Facial Landmark Localization

5

Xiangyu Zhu, Zhenhua Feng, and Hailin Shi

5.1 Introduction

Facial landmark localization aims to detect a sparse set of facial fiducial points on a human face, some of which include “eye corner”, “nose tip”, and “chin center”. In the pipeline of face analysis, landmark detectors take the input of a face image and the bounding box provided by face detection, and output a set of coordinates of the predefined landmarks, which is illustrated in Fig. 5.1. It provides a fine-grained description of the face topology, such as facial features locations and face region contours, which is essential for many face analysis tasks, e.g., recognition [32], animation [33], attributes classification [34], and face editing [35]. These applications usually run on lightweight devices in uncontrolled environments, requiring landmark detectors to be accurate, robust, and computationally efficient, all at the same time.

Over the past few decades, there have been significant developments in facial landmark detection. The early works consider landmark localization as the process of moving and deforming a face model to an image, and they construct a statistical facial model to model the shape and albedo variations of human faces. The most prominent algorithms include Active Shape Model (ASM) [42], Active Appearance Model (AAM) [43], and Constrained

X. Zhu (✉)
Institute of Automation, Chinese Academy of Sciences, Beijing, China
e-mail: xiangyu.zhu@nlpr.ia.ac.cn

Z. Feng
School of Computer Science and Electronic Engineering, University of Surrey, Guildford, UK
e-mail: z.feng@surrey.ac.uk

H. Shi
Nio Inc., Beijing, China
e-mail: hailinshi.work@outlook.com

S. Z. Li et al. (eds.), *Handbook of Face Recognition*,
https://doi.org/10.1007/978-3-031-43567-6_5

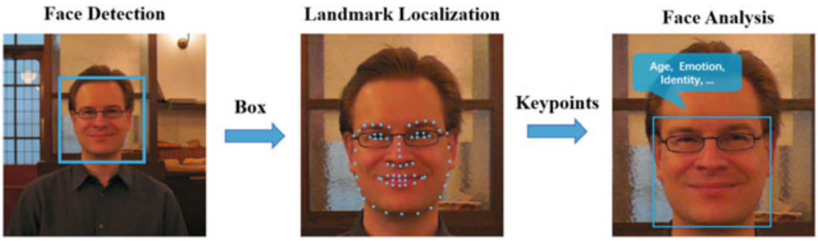


Fig. 5.1 Facial landmark localization

Local Model (CLM) [44], by which the faces in controlled environments (normal lighting and frontal poses) can be well handled. However, these methods deteriorate greatly when facing enormous challenges in the wild, such as large poses, extreme illuminations, low resolution, and partial occlusions. The next wave of methods is based on cascaded regression [45, 88, 89], which cascades a list of weak regressors to reduce the alignment error progressively. For example, the Supervised Descent Method (SDM) [88] updates the landmark locations by several iterations of regressions. In each iteration, a regressor takes the input of the appearance features (e.g., SIFT) around landmarks, and estimates a landmark update to approach the ground-truth locations. The Ensemble of Regression Trees (ERT) [45] learns an ensemble of regression trees to regress the landmarks from a sparse subset of intensity values, so as to handle partial or uncertain labels. One of the most popular landmark detectors Dlib [46] implements ERT as its landmark detector due to its high speed of 1 millisecond per face.

Following the great success of deep learning in computer vision [47], researchers started to predict facial landmarks by deep convolutional neural networks. In general, deep learning-based landmark detectors can be divided into coordinate-based and heatmap-based, illustrated in Fig. 5.2, depending on the detection head of network architecture. Coordinate-based methods output a vector consisting of 2D coordinates of landmarks. On the contrary, heatmap-based methods output one heatmap for each landmark, where the intensity value of the heatmap indicates the probability that this landmark locates in this position. It is commonly agreed [38, 39] that heatmap-based methods detect more accurate landmarks, but are computationally expensive and sensitive to outliers. In contrast, coordinate-based methods are fast and robust, but have sub-optimal accuracy.

In recent years, 3D landmark localization has attracted increasing attention due to its additional geometry information and superiority in handling large poses [40]. However, localizing 3D landmarks is more challenging than 2D landmarks because recovering depth from a monocular image is an ill-posed problem. This requires the model to build a strong 3D face prior from large-scale 3D data in order to accurately detect and locate the facial landmarks in 3D space. Unfortunately, acquiring 3D faces is expensive, and labeling 3D landmarks is also tedious. A feasible solution is to fit a 3D Morphable Model (3DMM) [41] by a neural network [40] and sample the 3D landmarks from the fitted 3D model. Another

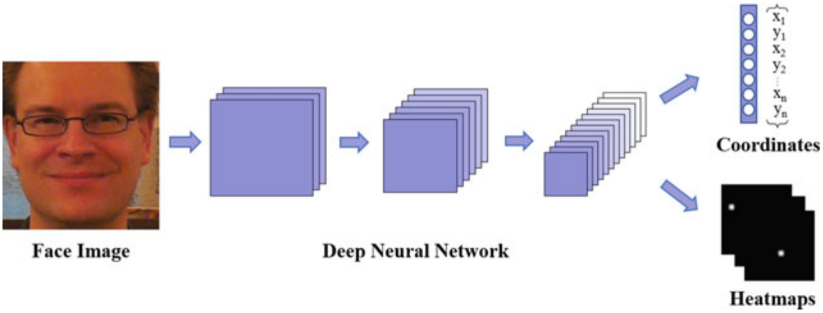


Fig. 5.2 Coordinate-based methods and heatmap-based methods

one is utilizing a fully convolutional network to regress the 3D heatmaps, on which the coordinates of the largest probabilities are sampled as 3D landmarks [51, 52].

5.2 Coordinate Regression

As deep learning has become the mainstream method for facial landmark localization, this section focuses on recent advances in deep learning-based coordinate regression approaches. Given an input face image, coordinate regression-based methods predict the 2D coordinates of a set of predefined facial landmarks directly from the deep features extracted by a backbone network, as shown in Fig. 5.3.

5.2.1 Coordinate Regression Framework

The task of coordinate regression-based facial landmark localization is to find a nonlinear mapping function (usually a deep CNN model):

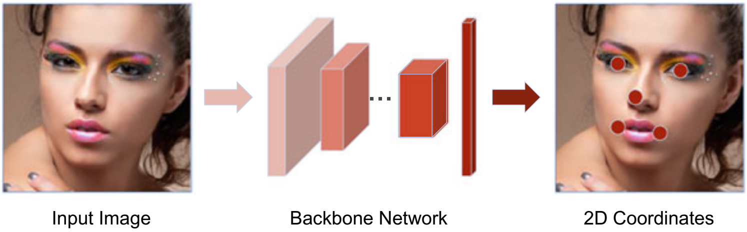


Fig. 5.3 Coordinate regression-based facial landmark localization. The input is an RGB face image, and the output is a vector consisting of the 2D coordinates of all the facial landmarks

$$\Phi : \mathcal{I} \rightarrow \mathbf{s}, \tag{5.1}$$

that outputs the 2D coordinates vector $\mathbf{s} \in \mathbb{R}^{2L}$ of L landmarks for a given facial image $\mathcal{I} \in \mathbb{R}^{H \times W \times 3}$. In general, the input image is cropped by using a bounding box obtained by a face detector in a full-stack facial image/video analysis pipeline. The 2D coordinate vector $\mathbf{s} = [x_1, ..., x_L, y_1, ..., y_L]^T$ consists of the coordinates of L predefined landmarks, where (x_l, y_l) are the X- and Y-coordinates of the lth landmark.

To obtain the above mapping function, a deep neural network can be used, which is formulated as a compositional function:

$$\Phi = (\phi_1 \circ ... \circ \phi_M)(\mathcal{I}), \tag{5.2}$$

with M sub-functions, and each sub-function (ϕ) represents a specific network layer, e.g., convolutional layer and nonlinear activation layer. Most existing deep learning-based facial landmark localization approaches use CNN as the backbone with a regression output layer [24–26].

Given a set of labeled training samples $\Omega = \{\mathcal{I}_i, \mathbf{s}_i\}_{i=1}^{N}$, the network training aims to find the best set of the parameters Φ so that to minimize:

$$\sum_{i=1}^{N} loss(\Phi(\mathcal{I}_i), \mathbf{s}_i), \tag{5.3}$$

where $loss()$ is a predefined loss function that measures the difference between the predicted and ground-truth coordinates over all the training samples. To optimize the above objective function, a variety of optimization methods, such as Stochastic Gradient Descent (SGD) and AdamW, can be used for network training.

5.2.2 Network Architectures

As shown in Fig. 5.3, the input for a coordinate regression-based facial landmark localization model is usually an image enclosing the whole face region. Then a backbone CNN network can be used for feature extraction and fully connected layers are used for regressing the landmark coordinates. With the development of deep learning, different backbone networks have been explored and evaluated for accurate and robust landmark localization. For example, Feng et al. [38] evaluated different backbone networks, including VGG, ResNet, MobileNet, etc., for efficient and high-performance facial landmark localization. As face landmarking is a key element in a full-stack facial image/video analysis system, the design of a lightweight network is crucial for real-time applications. For instance, Guo et al. [18] developed a light framework that is only 2.1 MB and runs at 140 fps on a mobile device. Gao et al. [19] proposed EfficientFAN that applies deep knowledge transfer via a teacher-student network for efficient and effective network training. Feng et al. [38] compared different designs of network architectures and evaluated their inference speed on different devices, including GPU, CPU, and portable devices.

Instead of the whole face image, shape- or landmark-related local patches have also been widely used as the input of neural networks [24, 83]. To use local patches, one can apply CNN-based feature extraction to the local patches centered at each landmark and for fine-grained landmark prediction or update [83]. The advantage of using the whole face region, in which the only input of the network is a cropped face image, is that it does not require the initialization of facial landmarks. In contrast, to use local patches, a system usually requires initial estimates of facial landmarks for a given image. This can be achieved by either using the mean-shape landmarks [83] or the output of another network that predicts coarse landmarks [24, 27, 61].

The accuracy of landmark localization can be degraded by in-plane face rotations and inaccurate bounding boxes output by a face detector. To address these issues, a widely used strategy is to cascade multiple networks to form a coarse-to-fine structure. For example, Huang et al. [28] proposed to use a global network to obtain coarse facial landmarks for transforming a face to the canonical view and then applied multiple networks trained on different facial parts for landmark refinement. Similarly, both Yang et al. [29] and Deng et al. [30] proposed to train a network that predicts a small number of facial landmarks (5 or 19) to transform the face to a canonical view. It should be noted that the first network can be trained on a large-scale dataset so it performs well for unconstrained faces with in-plane head rotation, scale, and translation. With the first stage, the subsequent networks that predict all the landmarks can be trained with the input of normalized faces.

Feng et al. [38] also proposed a two-stage network for facial landmark localization, as shown in Fig. 5.4. The coarse network is trained on a dataset with very heavy data augmentation by randomly rotating an original training image between [−180°, 180°] and perturbing the bounding box with 20% of the original bounding box size. Such a trained network is able to perform well for faces with large in-plane head rotations and low-quality

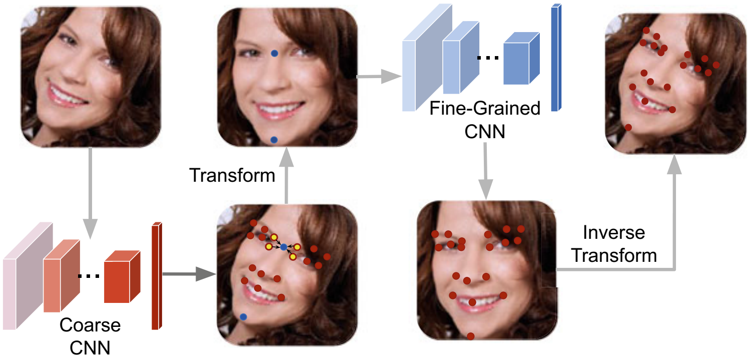


Fig. 5.4 A two-stage coarse-to-fine facial landmark localization framework

bounding boxes. For training the second network, each training sample is fed to the first network to obtain its coarse facial landmarks for geometric normalization. To be specific, two anchor points (blue points in Fig. 5.4) are computed to perform the rigid transformation, where one anchor is the mean of the four inner eye and eyebrow corners and the other one is the chin landmark. Afterward, the normalized training data is lightly augmented by randomly rotating the image between $[-10°, 10°]$ and perturbing the bounding box with 10% of the bounding box size. The aim is to address the issues caused by inaccurate landmark localization of the first network. Finally, a second network is trained on the normalized-and-lightly-augmented dataset for further performance boosting in localization accuracy. The joint use of these two networks in a coarse-to-fine fashion is instrumental in enhancing the generalization capacity and accuracy.

5.2.3 Loss Functions

Another important element for high-performance coordinates regression is the design of a proper loss function. Most existing regression-based facial landmark localization approaches with deep neural networks are based on the L2 loss function. Given a training image $\mathcal{I}$ and a network Φ, we can predict the facial landmarks as a vector $\mathbf{s}' = \Phi(\mathcal{I})$. The loss function is defined as:

$$loss(\mathbf{s}, \mathbf{s}') = \frac{1}{2L} \sum_{i=1}^{2L} f(s_i - s'_i), \tag{5.4}$$

where $\mathbf{s}$ is the ground-truth facial landmark coordinates and s_i is its ith element. For $f(x)$ in the above equation, the L2 loss is defined as:

$$f_{L2}(x) = \frac{1}{2}x^2. \tag{5.5}$$

However, it is well known that the L2 loss function is sensitive to outliers, which has been noted in connection with many existing studies, such as the bounding box regression problem in face detection [31]. To address this issue, L1 and smooth L1 loss functions are widely used for robust regression. The L1 loss is defined as:

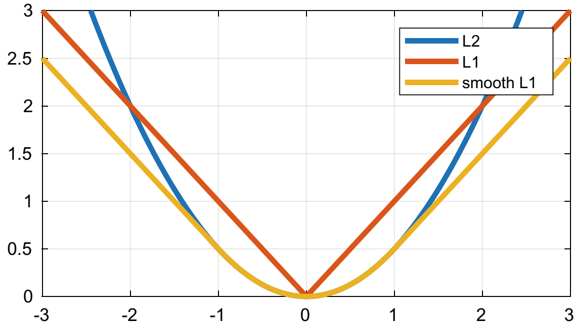
$$f_{L1}(x) = |x|. \tag{5.6}$$

The smooth L1 loss is defined piecewise as:

$$f_{smL1}(x) = \begin{cases} \frac{1}{2}x^2 & \text{if } |x| < 1 \\ |x| - \frac{1}{2} & \text{otherwise} \end{cases}, \tag{5.7}$$

which is quadratic for small values and linear for large values [31]. More specifically, smooth L1 uses $f_{L2}(x)$ for $x \in (-1, 1)$ and shifts to $f_{L1}(x)$ elsewhere. Figure 5.5 depicts the plots of these three loss functions.

Fig. 5.5 Plots of the L2, L1 and smooth L1 loss functions



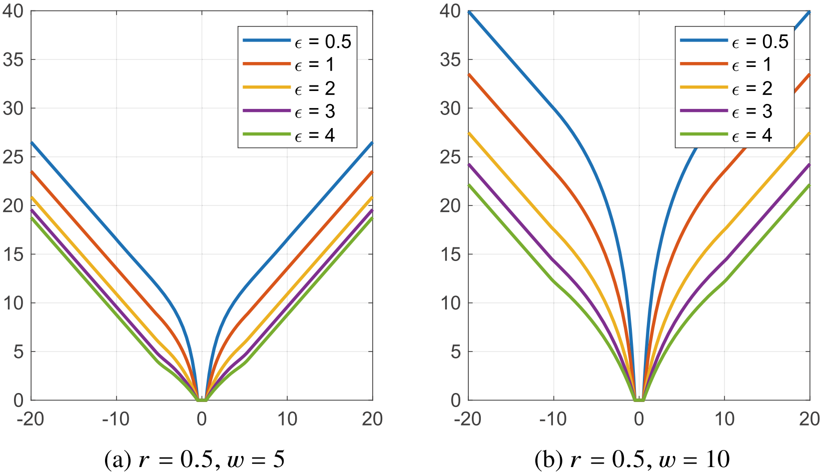
However, outliers are not the only subset of points which deserve special consideration. Feng et al. [38] argued that the behavior of the loss function at points exhibiting small-medium errors is just as crucial to finding a good solution to the landmark localization task. Based on a more detailed analysis, they proposed a new loss function, namely Rectified Wing (RWing) loss, for coordinate regression-based landmark localization. Similar to the original Wing loss function, RWing is also defined piecewise:

$$RWing(x) = \begin{cases} 0 & \text{if } |x| < r \\ w \ln(1 + (|x| - r)/\epsilon) & \text{if } r \leq |x| < w \\ |x| - C & \text{otherwise} \end{cases}, \tag{5.8}$$

where the non-negative parameter r sets the range of rectified region to $(-r, r)$ for very small values. The aim is to remove the impact of noise labels on network convergence. For a training sample with small-medium range errors in $[r, w)$, RWing uses a modified logarithm function, where ϵ limits the curvature of the nonlinear region and $C = w - w \ln(1 + (w - r)/\epsilon)$ is a constant that smoothly links the linear and nonlinear parts. Note that one should not set ϵ to a very small value because this would make the training of a network very unstable and cause the exploding gradient problem for small errors. In fact, the nonlinear part of the RWing loss function just simply takes a part of the curve of $\ln(x)$ and scales it along both the X-axis and Y-axis. Also, RWing applies translation along the Y-axis to allow $RWing(\pm r) = 0$ and to impose continuity on the loss function at $\pm w$. In Fig. 5.6, some examples of the RWing loss with different hyper parameters are demonstrated.

5.3 Heatmap Regression

Another main category of the state-of-the-art facial landmark localization methods is heatmap regression. Different from coordinate regression, heatmap regression outputs a heatmap for each facial landmark. In the heatmap, the intensity value of a pixel in a heatmap indicates the probability that its location is the predicted position of the corresponding



(a) $r = 0.5, w = 5$ (b) $r = 0.5, w = 10$

Fig. 5.6 The Rectified Wing loss function plotted with different hyper parameters, where r and w limit the range of the nonlinear part and ϵ controls the curvature. By design, the impact of the samples with small- and medium-range errors is amplified, and the impact of the samples with very small errors is ignored

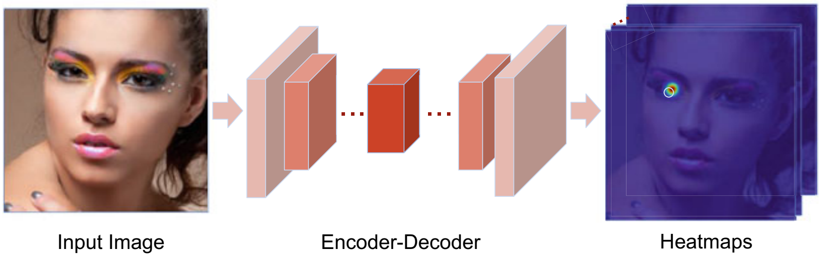


Fig. 5.7 Heatmap regression-based facial landmark localization. The input is a face image and the output are L 2D heatmaps, each for one predefined facial landmark. The backbone network usually has an encoder-decoder architecture

landmark. The task of heatmap regression-based facial landmark localization is to find a nonlinear mapping function:

$$\Phi : \mathcal{I} \rightarrow \mathcal{H}, \tag{5.9}$$

that outputs L 2D heatmaps $\mathcal{H} \in \mathbb{R}^{H \times W \times L}$ for a given image $\mathcal{I} \in \mathbb{R}^{H \times W \times 3}$. As shown in Fig. 5.7, heatmap regression usually uses an encoder-decoder architecture for heatmap generation. For network training, typical loss functions used for heatmap generation include MSE and L1.

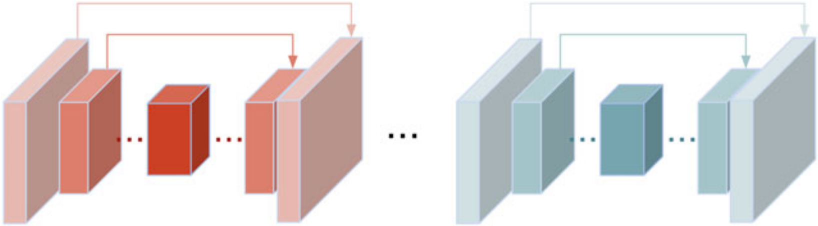


Fig. 5.8 A typical architecture of a stacked hourglass network

5.3.1 Network Architectures

As aforementioned, heatmap regression usually applies an encoder-decoder architecture for high-performance facial landmark localization. The most popular backbone network used for heatmap regression might be the stacked hourglass network [29, 30, 55, 68]. The key to the success of a stacked hourglass network is the use of multiple hourglass networks with residual connections, as shown in Fig. 5.8. On the one hand, the use of residual connections in each hourglass network maintains multi-scale facial features for fine-grained heatmap generation. On the other hand, stacking multiple hourglass networks improves the overall network capacity, so as to improve the quality of a generated heatmap. Besides the stacked hourglass network, another two popular network architectures used for heatmap regression are HRNet [75] and U-Net [77]. Similar to hourglass, both HRNet and U-Net try to find an effective way of using multi-scale features rather than the single use of a deep high-level semantic feature map for heatmap generation.

To reduce false alarms of a generated 2D heatmap, Wu et al. [22] proposed a distance-aware softmax function that facilitates the training of a dual-path network. Lan et al. [79] further investigated the issue of quantization error in heatmap regression, and proposed a heatmap-in-heatmap method for improving the prediction accuracy of facial landmarks. Instead of using a Gaussian map for each facial landmark, Wu et al. [68] proposed to create a boundary heatmap mask for feature map fusion and demonstrated its merits in robust facial landmark localization.

5.3.2 Loss Function

Similar to coordinate regression, the design of a proper loss function is crucial for heatmap regression-based facial landmark localization. Most of the existing heatmap regression methods use MSE or L1 loss for heatmap generation via an encoder-decoder network. However, a model trained with MSE or L1 loss tends to predict blurry and dilated heatmaps with low intensity on foreground pixels compared to the ground-truth ones. To address this issue, Wang et al. [76] proposed an adaptive Wing loss function for heatmap regression. In contrast to the original Wing loss [20], the adaptive Wing loss is a tailored version for heatmap generation. The adaptive Wing loss is able to adapt its shape to different types of ground-truth heatmap pixels. This adaptability penalizes loss more on foreground pixels while less on background pixels, hence improving the quality of a generated heatmap and the performance of the final landmark localization task in terms of accuracy.

To be specific, the adaptive Wing loss function is defined as:

$$AWing(y, \hat{y}) = \begin{cases} w \ln(1 + |\frac{y-\hat{y}}{\epsilon}|^{\alpha-y}) & \text{if } |y-\hat{y}| < \theta \\ A|y-\hat{y}| - C & \text{otherwise} \end{cases}, \tag{5.10}$$

where y and $\hat{y}$ are the intensities of the pixels on the ground truth and predicted heatmaps, respectively. w, θ, ϵ and α are positive values, $A = w(1/(1 + (\theta/\epsilon)^{(\alpha-y)}))(\alpha - y)((\theta/\epsilon)^{(\alpha-y-1)})(1/\epsilon)$ and $C = (\theta A - w \ln(1 + (\theta/\epsilon)^{\alpha-y}))$ are designed to link different parts of the loss function continuously and smoothly at $|y - \hat{y}| = \theta$. Unlike the Wing loss, which uses w as the threshold, the adaptive Wing loss introduces a new variable θ as the threshold to switch between linear and nonlinear parts. For heatmap regression, a deep network usually regresses a value between 0 and 1, so the adaptive Wing loss sets the threshold in this range. When $|y - \hat{y}| < \theta$, adaptive Wing considers the error to be small and thus needs stronger influence. More importantly, this new loss function adopts an exponential term $\alpha - y$, which is used to adapt the shape of the loss function to y and makes the loss function smooth at the origin.

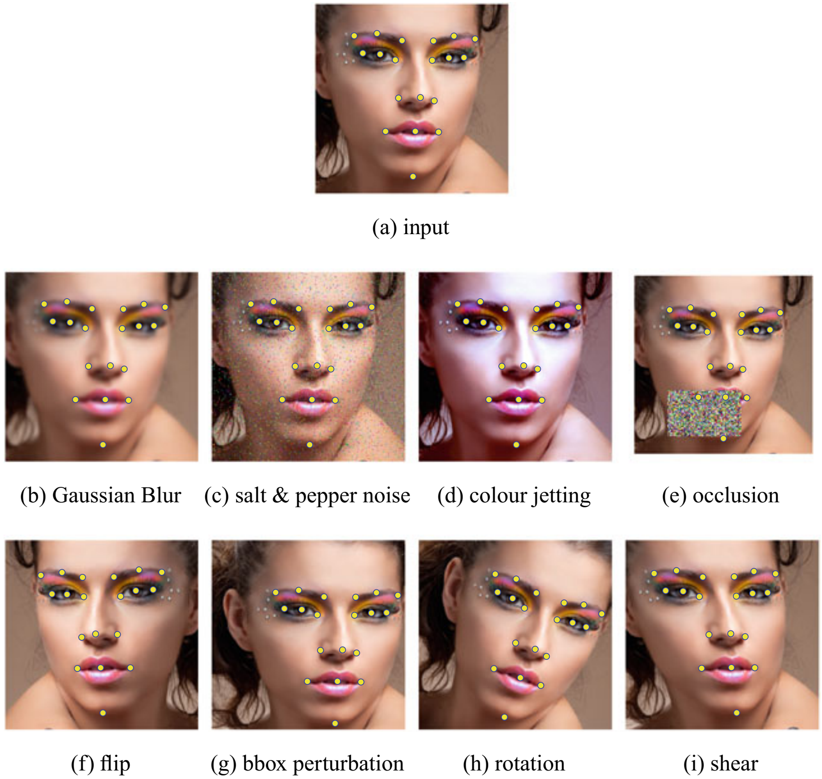
It should be noted that adaptive Wing loss is able to adapt its curvature to the ground-truth pixel values. This adaptive property reduces small errors on foreground pixels for accurate landmark localization, while tolerating small errors on background pixels for better convergence of a network.

5.4 Training Strategies

5.4.1 Data Augmentation

For a deep learning-based facial landmark localization method, a key to the success of network training is big labeled training data. However, it is a difficult and tedious task to manually label a large-scale dataset with facial landmarks. To mitigate this issue, effective data augmentation has become an essential alternative. Existing data augmentation approaches in facial landmark localization usually inject geometric and textural variations into training images. These augmentation approaches are efficient to implement and thus can be easily performed online for network training.

To investigate the impact of these data augmentation methods on the performance of a facial landmark localization model, Feng et al. [26] introduced different data augmentation approaches and performed a systematic analysis of their effectiveness in the context of



(a) input

(b) Gaussian Blur (c) salt & pepper noise (d) colour jetting (e) occlusion

(f) flip (g) bbox perturbation (h) rotation (i) shear

Fig. 5.9 Different geometric and textural data augmentation approaches for facial landmark localization. “bbox” refers to “bounding box”

deep-learning-based facial landmark localization. Feng et al. divided the existing data augmentation techniques into two categories: textural and geometric augmentation, as shown in Fig. 5.9. Textural data augmentation approaches include Gaussian blur, salt and pepper noise, color jetting, and random occlusion. Geometric data augmentation consists of horizontal image flip, bounding box perturbation, rotation and shear transformation. According to the experimental results, all data augmentation approaches improve the accuracy of the baseline model. However, the key finding is that the geometric data augmentation methods are more effective than the textural data augmentation methods for performance boosting. Furthermore, the joint use of all data augmentation approaches performs better than only using a single augmentation method.

In addition, Feng et al. [26] argued that, by applying random textural and geometric variations to the original labeled training images, some augmented samples may be harder and more effective for deep network training. However, some augmented samples are less effective. To select the most effective augmented training samples, they proposed a Hard

Augmented Example Mining (HAEM) method for effective sample mining. In essence, HAEM selects N hard samples from each mini-batch those which exhibit the largest losses but excludes the one of dominant loss. The main reason for this conservative method is that some of the samples generated by a random data augmentation method might be too difficult to train networks. Such samples become "outliers" that could disturb the convergence of the network training. Thus in each mini-batch, HAEM identifies $N + 1$ hardest samples and discards the hardest one to define the hard sample set.

5.4.2 Pose-Based Data Balancing

Existing facial landmark localization methods have achieved good performance for faces in the wild. However, extreme pose variations are still very challenging. To mitigate this problem, Feng et al. [20] proposed a simple but very effective Pose-based Data Balancing (PDB) strategy. PDB argues that the difficulty for accurately localizing faces with large poses is mainly due to data imbalance. This is a well-known problem in many computer vision applications [21].

To perform pose-based data balancing, PDB applies Principal Component Analysis (PCA) to the aligned shapes and projects them to a one dimensional space defined by the shape eigenvector (pose space) controlling pose variations. To be more specific, for a training dataset $\{\mathbf{s}_i\}_{i=1}^{N}$ with N samples, where $\mathbf{s}_i \in \mathbb{R}^{2L}$ is the ith training shape vector consisting of the 2D coordinates of all the L landmarks, the use of Procrustes Analysis aligns all the training shapes to a reference shape, i.e. the mean shape, using rigid transformations. Then PDB approximates any training shape or a new shape, $\mathbf{s}$, using a statistical linear shape model:

$$\mathbf{s} \approx \bar{\mathbf{s}} + \sum_{j=1}^{N_s} p_j \mathbf{s}_j^*, \tag{5.11}$$

where $\bar{\mathbf{s}} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{s}_i$ is the mean shape over all the training samples, $\mathbf{s}_j^*$ is the jth eigenvector obtained by applying PCA to all the aligned training shapes and p_j is the coefficient of the jth shape eigenvector. Among those shape eigenvectors, we can find an eigenvector, usually the first one, that controls the yaw rotation of a face. We denote this eigenvector as $\hat{\mathbf{s}}$. Then we can obtain the pose coefficient of each training sample $\mathbf{s}_i$ as:

$$\hat{p}_i = \hat{\mathbf{s}}^T(\mathbf{s}_i - \bar{\mathbf{s}}). \tag{5.12}$$

The distribution of the pose coefficients of all the AFLW training samples is shown in Fig. 5.10. According to the Fig. 5.10, it can be seen that the AFLW dataset is not well-balanced in terms of pose variation.

With the pose coefficients of all the training samples, PDB first categorizes the training dataset into K subsets. Then it balances the training data by duplicating the samples falling into the subsets of lower cardinality. To be more specific, the number of training samples in

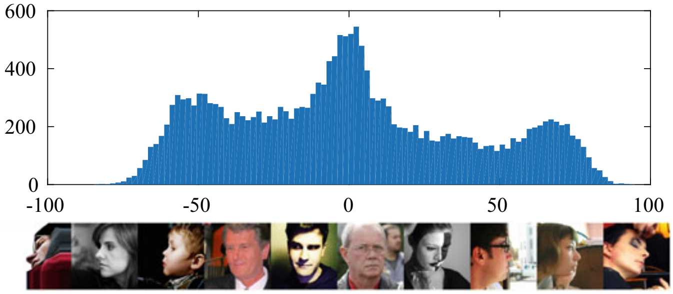


Fig. 5.10 Distribution of the head poses of the AFLW training set

the kth subset is denoted as B_k, and the maximum size of the K subsets is denoted as B^*. To balance the whole training dataset in terms of pose variation, PDB adds more training samples to the kth subset by randomly sampling $B^* - B_k$ samples from the original kth subset. Then all the subsets have the size of B^* and the total number of training samples is increased from $\sum_{k=1}^{K} B_k$ to KB^*. It should be noted that pose-based data balancing is performed before network training by randomly duplicating some training samples of each subset of lower occupancy. After pose-based data balancing, the training samples of each mini-batch are randomly sampled from the balanced training dataset for network training. As the samples with different poses have the same probability to be sampled for a mini-batch, the network training is pose-balanced.

5.5 Landmark Localization in Specific Scenarios

5.5.1 3D Landmark Localization

3D landmark localization aims to locate the 3D coordinates, including 2D positions and depth, of landmarks. The 2D landmark setting assumes that each landmark can be detected by its visual patterns. However, when faces deviate from the frontal view, the contour landmarks become invisible due to self-occlusion. In medium poses, this problem can be addressed by changing the semantic positions of contour landmarks to the silhouette, which is termed landmark marching [62]. However, in large poses where half of the face is occluded, some landmarks are inevitably invisible. In this case, the 3D landmark setting is employed to make the semantic meanings of landmarks consistent, and the face shape can be robustly recovered. As shown in Fig. 5.11, 3D landmarks are always located in their semantic positions, and they should be detected even if they are self-occluded.

In recent years, 3D face alignment has achieved satisfying performance. The methods can be divided into two categories: model-based methods and non-model-based methods. The former performs the 3D face alignment by fitting a 3D Morphable Model (3DMM),

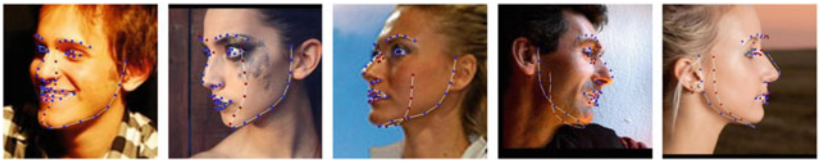


Fig. 5.11 Examples of 3D landmark localization. The blue/red ones indicate visible/invisible landmarks

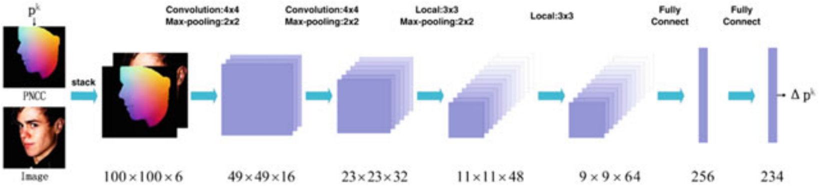


Fig. 5.12 The overview of 3DDFA. At *k*th iteration, 3DDFA takes the images and the projected normalized coordinate code (PNCC) generated by $\mathbf{p}^k$ as inputs and uses a convolutional neural network to predict the parameter update $\Delta\mathbf{p}^k$

which provides a strong prior of face topology. The latter extracts features from the image and directly regresses that to the 3D landmarks by deep neural networks.

5.5.1.1 3D Dense Face Alignment (3DDFA)

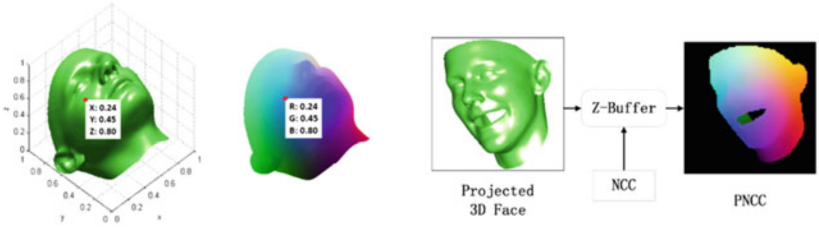
Estimating depth information from a monocular image is an ill-posed problem, and a feasible solution to realize 3D face alignment is introducing a strong 3D face prior. The 3D Dense Face Alignment (3DDFA) is a typical model-based method, which fits a 3DMM by a cascaded convolutional neural network to recover the 3D dense shape. Since the 3DMM is topology-unified, the 3D landmarks can be easily indexed after 3D shape recovery. An overview of 3DDFA is shown in Fig. 5.12. Specifically, the 3D face shape is described as:

$$\mathbf{S} = \overline{\mathbf{S}} + \mathbf{A}_{id}\alpha_{id} + \mathbf{A}_{exp}\alpha_{exp}, \tag{5.13}$$

where $\mathbf{S}$ is the 3D face shape, $\overline{\mathbf{S}}$ is the mean shape, $\mathbf{A}_{id}$ is the principle axes for identity, and $\mathbf{A}_{exp}$ is the principle axes for expression, α_{id} and α_{exp} are the identity and expression parameters that need to be estimated. To obtain the 2D positions of the 3D vertices, the 3D face is projected to the image plane by the weak perspective projection:

$$V(\mathbf{p}) = f * \mathbf{Pr} * \mathbf{R} * (\overline{\mathbf{S}} + \mathbf{A}_{id}\alpha_{id} + \mathbf{A}_{exp}\alpha_{exp}) + \mathbf{t}_{2d}, \tag{5.14}$$

where f is the scalar parameter, $\mathbf{Pr}$ is the orthographic projection matrix $\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$, $\mathbf{R}$ is the rotation matrix derived from the rotation angles $pitch$, yaw, $roll$, and $\mathbf{t}_{2d}$ is the translation



(a) NCC (b) PNCC

Fig. 5.13 The illustration of the Normalized Coordinate Code (NCC) and the Projected Normalized Coordinate Code (PNCC). NCC denotes the position as its texture ($\text{NCC}_x = R, \text{NCC}_y = G, \text{NCC}_z = B$) and PNCC is generated by rendering the 3D face with NCC as its colormap

vector. Parameters for shape recovery are collected as $\mathbf{p} = [f, pitch, yaw, roll, \mathbf{t}_{2d}, \alpha_{id}, \alpha_{exp}]^T$, and the purpose of 3DDFA is to estimate $\mathbf{p}$ from the input image.

3DDFA is a cascaded-regression-based method that employs several networks to update the parameters step by step. A specially designed feature Projected Normalized Coordinate Code (PNCC) is proposed to reflect the fitting accuracy, which is formulated as:

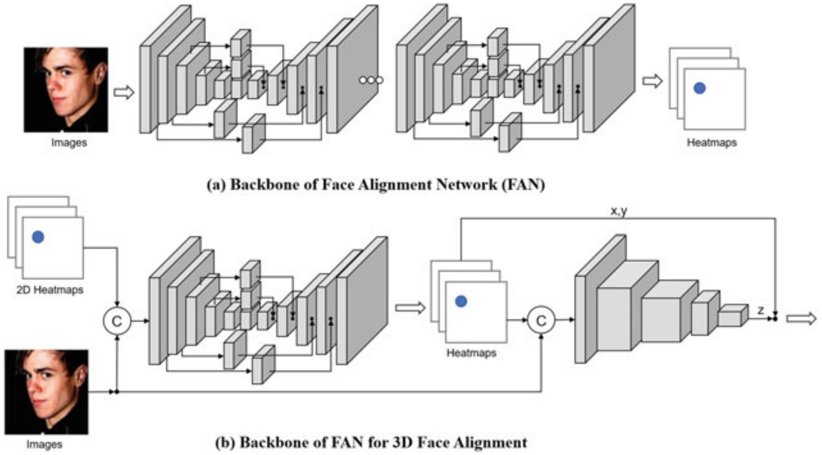
$$\begin{aligned} \text{NCC}_d &= \frac{\overline{\mathbf{S}}_d - \min(\overline{\mathbf{S}}_d)}{\max(\overline{\mathbf{S}}_d) - \min(\overline{\mathbf{S}}_d)} \quad (d = x, y, z), \\ \text{PNCC} &= \textit{Z-Buffer}(V(\mathbf{p}), \text{NCC}), \end{aligned} \tag{5.15}$$

where $\overline{\mathbf{S}}$ is the mean shape of 3DMM, $\textit{Z-Buffer}(\nu, \tau)$ is the render operation that renders 3D mesh ν colored by τ to an image. PNCC represents the 2D locations of the visible 3D vertices on the image plane. Note that both NCC and PNCC have three channels for x, y, z, which is similar to RGB, and they can be shown in color as in Fig. 5.13.

At the kth iteration, 3DDFA constructs PNCC by the current parameter $\mathbf{p}^k$ and concatenates it with the image as input. Then, a neural network is adopted to predict the parameter update $\Delta\mathbf{p}^k$:

$$\Delta\mathbf{p}^k = Net^k(\mathbf{I}, \text{PNCC}(\mathbf{p}^k)). \tag{5.16}$$

Afterward, the parameter for the $k+1$ iteration is updated: $\mathbf{p}^{k+1} = \mathbf{p}^k + \Delta\mathbf{p}^k$, and another network is adopted to further update the parameters until convergence. By incorporating 3D prior, 3DDFA localizes the invisible landmarks in large poses, achieving the-state-of-the-art performance. However, it is limited by the computation cost since it cascades several networks to progressively update the fitting result. To deploy 3DDFA on lightweight devices, 3DDFAv2 [63] employs a mobilenet [64] to directly regress the target parameters and also achieves satisfactory performance.



(a) Backbone of Face Alignment Network (FAN)

(b) Backbone of FAN for 3D Face Alignment

Fig. 5.14 **a** The backbone of the Face Alignment Network (FAN). It consists of stacked Hourglass networks [55] in which the bottleneck blocks are replaced with the residual block of [56]. **b** The illustration of FAN for 3D face alignment. The network takes the images and their corresponding 2D landmark heatmaps as input to regress the heatmaps of the projected 3D landmarks, which are then concatenated with the image to regress the depth values of landmarks

5.5.1.2 Face Alignment Network (FAN)

Face Alignment Network (FAN) [52] is a non-model-based method for 3D face alignment, which trains a neural network to regress the landmark heatmaps. FAN constructs a strong backbone to localize 3D landmarks, shown in Fig. 5.14a. Specifically, FAN consists of four stacked hourglass networks [55], and the bottleneck blocks in each hourglass are replaced with the hierarchical, parallel, and multi-scale residual block [56] to further improve the performance. Given an input image, FAN utilizes the network to regress the landmark heatmaps, where each channel of the heatmap is a 2D Gaussian centered at the corresponding landmark's location with a standard deviation of one pixel.

To realize the regression of 3D positions, FAN designs a guided-by-2D-landmarks network to convert 2D landmarks to 3D landmarks, which bridges the performance gap between the saturating 2D landmark localization and the challenging 3D landmark localization. The overview of FAN for 3D landmark localization is shown in Fig. 5.14b. Specifically, given an RGB image and their corresponding 2D landmark heatmaps as input, FAN first regresses the heatmaps of the projected 3D landmarks, obtaining the x, y of 3D landmarks. Then, the projected 3D landmark heatmaps are combined with the input image and sent to a followed network to regress the depth value of each landmark, obtaining the full x, y, z coordinates of 3D landmarks.

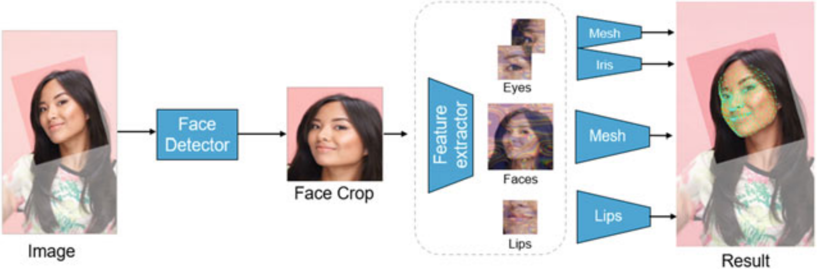


Fig. 5.15 The pipeline of the MediaPipe. Given an input image, the face region is first cropped by the face detector and then sent to the feature extractor. After that, the model is split into several sub-models to predict the global landmarks and important local landmarks including eyes and lips

5.5.1.3 MediaPipe

MediaPipe [60] is a widely used pipeline for 2D and 3D landmark localization. It is proposed to meet the real-time application requirements for face localization such as AR make-up, eye tracking, AR puppeteering, etc. Different from the cascaded framework, MediaPipe uses a single model to achieve comparable performance. The pipeline of MediaPipe is shown in Fig. 5.15. The network first extracts the global feature map from the cropped images, and then the network is split into several sub-networks. One sub-network predicts the 3D face mesh, including 3D landmarks, and outputs the regions of interest (eyes and lips). The remaining two sub-networks are employed to estimate the local landmarks of eyes and lips, respectively. The output of MediaPipe is a sparse mesh composed of 468 points. Through the lightweight architecture [61] and the region-specific heads for meaningful regions, MediaPipe has good efficiency and achieves comparable performance compared with the cascaded methods, realizing the real-time on-device inference.

5.5.1.4 3D Landmark Data

One of the main challenges of 3D landmark localization is the lack of data. Acquiring high-precision 3D face models requires expensive devices and a fully controlled environment, making large-scale data collection infeasible. To overcome this bottleneck, current methods usually label 2D projections of 3D landmarks as an alternative solution. However, it is still laborious since the self-occluded parts have to be guessed by intuition. In recent years, 300W-LP [40, 85], AFLW2000-3D [40, 85], and Menpo-3D [84] have been popular data sets for building 3D landmark localization systems. In addition to hand annotation, training data can be generated by virtual synthesis. Face Profiling [40, 85] proposes to recover a textured 3D mesh from a 2D face image and rotate the 3D mesh to given rotation angles, which can be rendered to generate virtual data, shown in Fig. 5.16. Through face profiling, not only the face samples in large poses (yaw angle up to 90°) can be obtained, but also the dataset can be augmented to any desired scale.

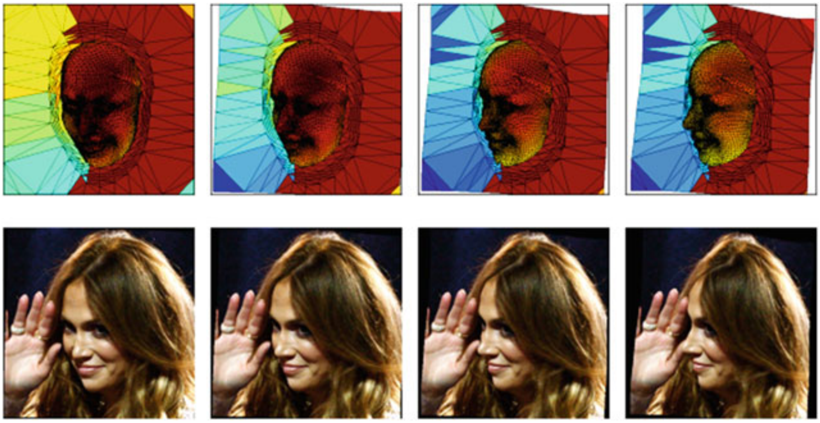


Fig. 5.16 The face profiling process

5.5.2 Landmark Localization on Masked Face

Since the outbreak of the worldwide pandemic COVID-19, facial landmark localization has encountered the great challenge of mask occlusion. First, the collection of masked face data is costly and difficult, especially during the spread of COVID-19. Second, the masked facial image suffers from severe occlusion, making the landmarks more difficult to detect. Taking the 106-point landmark setting as an example, there are around 27 nose and mouth points occluded by the facial mask (Fig. 5.18), which brings not only additional difficulty to landmark detection, but also adverse uncertainty to the ground-truth labeling. These issues cause serious harm to the deep-learning-based landmark localization that relies on labeled data.

It can be perceived that most of the issues lie in the masked face data. Therefore, a feasible and straightforward solution is synthesizing photo-realistic masked face images from mask-free ones, so as to overcome the problems of data collection and labeling. One popular approach [14] , as shown in Fig. 5.17, is composed of three steps, i.e., 3D reconstruction, mask segmentation, and re-rendering of the blended result. Given the source masked face and the target mask-free face, their 3D shapes are first recovered by a 3D face reconstruction method (such as PRNet [53]) to warp the image pixels to the UV space to generate the UV texture. Second, the mask area in the source image is detected by a facial segmentation method [90], which is also warped to the UV space to get a UV mask. Finally, the target UV texture is covered by the UV mask, and the synthesized target texture is re-rendered to the original 2D plane.

There are two benefits of this practice. First, a large number of masked face images can be efficiently produced with geometrically-reasonable and photo-realistic masks, and the mask styles are fully controlled. Second, once the target image has annotated landmarks, the synthesized one does not have to be labeled again. It can directly inherit the ground-truth

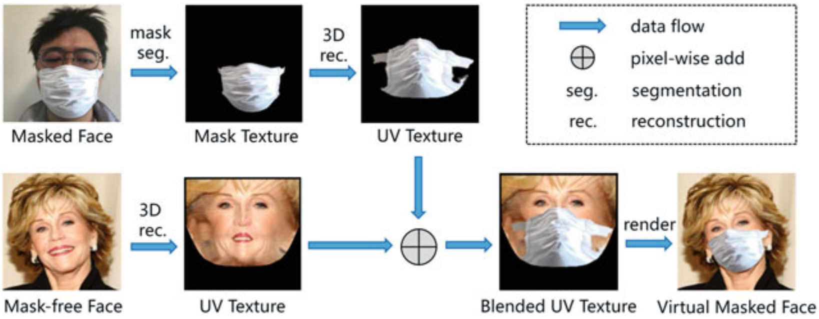
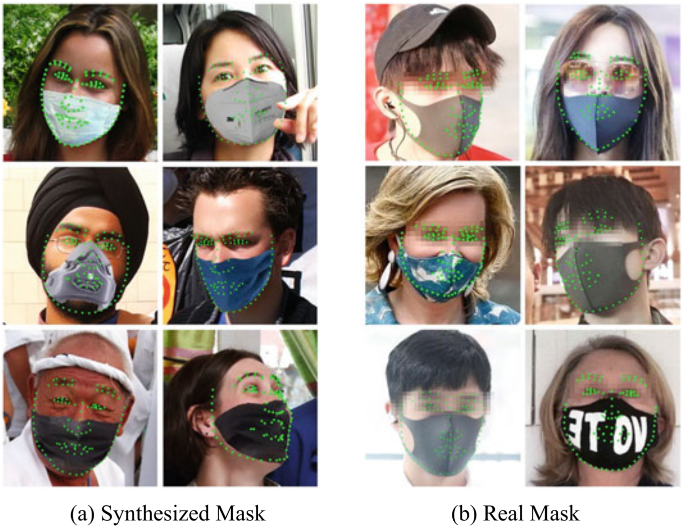


Fig. 5.17 Adding virtual mask to face images by 3D reconstruction and face segmentation


(a) Synthesized Mask

(b) Real Mask

Fig. 5.18 Examples of synthesized and real masked face images [1]

landmarks for training and testing (Fig. 5.18a). With the synthesized masked face images, the mask-robust landmark detection model can be built in the similar manner as in the mask-free condition.

5.5.3 Joint Face Detection and Landmark Localization

The joint detection of face boxes and landmarks has been studied since the early ages when deep learning begins to thrive in biometrics. The initial motivation of joint detection is to boost face detection itself by incorporating landmarks to handle certain hard cases, e.g., large pose, severe occlusion, and heavy cosmetics [5, 6]. Afterward, the community

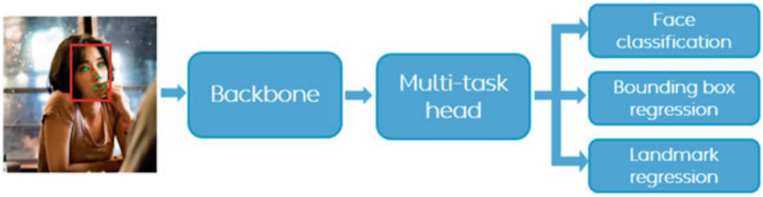


Fig. 5.19 The typical framework of joint detection of face and landmark

pays increasing attention to merging the two tasks as one. The advantages are three-fold: First, the two highly correlated tasks benefit each other when the detector is trained by the annotations from both sides. Second, the unified style brings better efficiency to the whole pipeline of face-related applications, as the two detection tasks can be accomplished by a single lightweight model. Finally, the joint model can be conveniently applied in many tasks, including face recognition, simplifying the implementation in practice. Despite the obvious advantages of the multi-task framework, building such a system requires more expensive training data with labels of multiple face attributes, improving the cost of data annotations.

Networks. The typical framework of joint face and landmark detection is shown in Fig. 5.19. The input image contains human faces that occur with arbitrary pose, occlusion, illumination, cosmetics, resolution, etc. The backbone extracts an effective feature from the input image and feeds it into the multi-task head. The multi-task head outputs the joint detection results, including at least three items, i.e., face classification, face bounding box coordinates, and landmark coordinates. Beyond typical tasks, some methods also predict the head pose, gender [8], and 3D reconstruction [11] simultaneously. The major backbones include FPN [10], Cascaded-CNN [7], multi-scale fusion within rapidly digested CNN [9], YOLO-vX style [3], etc. The former two make full use of hierarchical features and predict fine results, and the latter two have excellent efficiency for CPU-real-time applications.

Learning objectives. The framework should be trained with multiple objectives to perform joint predictions. Equation (5.17) is the typical loss formulation for multiple objective training. $\mathcal{L}_{face-cls}$ is the cross-entropy loss for face classification, which predicts the confidence of whether the candidate is a human face. $\mathcal{L}_{bbox-reg}$ is defined as the L2 or smooth L1 distance between the coordinates of the predicted bounding box and the ground truth, supervising the model to learn the bounding box locations. Similarly, $\mathcal{L}_{lm-reg}$ supervises the model to predict the landmark coordinates in the same way.

$$\mathcal{L} = \alpha_1 \beta_1 \mathcal{L}_{face-cls} + \alpha_2 \beta_2 \lambda \mathcal{L}_{bbox-reg} + \alpha_3 \beta_3 \lambda \mathcal{L}_{lm-reg}, \tag{5.17}$$

where $\{\alpha_1, \alpha_2, \alpha_3\} \in \mathbb{R}$ are the weights for balancing the training toward three objectives, $\{\beta_1, \beta_2, \beta_3\} \in \{0, 1\}$ are binary indicators that activate the supervision if the corresponding annotation presents in the training sample, and $\lambda \in \{0, 1\}$ is applied to activate the supervision of bounding box and landmark if the candidate's ground truth is human face [9]. It is worth noting that the incorporation of β enables the training on partially annotated datasets.

Datasets. The dataset most commonly used for joint detection is the WIDER FACE [13] dataset with the supplementary annotations [11]. The initial purpose of WIDER FACE is to train and evaluate face detection models. The supplementary annotation provides five-point landmarks on each face, enabling the usage for the joint detection task. Owing to the wide utilization of this dataset, most joint detection models predict five-point landmarks, which are sufficient for face alignment in most cases. Besides, some models [8, 30] trained by the 300W [57] dataset predict 68 landmarks for joint detection.

5.6 Evaluations of the State of the Arts

In this section, we introduce how to evaluate the performance of a landmark localization method, including various datasets and evaluation metrics. The evaluation results of representative methods on different datasets are also collected and demonstrated.

5.6.1 Datasets

In recent years, many datasets have been collected for training and testing of 2D facial landmark localization, including COFW [67], COFW-68 [72], 300W [65], 300W-LP [85], WFLW [68], Menpo-2D [83], AFLW [66], AFLW-19 [86], AFLW-68 [87], MERL-RAV [77] and WFLW-68 [39], which are listed in Table 5.1. We introduce some representative datasets as follows:

Table 5.1 An overview of 2D facial landmark datasets. “Train” and “Test” are the number of samples in the training set and the test set, respectively. “Landmark Num.” represents the number of annotated landmarks

Dataset	Year	Train	Test	Landmark Num.
AFLW [66]	2011	20, 000	4, 386	21
300W [65]	2013	3, 148	689	68
COFW [67]	2013	1, 345	507	29
COFW-68 [72]	2014	–	507	68
300W-LP [85]	2016	61, 225	–	68
Menpo-2D [83]	2016	7, 564	7, 281	68/39
AFLW-19 [86]	2016	20, 000	4, 386	19
WFLW [68]	2018	7, 500	2, 500	98
AFLW-68 [87]	2019	20, 000	4, 386	68
MERL-RAV [77]	2020	15, 449	3, 865	68
WFLW-68 [39]	2021	7, 500	2, 500	68

300W contains 3, 837 images, some images may have more than one face. Each face is annotated with 68 facial landmarks. The 3, 148 training images are from the full set of AFW [69] (337 images), the training part of LFPW [70] (811 images), and HELEN [71] (2, 000 images). The test set is divided into a common test set and a challenging set. The common set with 554 images comes from the testing part of LFPW (224 images) and HELEN (330 images). The challenging set with 135 images is from the full set of IBUG [65]. 300W-LP [85] augments the pose variations of 300W by the face profiling technique and generates a large data set with 61, 225 samples, much of which are in profile.

COFW contains 1, 007 images with 29 annotated landmarks. The training set with 1, 345 samples is the combination of 845 LFPW samples and 500 COFW samples. The test set with 507 samples has two cases. They are annotated with 29 landmarks (the same as the training set) or 68 landmarks, and the latter is called COFW-68 [72]. Most faces in COFW have large variations in occlusion.

AFLW contains 25, 993 faces with at most 21 visible facial landmarks annotated, but excludes the annotations of invisible landmarks. A protocol [86] is built on the original AFLW and divides the dataset into 20, 000 training samples and 4, 386 test samples. The dataset has large pose variations, especially has thousands of faces in profile. AFLW-19 [86] builds a 19-landmark annotation by removing the 2 ear landmarks. AFLW-68 [87] follows the configuration in 300 W and re-annotates the images with 68 facial landmarks.

Menpo-2D has a training set with 7, 564 images, including 5, 658 front faces and 1, 906 profile faces, and a test set with 7, 281 images, including 5, 335 front faces and 1, 946 profile faces. There are two settings for different poses. The front faces are annotated by 68 landmarks, and the profile faces are annotated by 39 landmarks.

WFLW contains 7, 500 images for training and 2, 500 images for testing. Each face in WFLW is annotated with 98 landmarks and some attributes such as occlusion, make-up, expression and blur. WFLW-68 [39] converts the original 98 landmarks to 68 landmarks for convenient evaluation.

5.6.2 Evaluation Metric

There are three commonly utilized metrics to evaluate the precision of landmark localization, including Normalized Mean Error (NME), Failure Rate (FR) and Cumulative Error Distribution (CED).

Normalized Mean Error (NME) is one of the most widely used metrics in face alignment, which is defined as:

$$\mathrm{NME} = \frac{1}{\mathrm{M}} \sum_{\mathrm{i}=1}^{\mathrm{M}} \frac{||\mathbf{P}_{\mathrm{i}} - \mathbf{P}_{\mathrm{i}}{}^{*}||_2}{\mathrm{d}}, \tag{5.18}$$

where $\{\mathbf{P}_i\}$ is the predicted landmark coordinates, $\{\mathbf{P}_i{}^*\}$ is the ground-truth coordinates, M is the total number of landmarks, and d is the distance between outer eye corners (inter-ocular) [39, 68, 75, 79, 82]) or pupil centers (inter-pupils [76, 80]). It can be seen that the error is

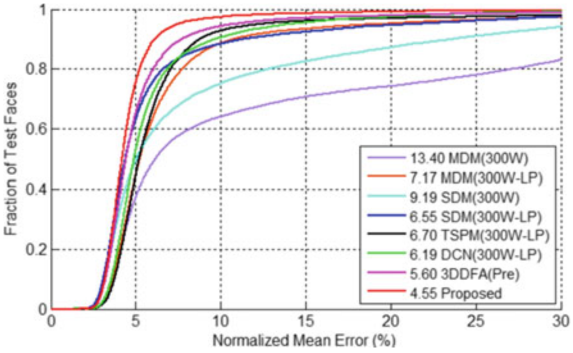


Fig. 5.20 An example of CED curve from [40]. In the curve, x is NME and y is the proportion of samples in the test set whose NMEs are less than x

normalized by d to reduce the deviation caused by face scale and image size. In some cases, the image size [39] or face box size [77] is also used as the normalization factor d. A smaller NME indicates better performance.

Failure Rate (FR) is the percentage of samples whose NMEs are higher than a certain threshold f, denoted as FR_f (f is usually set to 0.1) [57, 68, 92]. A smaller FR means better performance.

Cumulative Error Distribution (CED) is defined as a curve (x, y), where x indicates NME and y is the proportion of samples in the test set whose NMEs are less than x. Figure 5.20 shows an example of CED curve, which provides a more detailed summary of landmark localization performance. Based on CED, the **Area Under the Curve (AUC)** can be obtained by the area enclosed between the CED curve and the x-axis, whose integral interval is $x = 0$ to a threshold $x = f$, denoted as AUC_f. A larger AUC means better performance.

5.6.3 Comparison of the State of the Arts

We demonstrate the performance of some state-of-the-art methods from 2018 to 2022 on commonly used datasets, including LAB [68], SAN [73], HG-HSLE [74], AWing [76], DeCaFA [78], RWing [38], HRNet [75], LUVLi [77], SDL [81], PIPNet [39], HIH [79], ADNet [80], and SLPT [82]. It is worth noting that the reported results should not be compared directly because the model sizes and training data are different.

300W: Table 5.2 summarizes the results on the most commonly used dataset 300W, with three test subsets of “common”, “challenging”, and “full”. The NME of the 68 facial landmarks is calculated to measure the performance. All the results are collected from the corresponding papers.

COFW: Table 5.3 summarizes the results on the COFW and COFW-68, which mainly measure the robustness to occlusion. There are two protocols, the within-dataset protocol (COFW) and cross-dataset protocol (COFW-68). For the within-dataset protocol, the model is trained with $1,345$ images and validated with 507 images on COFW. The NME and $\mathrm{FR}_{0.1}$

Table 5.2 Performance comparison on 300 W, “Common”, “Challenge”, and “Full” represent common set, challenging set, and full set of 300W, respectively. “Backbone” represents the model architecture used by each method

Method	Year	Backbone	NME(%, inter-ocular)		
			Full	Common	Challenge
LAB [68]	2018	ResNet-18	3.49	2.98	5.19
SAN [73]	2018	ITN-CPM	3.98	3.34	6.60
HG-HSLE [74]	2019	Hourglass	3.28	2.85	5.03
AWing [76]	2019	Hourglass	3.07	2.72	4.52
DeCaFA [78]	2019	Cascaded U-net	3.39	2.93	5.26
HRNet [75]	2020	HRNetV2-W18	3.32	2.87	5.15
LUVLi [77]	2020	DU-Net	3.23	2.76	5.16
SDL [81]	2020	DA-Graph	3.04	2.62	4.77
PIPNet [39]	2021	ResNet-101	3.19	2.78	4.89
HIH [79]	2021	2 Stacked HGs	3.33	2.93	5.00
ADNet [80]	2021	Hourglass	2.93	2.53	4.58
SLPT [82]	2022	HRNetW18C-lite	3.17	2.75	4.90

of the 29 landmarks are utilized for comparison. For the cross-dataset protocol, the training set includes the complete 300W dataset (3, 837 images), and the test set is COFW-68 (507 images). The NME and $FR_{0.1}$ of the 68 landmarks are reported. All the results are collected from the corresponding papers.

WFLW: Table 5.4 summarizes the results on WFLW. The test set is divided into six subsets to evaluate the models in various specific scenarios, which are pose (326 images), expression (314 images), illumination (698 images), make-up (206 images), occlusion (736 images), and blur (773 images). The three metrics of NME, $FR_{0.1}$ and $AUC_{0.1}$ of the 98 landmarks are employed to demonstrate the stability of landmark localization. The results of SAN are from the supplemental material of [82]. The results of LUVLi are from the supplemental materials of [77]. The results of SLPT are from the supplemental materials of [82]. For HRNet, the NME is from [75], and the $FR_{0.1}$ and $AUC_{0.1}$ are from [81]. The other results are from the corresponding papers.

Table 5.3 Performance comparison on COFW and COFW-68. The threshold of Failure Rate (FR) and Area Under the Curve (AUC) are set to 0.1

Method	Year	Backbone	COFW				COFW-68	
			Inter-Ocular		Inter-Pupil		Inter-Ocular	
			NME(%)	$\text{FR}_{0.1}$(%)	NME(%)	$\text{FR}_{0.1}$(%)	NME(%)	$\text{FR}_{0.1}$(%)
LAB [68]	2018	ResNet-18	3.92	0.39	–	–	4.62	2.17
AWing [76]	2019	Hourglass	–	–	4.94	0.99		
RWing [38]	2020	CNN-6&8	–	–	4.80	–	–	–
HRNet [75]	2020	HRNetV2-W18	3.45	0.19	–	–	–	–
SDL [81]	2020	DA-Graph	–	–	–	–	4.22	0.39
PIPNet [39]	2021	ResNet-101	3.08	–	–	–	4.23	–
HIH [79]	2021	2 Stacked HGs	3.28	0.00	–	–	–	–
ADNet [80]	2021	Hourglass	–	–	4.68	0.59	–	–
SLPT [82]	2022	HRNetW18C-lite	3.32	0.00	4.79	1.18	4.10	0.59

5.7 Conclusion

Landmark localization has been the cornerstone of many widely used applications. For example, face recognition utilizes landmarks to align faces, face AR applications use landmarks to enclose eyes and lips, and face animation fits 3D face models by landmarks. In this chapter, we have discussed typical methods of landmark localization, including coordinate regression and heatmap regression, and some special landmark localization scenarios. Although these strategies have made great progress and enabled robust localization in most cases, there are still many challenging problems remaining to be addressed in advanced applications, including faces in profile, large-region occlusion, temporal consistency, and pixel-level accuracy. With the development of face applications, the benchmark of landmarks on accuracy, robustness, and computation cost becomes higher and higher and more sophisticated landmark localization strategies are needed.

Table 5.4 Performance comparison on WFLW. All, Pose, Expr., Illu., M.u., Occ. and Blur represent full set, pose set, expression set, illumination set, make-up set, occlusion set, and blur set of WFLW, respectively. All results used inter-ocular distance for normalization. The threshold of Failure Rate (FR) and Area Under the Curve (AUC) are set to 0.1

Metric	Method	Year	Backbone	Pose	Expr.	Illu.	M.u.	Occ.	Blur	All
NME (%)	LAB [68]	2018	ResNet-18	10.24	5.51	5.23	5.15	6.79	6.32	5.27
	SAN [73]	2018	ITN-CPM	10.39	5.71	5.19	5.49	6.83	5.80	5.22
	AWing [76]	2019	Hourglass	7.38	4.58	4.32	4.27	5.19	4.96	4.36
	DeCaFA [78]	2019	Cascaded U-net	8.11	4.65	4.41	4.63	5.74	5.38	4.62
	RWing [38]	2020	CNN-6&8	9.79	6.16	5.54	6.65	7.05	6.41	5.60
	HRNet [75]	2020	HRNetV2-W18	7.94	4.85	4.55	4.29	5.44	5.42	4.60
	LUVLi [77]	2020	DU-Net	7.56	4.77	4.30	4.33	5.29	4.94	4.37
	SDL [81]	2020	DA-Graph	7.36	4.49	4.12	4.05	4.98	4.82	4.21
	PIPNet [39]	2021	ResNet-101	7.51	4.44	4.19	4.02	5.36	5.02	4.31
	HIH [79]	2021	2 Stacked HGs	7.20	4.28	4.42	4.03	5.00	4.79	4.21
	ADNet [80]	2021	Hourglass	6.96	4.38	4.09	4.05	5.06	4.79	4.14
	SLPT [82]	2022	HRNetW18C-lite	6.96	4.45	4.05	4.00	5.06	4.79	4.14
$\text{FR}_{0.1}$ (%)	LAB [68]	2018	ResNet-18	28.83	6.37	6.73	7.77	13.72	10.74	7.56
	SAN [73]	2018	ITN-CPM	27.91	7.01	4.87	6.31	11.28	6.60	6.32
	AWing [76]	2019	Hourglass	13.50	2.23	2.58	2.91	5.98	3.75	2.84
	DeCaFA [78]	2019	Cascaded U-net	21.40	3.73	3.22	6.15	9.26	6.61	4.84

(continued)

Table 5.4 (continued)

Metric	Method	Year	Backbone	Pose	Expr.	Illu.	M.u.	Occ.	Blur	All
	RWing [38]	2020	CNN-6&8	34.36	9.87	7.16	9.71	15.22	10.61	8.24
	HRNet [75]	2020	HRNetV2-W18	23.01	3.50	4.72	2.43	8.29	6.34	4.64
	LUVLi [77]	2020	DU-Net	15.95	3.18	2.15	3.40	6.39	3.23	3.12
	SDL [81]	2020	DA-Graph	15.95	2.86	2.72	1.45	5.29	4.01	3.04
	HIH [79]	2021	2 Stacked HGs	14.41	2.55	2.15	1.46	5.71	3.49	2.84
	ADNet [80]	2021	Hourglass	12.72	2.15	2.44	1.94	5.79	3.54	2.72
	SLPT [82]	2022	HRNetW18C-lite	12.27	2.23	1.86	3.40	5.98	3.88	2.76
$AUC_{0.1}$	LAB [68]	2018	ResNet-18	0.235	0.495	0.543	0.539	0.449	0.463	0.532
	SAN [73]	2018	ITN-CPM	0.236	0.462	0.555	0.522	0.456	0.493	0.536
	AWing [76]	2019	Hourglass	0.312	0.515	0.578	0.572	0.502	0.512	0.572
	DeCaFA [78]	2019	Cascaded U-net	0.292	0.546	0.579	0.575	0.485	0.494	0.563
	HRNet [75]	2020	HRNetV2-W18	0.251	0.510	0.533	0.545	0.459	0.452	0.524
	RWing [38]	2020	CNN-6&8	0.290	0.465	0.518	0.510	0.456	0.456	0.518
	LUVLi [77]	2020	DU-Net	0.310	0.549	0.584	0.588	0.505	0.525	0.557
	SDL [81]	2020	DA-Graph	0.315	0.566	0.595	0.604	0.524	0.533	0.589
	HIH [79]	2021	2 Stacked HGs	0.332	0.583	0.605	0.601	0.525	0.546	0.593
	ADNet [80]	2021	Hourglass	0.344	0.523	0.580	0.601	0.530	0.548	0.602
	SLPT [82]	2022	HRNetW18C-lite	0.348	0.574	0.601	0.605	0.515	0.535	0.595

Acknowledgements We acknowledge Xiaomei Zhang, Chang Yu, Zidu Wang for helpful discussions and contributions related to this chapter. This work is supported in part by Chinese National Natural Science Foundation Projects under Grants 62176256, and in part by the Youth Innovation Promotion Association CAS under Grant Y2021131.

References

1. Xiang, M., Liu, Y., Liao, T., Zhu, X., Yang, C., Liu, W., Shi, H.: The 3rd grand challenge of lightweight 106-point facial landmark localization on masked faces. In: International Conference on Multimedia & Expo Workshops, pp. 1–6. IEEE (2021)
2. Sha, Y., Zhang, J., Liu, X., Wu, Z., Shan, S.: Efficient face alignment network for masked face. In: International Conference on Multimedia & Expo Workshops, pp. 1–6. IEEE (2021)
3. Liu, Y., Chen, C., Zhang, M., Li, J., Xu, W.: Joint face detection and landmark localization based on an extremely lightweight network. In: International Conference on Image and Graphics, pp. 351–361. Springer, Cham (2021)
4. Sha, Y.: Towards occlusion robust facial landmark detector. In: International Conference on Automatic Face and Gesture Recognition, pp. 1–8. IEEE (2021)
5. Li, Y., Sun, B., Wu, T., Wang, Y.: Face detection with end-to-end integration of a convnet and a 3d model. In: European Conference on Computer Vision, pp. 420–436. Springer, Cham (2016)
6. Chen, D., Hua, G., Wen, F., Sun, J.: Supervised transformer network for efficient face detection. In: European Conference on Computer Vision, pp. 122–138. Springer, Cham (2016)
7. Zhang, K., Zhang, Z., Li, Z., Qiao, Y.: Joint face detection and alignment using multitask cascaded convolutional networks. Signal Process. Lett. IEEE 1499–1503 (2016)
8. Ranjan, R., Patel, V. M., Chellappa, R.: Hyperface: a deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition. Trans. Pattern Anal. Mach. Intell. IEEE. 121–135 (2017)
9. Zhuang, C., Zhang, S., Zhu, X., Lei, Z., Wang, J., Li, S. Z.: Fldet: a cpu real-time joint face and landmark detector. In: International Conference on Biometrics, pp. 1–8. IEEE (2019)
10. Xu, Y., Yan, W., Yang, G., Luo, J., Li, T., He, J.: CenterFace: joint face detection and alignment using face as point. Scientific Programming (2020)
11. Deng, J., Guo, J., Ververas, E., Kotsia, I., Zafeiriou, S.: Retinaface: single-shot multi-level face localisation in the wild. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5203–5212. IEEE (2020)
12. Deng, J., Guo, J., Zafeiriou, S.: Single-stage joint face detection and alignment. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. IEEE (2019)
13. Yang, S., Luo, P., Loy, C. C., Tang, X.: Wider face: a face detection benchmark. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5525–5533. IEEE (2016)
14. Wang, J., Liu, Y., Hu, Y., Shi, H., Mei, T.: Facex-zoo: a pytorch toolbox for face recognition. In: Proceedings of the 29th ACM International Conference on Multimedia, pp. 3779–3782 (2021)
15. Wen, T., Ding, Z., Yao, Y., Ge, Y., Qian, X.: Towards efficient masked-face alignment via cascaded regression. In: International Conference on Multimedia & Expo Workshops (ICMEW), pp. 1–5. IEEE (2021)
16. Hu, H., Wang, C., Jiang, T., Guo, Z., Han, Y., Qian, X.: Robust and efficient facial landmark localization. In: International Conference on Multimedia & Expo Workshops (ICMEW), pp. 1–7. IEEE (2021)

17. Lai, S., Liu, L., Chai, Z., Wei, X.: Light weight facial landmark detection with weakly supervised learning. In: International Conference on Multimedia & Expo Workshops (ICMEW), pp. 1–6. IEEE (2021)
18. Guo, X., Li, S., Yu, J., Zhang, J., Ma, J., Ma, L., Ling, H.: PFLD: a practical facial landmark detector. arXiv preprint arXiv:1902.10859 (2019)
19. Gao, P., Lu, K., Xue, J., Lyu, J., Shao, L.: A facial landmark detection method based on deep knowledge transfer. IEEE Trans. Neural Netw. Learn. Syst. (2021)
20. Feng, Z.H., Kittler, J., Awais, M., Huber, P., Wu, X.J.: Wing loss for robust facial landmark localisation with convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2235–2245 (2018)
21. Shrivastava, A., Gupta, A., Girshick, R.: Training region-based object detectors with online hard example mining. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 761–769 (2016)
22. Wu, Y., Shah, S.K., Kakadiaris, I.A.: GoDP: globally Optimized Dual Pathway deep network architecture for facial landmark localization in-the-wild. Image Vis. Comput. **73**, 1–16 (2018)
23. Cootes, T.F., Walker, K., Taylor, C.J.: View-based active appearance models. In: Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition, pp. 227–232 (2000)
24. Sun, Y., Wang, X., Tang, X.: Deep convolutional network cascade for facial point detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3476–3483 (2013)
25. Rashid, M., Gu, X., Jae Lee, Y.: Interspecies knowledge transfer for facial keypoint detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6894–6903 (2017)
26. Feng, Z.H., Kittler, J., Wu, X.J.: Mining hard augmented samples for robust facial landmark localization with CNNs. IEEE Signal Process. Lett. **26**(3), 450–454 (2019)
27. Xiao, S., Feng, J., Liu, L., Nie, X., Wang, W., Yan, S., Kassim, A.: Recurrent 3d-2d dual learning for large-pose facial landmark detection. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1633–1642 (2017)
28. Huang, Z., Zhou, E., Cao, Z.: Coarse-to-fine face alignment with multi-scale local patch regression. arXiv preprint arXiv:1511.04901 (2015)
29. Yang, J., Liu, Q., Zhang, K.: Stacked hourglass network for robust facial landmark localisation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 79–87 (2017)
30. Deng, J., Trigeorgis, G., Zhou, Y., Zafeiriou, S.: Joint multi-view face alignment in the wild. IEEE Trans. Image Process. **28**(7), 3636–3648 (2019)
31. Girshick, R.: Fast R-Cnn. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1440–1448 (2015)
32. Taigman, Y., Yang, M., Ranzato, M.A., Wolf, L.: Deepface: closing the gap to human-level performance in face verification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1701–1708 (2014)
33. Cao, C., Weng, Y., Lin, S., Zhou, K.: 3D shape regression for real-time facial animation. ACM Trans. Graph. (TOG) **32**(4), 1–10 (2013)
34. Bettadapura, V.: Face expression recognition and analysis: the state of the art. arXiv preprint arXiv:1203.6722 (2012)
35. Thies, J., Zollhofer, M., Stamminger, M., Theobalt, C., Niebner, M.: Face2face: real-time face capture and reenactment of rgb videos. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2016)

36. Trigeorgis, G., Snape, P., Nicolaou, M.A., Antonakos, E., Zafeiriou, S.: Mnemonic descent method: a recurrent process applied for end-to-end face alignment. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4177–4187 (2016)
37. Broy, M.: Software engineering — from auxiliary to key technologies. In: Broy, M., Dener, E. (eds.) Software Pioneers, pp. 10–13. Springer, Heidelberg (2002)
38. Feng, Z.H., Kittler, J., Awais, M., Wu, X.J.: Rectified wing loss for efficient and robust facial landmark localisation with convolutional neural networks. Int. J. Comput. Vis. **128**(8), 2126–2145 (2020)
39. Jin, H., Liao, S., Shao, L.: Pixel-in-pixel net: towards efficient facial landmark detection in the wild. Int. J. Comput. Vis. **129**(12), 3174–3194 (2021)
40. Zhu, X., Liu, X., Lei, Z., Li, S.Z.: Face alignment in full pose range: a 3D total solution. IEEE Trans. Pattern Anal. Mach. Intell. **41**(1), 78–92 (2017)
41. Blanz, V., Vetter, T.: Face recognition based on fitting a 3D morphable model. IEEE Trans. Pattern Anal. Mach. Intell. **25**(9), 1063–1074 (2003)
42. Cootes, T.F., Taylor, C.J., Cooper, D.H., Graham, J.: Training models of shape from sets of examples. In: BMVC92, pp. 9–18. Springer, London (1992)
43. Cootes, T.F., Taylor, C.J.: Combining elastic and statistical models of appearance variation. In: European Conference on Computer Vision, pp. 149–163 (2000)
44. Wang, N., Gao, X., Tao, D., Yang, H., Li, X.: Facial feature point detection: a comprehensive survey. Neurocomputing **275**, 50–65 (2018)
45. Kazemi, V., Sullivan, J.: One millisecond face alignment with an ensemble of regression trees. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1867–1874 (2014)
46. King, D.E.: Dlib-ml: a machine learning toolkit. J. Mach. Learn. Res. **10**, 1755–1758 (2009)
47. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Commun. ACM **60**(6), 84–90 (2017)
48. Yu, R., Saito, S., Li, H., Ceylan, D., Li, H.: Learning dense facial correspondences in unconstrained images. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 4723–4732 (2017)
49. Jourabloo, A., Liu, X.: Pose-invariant 3D face alignment. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 3694–3702 (2015)
50. Liu, Y., Jourabloo, A., Ren, W., Liu, X.: Dense face alignment. In: Proceedings of the IEEE International Conference on Computer Vision Workshops, pp. 1619–1628 (2017)
51. Bulat, A., Tzimiropoulos, G.: Two-stage convolutional part heatmap regression for the 1st 3d face alignment in the wild (3dfaw) challenge. In: European Conference on Computer Vision, pp. 616–624. Springer, Cham (2016)
52. Bulat, A., Tzimiropoulos, G.: How far are we from solving the 2d & 3d face alignment problem? (and a dataset of 230,000 3d facial landmarks). In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1021–1030 (2017)
53. Feng, Y., Wu, F., Shao, X., Wang, Y., Zhou, X.: Joint 3d face reconstruction and dense alignment with position map regression network. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 534–551 (2018)
54. Gu, L., Kanade, T.: 3D alignment of face in a single image. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1305–1312 (2006)
55. Newell, A., Yang, K., Deng, J.: Stacked hourglass networks for human pose estimation. In: European Conference on Computer Vision, pp. 483–499 (2016)
56. Bulat, A., Tzimiropoulos, G.: Binarized convolutional landmark localizers for human pose estimation and face alignment with limited resources. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 3706–3714 (2017)

57. Sagonas, C., Antonakos, E., Tzimiropoulos, G., Zafeiriou, S., Pantic, M.: 300 faces in-the-wild challenge: database and results. Image Vis. Comput. **47**, 3–18 (2016)
58. Shen, J., Zafeiriou, S., Chrysos, G.G., Kossaifi, J., Tzimiropoulos, G., Pantic, M.: The first facial landmark tracking in-the-wild challenge: benchmark and results. In: Proceedings of the IEEE International Conference on Computer Vision Workshops, pp. 50–58 (2015)
59. Zafeiriou, S., Trigeorgis, G., Chrysos, G., Deng, J., Shen, J.: The menpo facial landmark localisation challenge: a step towards the solution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 170–179 (2017)
60. Grishchenko, I., Ablavatski, A., Kartynnik, Y., Raveendran, K., Grundmann, M.: Attention mesh: high-fidelity face mesh prediction in real-time. arXiv preprint arXiv:2006.10962 (2020)
61. Lv, J., Shao, X., Xing, J., Cheng, C., Zhou, X.: A deep regression architecture with two-stage re-initialization for high performance facial landmark detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3317–3326 (2017)
62. Zhu, X., Lei, Z., Yan, J., Yi, D., Li, S.Z.: High-fidelity pose and expression normalization for face recognition in the wild. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 787–796 (2015)
63. Guo, J., Zhu, X., Yang, Y., Yang, F., Lei, Z., Li, S.Z.: Towards fast, accurate and stable 3d dense face alignment. In: European Conference on Computer Vision, pp. 152–168 (2020)
64. Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H.: Mobilenets: efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861 (2017)
65. Sagonas, C., Tzimiropoulos, G., Zafeiriou, S., Pantic, M.: 300 faces in-the-wild challenge: the first facial landmark localization challenge. In: Proceedings of the IEEE International Conference on Computer Vision Workshops, pp. 397–403 (2013)
66. Koestinger, M., Wohlhart, P., Roth, P.M., Bischof, H.: Annotated facial landmarks in the wild: a large-scale, real-world database for facial landmark localization. In: 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops), pp. 2144–2151 (2011)
67. Burgos-Artizzu, X.P., Perona, P., Dollár, P.: Robust face landmark estimation under occlusion. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1513–1520 (2013)
68. Wu, W., Qian, C., Yang, S., Wang, Q., Cai, Y., Zhou, Q.: Look at boundary: a boundary-aware face alignment algorithm. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2129–2138 (2018)
69. Zhu, X., Ramanan, D.: Face detection, pose estimation, and landmark localization in the wild. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 2879–2886 (2012)
70. Belhumeur, P.N., Jacobs, D.W., Kriegman, D.J., Kumar, N.: Localizing parts of faces using a consensus of exemplars. IEEE Trans. Pattern Anal. Mach. Intell. **35**(12), 2930–2940 (2013)
71. Le, V., Brandt, J., Lin, Z., Bourdev, L., Huang, T.S.: Interactive facial feature localization. In: European Conference on Computer Vision, pp. 679–692 (2012)
72. Ghiasi, G., Fowlkes, C.C.: Occlusion coherence: localizing occluded faces with a hierarchical deformable part model. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2385–2392 (2014)
73. Dong, X., Yan, Y., Ouyang, W., Yang, Y.: Style aggregated network for facial landmark detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 379–388 (2018)
74. Zou, X., Zhong, S., Yan, L., Zhao, X., Zhou, J., Wu, Y.: Learning robust facial landmark detection via hierarchical structured ensemble. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 141–150 (2019)

75. Wang, J., Sun, K., Cheng, T., Jiang, B., Deng, C., Zhao, Y., Liu, D., Mu, Y., Tan, M., Wang, X., Liu, W.: Deep high-resolution representation learning for visual recognition. IEEE Trans. Pattern Anal. Mach. Intell. **43**(10), 3349–3364 (2020)
76. Wang, X., Bo, L., Fuxin, L.: Adaptive wing loss for robust face alignment via heatmap regression. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 6971–6981 (2019)
77. Kumar, A., Marks, T.K., Mou, W., Wang, Y., Jones, M., Cherian, A., Koike-Akino, T., Liu, X., Feng, C.: Luvli face alignment: estimating landmarks' location, uncertainty, and visibility likelihood. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8236–8246 (2020)
78. Dapogny, A., Bailly, K., Cord, M.: Decafa: deep convolutional cascade for face alignment in the wild. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 6893–6901 (2019)
79. Lan, X., Hu, Q., Cheng, J.: Revisting quantization error in face alignment. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 1521–1530 (2021)
80. Huang, Y., Yang, H., Li, C., Kim, J., Wei, F.: Adnet: leveraging error-bias towards normal direction in face alignment. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 3080–3090 (2021)
81. Li, W., Lu, Y., Zheng, K., Liao, H., Lin, C., Luo, J., Cheng, C.T., Xiao, J., Lu, L., Kuo, C.F., Miao, S.: Structured landmark detection via topology-adapting deep graph learning. In: European Conference on Computer Vision, pp. 266–283 (2020)
82. Xia, J., Qu, W., Huang, W., Zhang, J., Wang, X., Xu, M.: Sparse local patch transformer for robust face alignment and landmarks inherent relation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4052–4061 (2022)
83. Trigeorgis, G., Snape, P., Nicolaou, M.A., Antonakos, E., Zafeiriou, S.: Mnemonic descent method: a recurrent process applied for end-to-end face alignment. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4177–4187 (2016)
84. Deng, J., Roussos, A., Chrysos, G., Ververas, E., Kotsia, I., Shen, J., Zafeiriou, S.: The menpo benchmark for multi-pose 2d and 3d facial landmark localisation and tracking. Int. J. Comput. Vis. **127**(6), 599–624 (2019)
85. Zhu, X., Lei, Z., Liu, X., Shi, H., Li, S.Z.: Face alignment across large poses: a 3d solution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 146–155 (2016)
86. Zhu, S., Li, C., Loy, C.C., Tang, X.: Unconstrained face alignment via cascaded compositional learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3409–3417 (2016)
87. Qian, S., Sun, K., Wu, W., Qian, C., Jia, J.: Aggregation via separation: boosting facial landmark detector with semi-supervised style translation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 10153–10163 (2019)
88. Xiong, X., De la Torre, F.: Supervised descent method and its applications to face alignment. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 532–539 (2013)
89. Cao, X., Wei, Y., Wen, F., Sun, J.: Face alignment by explicit shape regression. Int. J. Comput. Vis. **107**(2), 177–190 (2014)
90. Zheng, Q., Deng, J., Zhu, Z., Li, Y., Zafeiriou, S.: Decoupled multi-task learning with cyclical self-regulation for face parsing. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4156–4165 (2022)

91. Martyniuk, T., Kupyn, O., Kurlyak, Y., Krashenyi, I., Matas, J., Sharmanska, V.: DAD-3DHeads: a Large-scale dense, accurate and diverse dataset for 3D head alignment from a single image. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 20942–20952 (2022)

92. Shao, X., Xing, J., Lyu, J., Zhou, X., Shi, Y., Maybank, S.J.: Robust face alignment via deep progressive reinitialization and adaptive error-driven learning. IEEE Trans. Pattern Anal. Mach. Intell. (2021)