmarks error on the Multi-PLE dataset

As the same test step as the previous dataset, the three algorithms tested the 15 images with different expressions on the Multi-PLE dataset first. SDU has the best

performance among the three algorithms, and DCNN shows the poorest performance in Table 5.12.

Algorithms	Mean	Std
DCNN	0.02673	0.00015
DAN	0.01983	0.00009
SDU	0.01555	0.00011

Table 5.12 The statistical results on the Multi-PLE dataset with different expressions

The NME values at different facial attributes of the 15 images with different expressions on the Multi-PLE dataset are shown in Figure 5.26. DCNN still has a significantly poor performance of the landmark at the mouth, while SDU shows the highest performance.

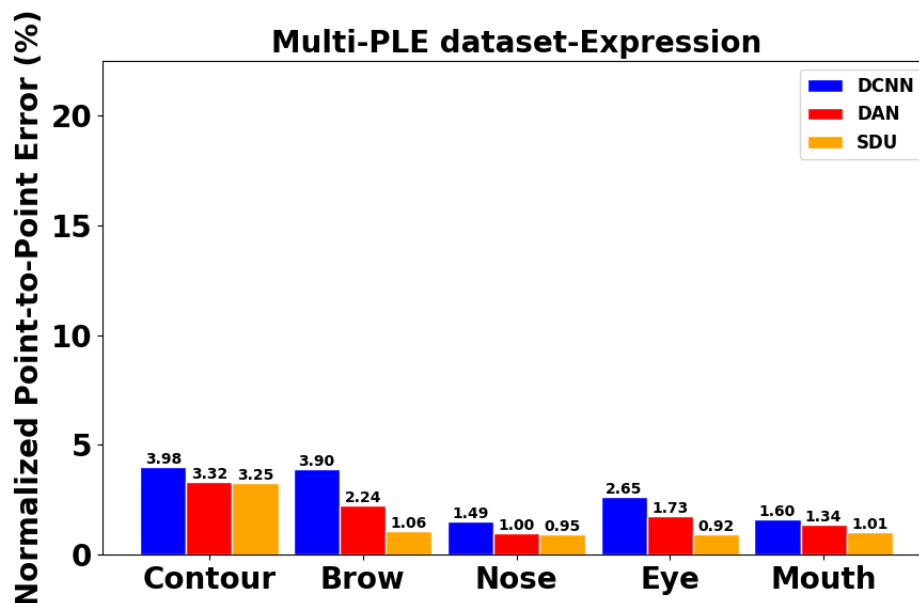


Figure 5.26 Comparison of landmarks error due to different facial attributes on the Multi-PLE dataset with different expressions

The landmark localisation results on the example images with different expressions on the Multi-PLE dataset are demonstrated in Figure 5.27. Examples of SDU have a remarkable ability to handle exaggerated expressions since the image does not have missing facial attributes or extreme illumination conditions.

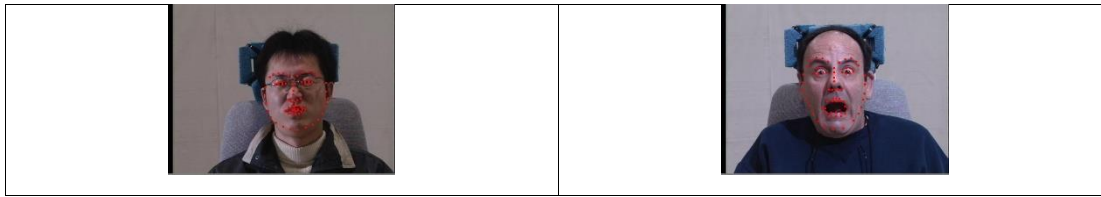


Figure 5.27 Examples of SDU landmark localisation

Secondly, the three algorithms test the 15 images with different illuminations on the Multi-PLE dataset. In Table 5.13, the statistic metric outlines that SDU is still the best-performing algorithm since the Multi-PLE dataset is a controlled dataset which does not have extreme situations to cause the missing facial attribute in the image.

Algorithms	Mean	Std
DCNN	0.02869	0.00017
DAN	0.01674	0.00004
SDU	0.01600	0.00010

Table 5.13 The statistical results on the Multi-PLE dataset with different illuminations

The NME values at different facial attributes of the 15 images with different illuminations on the Multi-PLE dataset are shown in Figure 5.28. SDU performs best at different facial attributes except for the landmarks at the contour, which is the same reason as the tested images in the 300W dataset and Menpo dataset.

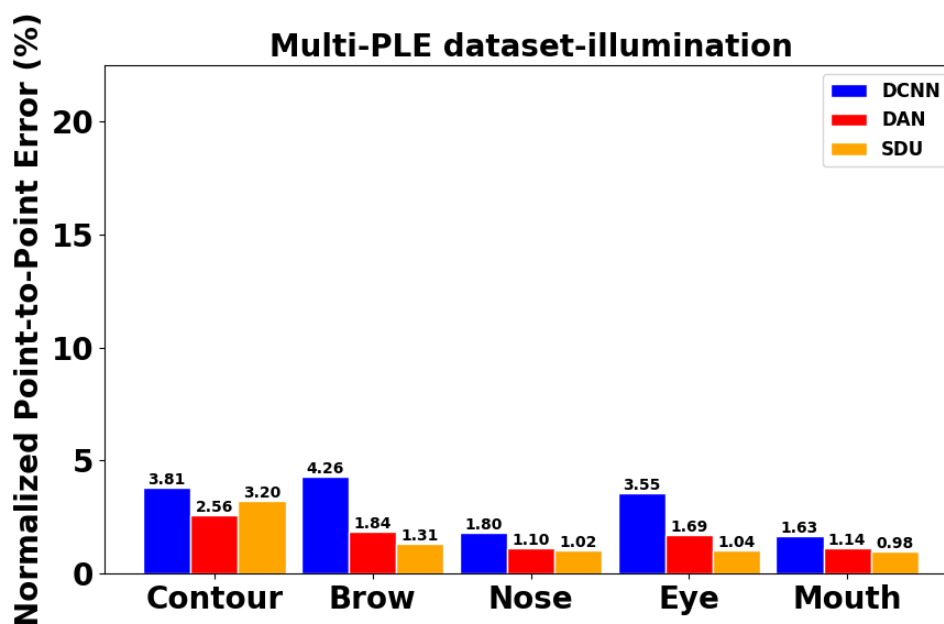


Figure 5.28 Comparison of landmarks error due to different facial attributes on the Multi-PLE dataset with different illuminations

Thirdly, several images are on the Multi-PLE dataset in which half of the face occludes. 15 example images are sampled to evaluate the performance of three algorithms. As seen in Table 5.14, DAN performs better than SDU, while DCNN still has the highest average of the NME.

Algorithms	Mean	Std
DCNN	0.07219	0.00011
DAN	0.02039	0.00007
SDU	0.02466	0.00031

Table 5.14 The statistical results on the Multi-PLE dataset with heavy occlusions

The NME values at different facial attributes of the 15 images with heavy occlusion on the Multi-PLE dataset are shown in Figure 5.29. Compared with DAN, SDU has a higher NME value at the contour, nose, and eye. Particularly, the NME of the contour is higher than over 0.015.

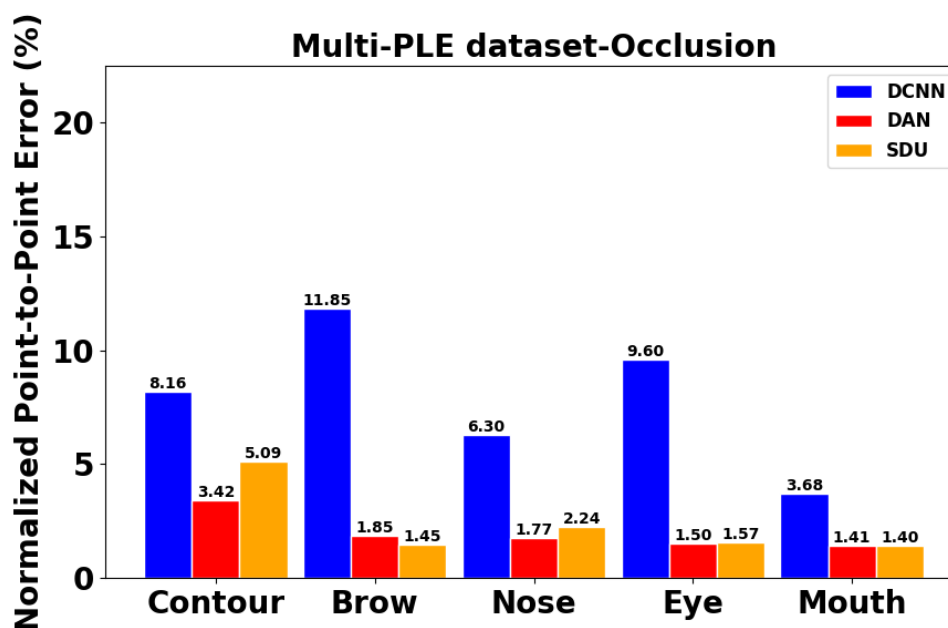


Figure 5.29 Comparison of landmarks error due to different facial attributes on the Multi-PLE dataset with heavy occlusions

The poor landmark localisation results of SDU are demonstrated in Figure 5.30. A few of the predicted landmarks are highly biased because of the missing facial attribute.



Figure 5.30 Examples of SDU poor landmark localisation

Finally, 15 frontal face images were sampled from the Multi-PLE dataset to evaluate the performance of three algorithms. In Table 5.15, SDU has the best performance again.

Algorithms	Mean	Std
DCNN	0.01983	0.00008
DAN	0.01273	0.00003
SDU	0.01190	0.00002

Table 5.15 The statistical results on the Multi-PLE dataset with frontal face

The NME values of the predicted landmarks at different facial attributes on the 15 images of the Multi-PLE dataset are illustrated in Figure 5.31. DCNN has the poorest performance at each face attribute. Furthermore, SDU has a higher mean error at the nose, and the NME of other facial attributes is lower than DAN.

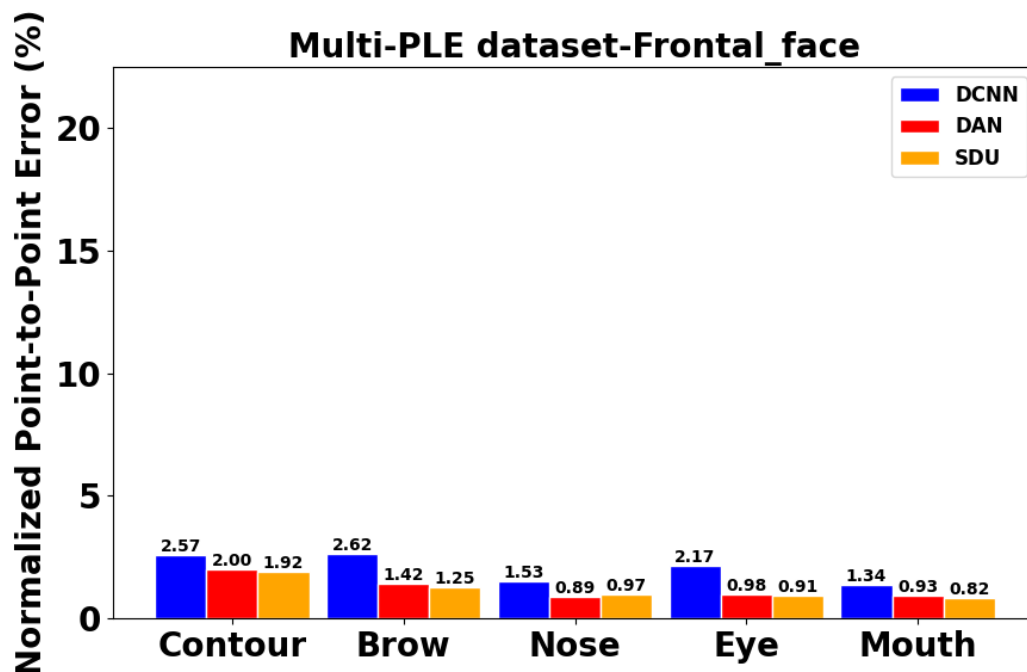


Figure 5.31 Comparison of landmarks error due to different facial attributes on the Multi-PLE dataset with frontal face

5.3.4 Summary of the three facial landmark algorithms

We examined the performance of DCNN, DAN, and SDU in three datasets, including 300W, Menpo, and Multi-PLE. Furthermore, each method was evaluated using three datasets with four circumstances: exaggerated expression, extreme illumination, heavy occlusion, and frontal-face conditions. The majority of the results demonstrate that SDU outperforms DCNN and DAN due to its network architecture as well as the input of landmark heatmap. However, the performance of SDU is still insufficient for a few images, particularly those with extreme illumination and heavy occlusion conditions, resulting in missing face attributes in the images. Compared with SDU, DAN has a better ability to handle this challenge, as it uses the mean shape to constraint the global face shape.

6. Conclusion and Future Work

This project has provided a comparative analysis of the existing automatic facial landmark algorithms in order to identify the most suitable one capable of handling various challenges in landmark-based facial analysis tasks. This chapter concludes the undertaking and outlines potential future research directions.

Automatic facial landmark detection algorithms can be divided into three categories based on facial shape and appearance: holistic methods, constrained local methods, and regression-based methods. The holistic methods and constrained local methods are both statistical methods that, in general, can demonstrate excellent generalisation capabilities with a small number of training data. However, the most recent advancements in regression-based methods can be broken down into three subcategories: direct, cascaded, and deep learning-based. We have chosen to employ deep learning-based methods because there are numerous publicly available datasets and powerful feature extractor.

For the purpose of comparing various deep learning-based algorithms, we utilised suitable datasets and a prominent and uniform landmark configuration. Three datasets, including the 300W dataset, the Menpo dataset, and the Multi-PLE dataset, were selected after a review of the varying number of public face datasets. In addition, the annotation was modified based on the MULTI-PLE 68 annotated landmarks configuration.

Three differing deep learning-based algorithms, including Deep Convolutional Neural Network (DCNN) Cascaded, Deep Alignment Network (DAN), and Stacked Dense U-Nets (SDU), were learned and implemented. In our implementation, these algorithms were developed in three steps: data pre-processing, model training, and model testing. In addition, face detectors such as Multi-Task Cascaded Convolutional Networks (MTCNN) and Sample and Computation Redistribution for efficient Face Detection. (SCRFD) were briefly introduced during the model testing phase. Following a comparison of the two face detectors, SCRFD was selected for use in our model testing.

Using ten-fold cross-validation, we trained and evaluated each algorithm using three datasets. Following that, we use evaluation metrics to compare the overall performance of each method. SDU is appropriate for use in medical auxiliary diagnosis because images used for medical diagnosis must capture the frontal face under controlled conditions, such as uniform illumination and absence of occlusion. In the absence of lacking facial attributes, SDU performs best among the three algorithms based on the total evaluation metrics. Since SDU is the most robust facial landmark algorithm, a novel facial asymmetry evaluation system (Wei et al., 2023; published at The 6th International Conference on Image and Graphics Processing) was developed using SDU.

References

- Akakin, H. Ç., & Sankur, B. (2011). Robust classification of face and head gestures in video. *Image and Vision Computing*, 29(7), 470–483.
- Asthana, A., Zafeiriou, S., Cheng, S., & Pantic, M. (2014). Incremental face alignment in the wild. *IEEE conference on computer vision and pattern recognition*, 1859–1866.
- Baltrušaitis, T., Robinson, P., & Morency, L. P. (2012). 3D constrained local model for rigid and non-rigid facial tracking. *IEEE conference on computer vision and pattern recognition*, 2610-2617.
- Belhumeur, P. N., Jacobs, D. W., Kriegman, D. J., & Kumar, N. (2011). Localizing parts of faces using a consensus of exemplars. *IEEE conference on computer vision and pattern recognition*, 545-552.
- Belhumeur, P., Jacobs, D., Kriegman, D., & Kumar, N. (2013). Localizing parts of faces using a consensus of exemplars. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(12), 2930–2940.
- Bulat, A., & Tzimiropoulos, G. (2017). Binarized Convolutional Landmark Localizers for Human Pose Estimation and Face Alignment with Limited Resources. *2017 IEEE International Conference on Computer Vision (ICCV)*, 3726-3734.
- Burgos-Artizzu, X. P., Perona, P., & Dollár, P. (2013). Robust Face Landmark Estimation under Occlusion. *IEEE International Conference on Computer Vision*, 1513–1520.
- Cao, X., Wei, Y., Wen, F., & Sun, J. (2014). Face alignment by explicit shape regression. *International Journal of Computer Vision*, 107, 177–190.
- Chen, D., Hua, G., Wen, F., & Sun, J. (2016). Supervised Transformer Network for Efficient Face Detection. *European Conference on Computer Vision*.

Cootes, T. F., Edwards, G. J., & Taylor, C. J. (2001). Active appearance models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6), 681–685.

Cootes, T. F., Ionita, M. C., Lindner, C., & Sauer, P. (2012). Robust and accurate shape model fitting using random forest regression voting. *European Conference on Computer Vision*, 278–291.

Cootes, T. F., Taylor, C. J., Cooper, D. H., & Graham, J. (1995). Active shape models their training and application. *Computer Vision and Image Understanding*, 61(1), 38–59.

Cristinacce, D., & Cootes, T. (2007). Boosted regression active shape models. *British Machine Vision Conference*, 880–889.

Cristinacce, D., & Cootes, T. F. (2006). Feature detection and tracking with constrained local models. *British Machine Vision Conference*, 929-938.

Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., & Wei, Y. (2017). Deformable Convolutional Networks. *2017 IEEE International Conference on Computer Vision (ICCV)*, 764-773.

Dantone, M., Gall, J., Fanelli, G., & Gool, L. V. (2012). Real-time facial feature detection using conditional regression forests. *IEEE conference on computer vision and pattern recognition*, 2578-2585.

Deng, J., Liu, Q., Yang, J., & Tao, D. (2016). M³CSR: Multi-view, multi-scale and multi-component cascade shape regression. *Image and Vision Computing*, 47, 19–26.

Deng, J., Roussos, A., Chrysos, G., Ververas, E., Kotsia, I., Shen, J., & Zafeiriou, S. (2019). The Menpo Benchmark for Multi-pose 2D and 3D Facial Landmark Localisation and Tracking. *International Journal of Computer Vision*, 127(6), 599–624.

- Ding, M., Kang, Y., Yuan, Z., Shan, X., & Cai, Z. (2021). Detection of facial landmarks by a convolutional neural network in patients with oral and maxillofacial disease. *International journal of oral and maxillofacial surgery*, 50(11), 1443–1449.
- Ding, X., Raziei, Z., Larson, E.C., Olinick, E.V., Krueger, P. and Hahsler, M. 2019. Swapped Face Detection using Deep Learning and Subjective Assessment. *Computer Vision and Pattern Recognition*, 6.
- Dornaika, F., & Davoine, F. (2004). Online appearance-based face and facial feature tracking. *Proceedings of the 17th International Conference on Pattern Recognition*, 3, 814-817.
- Pearson, K. (1901). On lines and planes of closest fit to systems of points in space, *Philosophical Magazine Series 1, 2*, 559-572.
- Fan, H., & Zhou, E. (2016). Approaching human level facial landmark localization by deep learning. *Image and Vision Computing*, 47, 27–35.
- Fink, M., Fergus, R., & Angelova, A. (2007). Caltech 10,000 web faces. Available from: https://www.vision.caltech.edu/datasets/caltech_10k_webfaces/.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press, Cambridge.
- Goodfellow, I., Bengio, Y., Courville, A. (2016). Convolutional Networks. Deep Learning, MIT Press, Cambridge, 330–372.
- Gower, J. C. (1975). Generalized procrustes analysis. *Psychometrika*, 40(1), 33–51.
- Gross, R., Matthews, I., & Baker, S. (2004). Appearance-based face recognition and light-fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(4), 449–465.

- Gross, R., Matthews, I., & Baker, S. (2005). Generic vs. person specific active appearance models. *Image Vision and Computing*, 23(12), 1080–1093.
- Gross, R., Matthews, I., Cohn, J., Kanade, T., & Baker, S. (2008). Multi-PIE. *IEEE International Conference on Automatic Face & Gesture Recognition*, 1–8.
- Gu, L., & Kanade, T. (2008). A generative shape regularization model for robust face alignment. *European Conference on Computer Vision*, 413–426.
- Guarin, D. L., Dusseldorp, J., Hadlock, T. A., & Jowett, N. (2018). A Machine Learning Approach for Automated Facial Measurements in Facial Palsy. *JAMA Facial Plastic Surgery*, 20(4), 335–337.
- Guo, J., Deng, J., Lattas, A. and Zafeiriou, S. (2021). Sample and computation redistribution for efficient face detection. *arXiv preprint*, arXiv:2105.04714.
- Guo, J., Deng, J., Xue, N., & Zafeiriou, S. (2018). Stacked Dense U-Nets with Dual Transformers for Robust Face Alignment. *ArXiv*, abs/1812.01936.
- Haghpannah, M., Saeedizade, E., Masouleh, M.T., & Kalhor, A. (2022). Real-Time Facial Expression Recognition using Facial Landmarks and Neural Networks. *2022 International Conference on Machine Vision and Image Processing (MVIP)*, 1-7.
- Hecht-Nielsen, R. (1989). Theory of the backpropagation neural network. *International 1989 Joint Conference on Neural Networks*, 1, 593-605.
- Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., & Adam, H. (2017). MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *ArXiv*, abs/1704.04861.
- Hu, C., Feris, R., & Turk, M. (2003). Real-time view-based face alignment using active wavelet networks. *IEEE international workshop on analysis and modeling of faces and gestures*, 215–221.

Ioffe, S., & Szegedy, C. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *ArXiv, abs/1502.03167*.

Jaderberg, M., Simonyan, K., Zisserman, A., & Kavukcuoglu, K. (2015). Spatial Transformer Networks. *NIPS*.

Jiao, F., Li, S., Shum, H., & Schuurmans, D. (2003). Face alignment using statistical models and wavelet features. *IEEE conference on computer vision and pattern recognition*, 1, 1-1.

Khan, F. (2018). Facial Expression Recognition using Facial Landmark Detection and Feature Extraction via Neural Networks. *ArXiv, abs/1812.04510*.

Kazemi, V., & Sullivan, J. (2014). One millisecond face alignment with an ensemble of regression trees. *IEEE conference on computer vision and pattern recognition (CVPR)*, 1867–1874.

Kingma, D.P., & Ba, J. (2014). Adam: A Method for Stochastic Optimization. *CoRR, abs/1412.6980*.

Kohavi, R. (1995). A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, 2, 1137–1143.

Köstinger, M., Wohlhart, P., Roth, P. M., & Bischof, H. (2011). Annotated Facial Landmarks in the Wild: A large-scale, real-world database for facial landmark localization. *IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, 2144–2151.

Kowalski, M., & Naruniec, J. (2016). Face Alignment Using K-Cluster Regression Forests With Weighted Splitting. *IEEE Signal Processing Letters*, 23, 1567-1571.

Kowalski, M., Naruniec, J., & Trzciński, T. (2017). Deep Alignment Network: A Convolutional Neural Network for Robust Face Alignment. *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2034-2043.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012), ImageNet classification with deep convolutional neural networks. *NIPS*, 1106–1114.

Le, V., Brandt, J., Lin, Z., Bourdev, L., & Huang, T. S. (2012). Interactive Facial Feature Localization. In A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, & C. Schmid (Eds.), *Computer Vision -- ECCV 2012*, 679–692.

Liu, Z., Luo, P., Wang, X., & Tang, X. (2014). Deep Learning Face Attributes in the Wild. *IEEE International Conference on Computer Vision (ICCV)*, 3730-3738.

Ma, W., & Lu, J. (2017), An Equivalence of Fully Connected Layer and Convolutional Layer. *ArXiv*, abs/1712.01252.

Martinez, A.M., & Benavente, R. (1998). The AR Face Database. *CVC Technical Report*, 24.

Martinez, B., Valstar, M. F., Binefa, X., & Pantic, M. (2013). Local evidence aggregation for regression-based facial point detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(5), 1149–1163.

Meng, Q., Zhao, S., Huang, Z., & Zhou, F. (2021). MagFace: A Universal Representation for Face Recognition and Quality Assessment. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 14220-14229.

Messer, K., Matas, J., Kittler, J., Luetin, J., & Maître, G. (1999). XM2VTSDB: The Extended M2VTS Database. *Second International Conference on Audio and Video-based Biometric Person Authentication*, 964, 965-966.

Milborrow, S., & Nicolls, F. (2008). Locating Facial Features with an Extended Active Shape Model. *European Conference on Computer Vision*, 504-513.

- Newell, A., Yang, K., & Deng, J. (2016). Stacked Hourglass Networks for Human Pose Estimation. *European Conference on Computer Vision*.
- Pantic, M., & Rothkrantz, L. J. M. (2000). Automatic analysis of facial expressions: the state of the art. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12), 1424–1445.
- Patrick Sauer, T. C., & Taylor, C. (2011). Accurate regression procedures for active appearance models. *British Machine Vision Conference*, 30.1-30.11.
- Perov, I.E., Gao, D., Chervoniy, N., Liu, K., Marangonda, S., Um'e, C., Dpfks, M., Facenheim, C.S., Luis, R., Jiang, J., Zhang, S., Wu, P., Zhou, B., & Zhang, W. (2020). DeepFaceLab: Integrated, flexible, and extensible face-swapping framework. *arXiv preprint*, arXiv:2005.05535.
- Phillips, P., Flynn, P., Scruggs, W., Bowyer, K., Chang, J., Hoffman, K., Marques, J., Min, J., & Worek, W. (2005). Overview of the Face Recognition Grand Challenge. *IEEE Conference on Computer Vision and Pattern Recognition*, 1, 947-954.
- Ranjan, R., Patel, V.M., & Chellappa, R. (2019). HyperFace: A Deep Multi-Task Learning Framework for Face Detection, Landmark Localization, Pose Estimation, and Gender Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41, 121-135.
- Ren, S., Cao, X., Wei, Y., & Sun, J. (2014). Face alignment at 3000 FPS via regressing local binary features. *IEEE conference on computer vision and pattern recognition (CVPR)*, 1685–1692.
- Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. *International Conference on Medical image computing and computer-assisted intervention*, 234-241.

Sagonas, C., & Zafeiriou, S. (2017), Facial point annotations. Available from: <https://ibug.doc.ic.ac.uk/resources/facial-point-annotations/>

Sagonas, C., Antonakos, E., Tzimiropoulos, G., Zafeiriou, S., & Pantic, M. (2016). 300 Faces In-The-Wild Challenge: database and results. *Image and Vision Computing*, 47, 3–18.

Sammut, C., & Webb, G.I. (2010). Mean squared error. *Encyclopedia of Machine Learning*, 653– 653.

Saragih, J. M., Lucey, S., & Cohn, J. F. (2011). Deformable model fitting by regularized landmark mean-shift. *International Journal of Computer Vision*, 91(2), 200–215.

Sarker I. H. (2021). Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions. *SN computer science*, 2(6), 420.

Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*, 6(1), 60.

Simonyan, K., & Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. *CoRR*, abs/1409.1556.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *J. Mach. Learn. Res.*, 15(1), 1929–1958.

Sun, Y., Wang, X., & Tang, X. (2013). Deep Convolutional Network Cascade for Facial Point Detection. *IEEE Conference on Computer Vision and Pattern Recognition*, 3476–3483.

Taufique, A. M. N., Savakis, A., & Leckenby, J. (2019). Automatic Quantification of Facial Asymmetry Using Facial Landmarks. *IEEE Western New York Image and Signal Processing Workshop (WNYISPW)*, 1-5.

- Tian, Y.-I., Kanade, T., & Cohn, J. F. (2001). Recognizing action units for facial expression analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(2), 97–115.
- Tong, Y., & Ji, Q. (2006). Multiview Facial Feature Tracking with a Multi-modal Probabilistic Model. 18th International Conference on Pattern Recognition (ICPR'06), 1, 307–310.
- Trigeorgis, G., Snape, P., Nicolaou, M. A., Antonakos, E., & Zafeiriou, S. (2016). Mnemonic descent method: A recurrent process applied for end-to-end face alignment. *IEEE conference on computer vision and pattern recognition (CVPR)*, 4177–4187.
- Valstar, M., Martinez, B., Binefa, V., & Pantic, M. (2010). Facial point detection using boosted regression and graph models. *IEEE conference on computer vision and pattern recognition*, 13-18.
- Wei, Q., Ziyu, Y., Bogdan, J.M. (2023). Development of Landmark-based Facial Asymmetry Evaluation. ICIGP 2023: 2023 the 6th *International Conference on Image and Graphics Processing (ICIGP)*.
- Wu, W., Qian, C., Yang, S., Wang, Q., Cai, Y., & Zhou, Q. (2018). Look at Boundary: A Boundary-Aware Face Alignment Algorithm. *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2129-2138.
- Wu, Y., & Ji, Q. (2018). Facial Landmark Detection: A Literature Survey. *International Journal of Computer Vision*, 127, 115-142.
- Wu, Y., Wang, Z., & Ji, Q. (2013). Facial feature tracking under varying facial expressions and face poses based on restricted Boltzmann machines. *IEEE conference on computer vision and pattern recognition*, 3452–3459.

Xiao, S., Feng, J., Xing, J., Lai, H., Yan, S., & Kassim, A.A. (2016). Robust Facial Landmark Detection via Recurrent Attentive-Refinement Networks. *European Conference on Computer Vision*, 57-72.

Yang, J., Liu, Q., & Zhang, K. (2017). Stacked Hourglass Network for Robust Facial Landmark Localisation. *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2025–2033.

Yu, F., Wang, D., Shelhamer, E., & Darrell, T. (2018). Deep Layer Aggregation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2403-2412.

Zafeiriou, S., Chrysos, G. G., Roussos, A., Ververas, E., Deng, J., & Trigeorgis, G. (2017). The 3D Menpo Facial Landmark Tracking Challenge. *IEEE International Conference on Computer Vision Workshops (ICCVW)*, 2503–2511.

Zeiler, M. D., Krishnan, D., Taylor, G. W., & Fergus, R. (2010). Deconvolutional networks. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2528–2535.

Zhang, C., & Zhang, Z. (2014). Improving multiview face detection with multi-task deep convolutional neural networks. *IEEE Winter Conference on Applications of Computer Vision*, 1036–1041.

Zhang, K., Zhang, Z., Li, Z., & Qiao, Y. (2016). Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks. *IEEE Signal Processing Letters*, 23(10), 1499–1503.

Zhang, K., Zhang, Z., Li, Z., & Qiao, Y. (2016). Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks. *IEEE Signal Processing Letters*, 23(10), 1499–1503.

Zhang, Z., Luo, P., Loy, C. C., & Tang, X. (2014). Facial Landmark Detection by Deep Multi-task Learning. In D. Fleet, T. Pajdla, B. Schiele, & T. Tuytelaars (Eds.), *Computer Vision -- ECCV 2014*, 94–108.

Zhou, E., Fan, H., Cao, Z., Jiang, Y., & Yin, Q. (2013). Extensive facial landmark localization with coarse-to-fine convolutional network cascade. *IEEE international conference on computer vision workshops*, 386–391.

Zhu, S., Li, C., Loy, C. C., & Tang, X. (2015). Face alignment by coarse-to-fine shape searching. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 4998–5006.

Zhu, X., & Ramanan, D. (2012). Face detection, pose estimation, and landmark localization in the wild. *IEEE Conference on Computer Vision and Pattern Recognition*, 2879–2886.

Zhu, X., & Ramanan, D. (2012). Face detection, pose estimation, and landmark localization in the wild. *IEEE conference on computer vision and pattern recognition*, 2879–2886.

Zhu, X., Lei, Z., Liu, X., Shi, H., Li, S. (2016). Face alignment across large poses: A 3D solution. *IEEE conference on computer vision and pattern recognition*, 146-155.