Pose-Invariant 3D Face Alignment

Amin Jourabloo, Xiaoming Liu
Department of Computer Science and Engineering
Michigan State University, East Lansing MI 48824
{jourablo, liuxm}@msu.edu

Abstract

Face alignment aims to estimate the locations of a set of landmarks for a given image. This problem has received much attention as evidenced by the recent advancement in both the methodology and performance. However, most of the existing works neither explicitly handle face images with arbitrary poses, nor perform large-scale experiments on non-frontal and profile face images. In order to address these limitations, this paper proposes a novel face alignment algorithm that estimates both 2D *and* 3D *landmarks and their* 2D *visibilities for a face image with an arbitrary pose. By integrating a* 3D *point distribution model, a cascaded coupled-regressor approach is designed to estimate both the camera projection matrix and the* 3D *landmarks. Furthermore, the* 3D *model also allows us to automatically estimate the* 2D *landmark visibilities via surface normal. We use a substantially larger collection of all-pose face images to evaluate our algorithm and demonstrate superior performances than the state-of-the-art methods.*

1. Introduction

This paper aims to advance *face alignment* in aligning face images with arbitrary *poses*. Face alignment is a process of applying a supervised learned model to a face image and estimating the locations of a set of facial landmarks, such as eye corners, mouth corners, etc [6]. Face alignment is a key module in the pipeline of most facial analysis algorithms, normally *after* face detection and *before* subsequent feature extraction and classification. Therefore, it is an enabling capability with a multitude of applications, such as face recognition [31], expression recognition [2], face de-identification [13], etc.

Given the importance of this problem, face alignment has been extensively studied since Dr. Cootes' Active Shape Model (ASM) in the 1990s [6]. Especially in recent years, face alignment has become one of the most published subjects in vision conferences [1, 21, 35, 36, 38, 43]. The existing approaches can be categorized into three types: Constrained Local Model (CLM)-based approach (e.g., [6, 26]), Active Appearance Model (AAM)-based approach (e.g., [16, 17, 22]) and regression-based approach (e.g., [4, 30]), and an excellent survey can be found in [33].

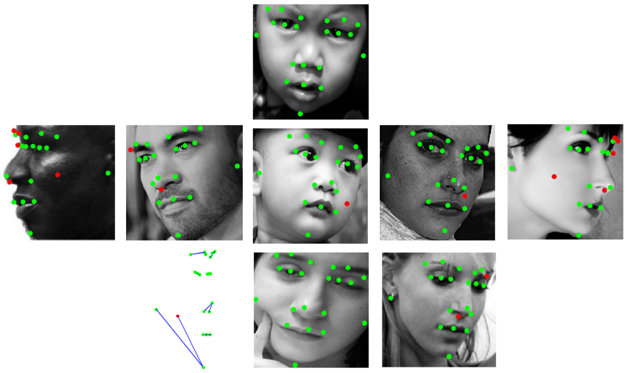

Figure 1: Given a face image with an arbitrary *pose*, our proposed algorithm automatically estimates the 2*D locations* and *visibilities* of facial landmarks, as well as 3*D landmarks*. The displayed 3D landmarks are estimated for the image in the center. Green/red points indicate visible/invisible landmarks.

Despite the continuous improvement on the alignment accuracy, face alignment is still a very challenging problem, due to the non-frontal face *pose*, low image *quality*, *occlusion*, etc. Among all the challenges, we identify the *pose invariant face alignment* as the one deserving substantial research efforts, for a number of reasons. First, face detection has substantially advanced its capability in detecting faces in all poses, including profiles [42], which calls for the subsequent face alignment to handle faces with arbitrary poses. Second, many facial analysis tasks would benefit from the robust alignment of faces at all poses, such as expression recognition and 3D face reconstruction [24]. Third, there are very few existing approaches that can align a face with any view angle, or have conducted extensive evaluations on face images across ±90° yaw angles [40, 48], which is a clear *contrast* with the vast face alignment literature [33].

Motivated by the needs to address the pose variation, and the lack of prior work in handling poses, as shown in Fig. 1,

Table 1: The comparison of face alignment algorithms in pose handling (estimation errors may have different definitions).

Method	3D landmark	Visibility	Pose-related database	Pose range	Training face #	Testing face #	Landmark #	Estimation errors
RCPR [3]	No	Yes	COFW	frontal w. occlu.	1,345	507	19	8.5
CoR [41]	No	Yes	COFW; LFPW-O; Helen-O	frontal w. occlu.	1,345; 468; 402	507; 112; 290	19; 49; 49	8.5
TSPM [48]	No	No	AFW	all poses	2,118	468	6	11.1
CDM [40]	No	No	AFW	all poses	1,300	468	6	9.1
OSRD [35]	No	No	MVFW	$< \pm 40^\circ$	2,050	450	68	N/A
TCDCN [46]	No	No	AFLW, AFW	$< \pm 60^\circ$	10,000	3,000; $\sim$313	5	8.0; 8.2
PIFA	Yes	Yes	AFLW, AFW	all poses	3,901	1,299; 468	21, 6	6.5; 8.6

this paper proposes a novel regression-based approach for *pose-invariant face alignment*, which aims to estimate the *2D and 3D locations* of face landmarks, as well as their *visibilities* in the 2D image, for a face with *arbitrary pose* (e.g., ±90° yaw). By extending the popular cascaded regressor for 2D landmark estimation, we learn two regressors for each cascade layer, one for predicting the update for the camera projection matrix, and the other for predicting the update for the 3D shape parameter. The learning of two regressors is conducted alternatively with the goal of minimizing the difference between the ground truth updates and the predicted updates. By using the 3D surface normals of 3D landmarks, we can automatically estimate the visibilities of their 2D projected landmarks by inspecting whether the transformed surface normal has a positive z coordinate, and these visibilities are dynamically incorporated into the regressor learning such that only the local appearance of visible landmarks contribute to the learning. Finally, extensive experiments are conducted on a large subset of AFLW dataset [15] with a wide range of poses, and the AFW dataset [48], with the comparison with a number of state-of-the-art methods. We demonstrate superior 2D alignment accuracy and quantitatively evaluate the 3D alignment accuracy.

In summary, the main contributions of this work are:

- To the best of our knowledge, this is the first face alignment that can estimate 2D/3D landmarks and their visibilities for a face image with an arbitrary pose.
- By integrating with a 3D point distribution model, a cascaded coupled-regressor approach is developed to estimate both the camera projection matrix and the 3D landmarks, where 3D model enables the automatically computed landmark visibilities via surface normal.
- A substantially larger number of non-frontal view face images are utilized in evaluation with demonstrated superior performances than the state of the art.

2. Prior Work

We now review the prior work in generic face alignment, pose-invariant face alignment, and 3D face alignment.

The first type of face alignment approach is based on Constrained Local Model (CLM), where an early example is ASM [6]. The basic idea is to learn a set of local appearance models, one for each landmark, and the decisions from the local models are fused with a global shape model. There are generative or discriminative [8] approaches in learning the local model, and various approaches in utilizing the shape constraint [1]. While the local models are favored for higher estimation precision, it also creates difficulty for alignment on low-resolution images due to limited local appearance. In contrast, the AAM method [5, 22] and its extension [20, 25] learn a global appearance model, whose similarity to the input image drives the landmark estimation. While AAM is known to have difficulty with unseen subjects [10], the recent development has substantially improved its generalization capability [29]. Motivated by the Shape Regression Machine [44, 47] in the medical domain, cascaded regressor-based methods have been very popular in recent years [4, 30]. On one hand, the series of regressors progressively reduce the alignment error and lead to a higher accuracy. On the other hand, advanced feature learning also renders ultra-efficient alignment procedures [14, 23]. Other than the three major types of algorithms, there are also works based on deep learning [46], graph-model [48], and semi-supervised learning [28].

Despite the explosion of methodology and efforts on face alignment, the literature on pose-invariant face alignment is rather limited, as shown in Tab. 1. There are four approaches explicitly handling faces with a wide range of poses. Zhu and Ranaman propose the TSPM approach for simultaneous face detection, pose estimation and face alignment [48]. An AFW dataset of in-the-wild faces with all poses is labeled with 6 landmarks and used for experiments. The cascaded deformable shape model (CDM) is a regression-based approach and probably the first approach claiming to be "pose-free" [40], therefore it is the most relevant work to ours. However, most of the experimental datasets contain near-frontal view faces, except the AFW dataset with improved performance than [48]. Also, there is no visibility estimation of the 2D landmarks. Zhang et al. develop an effective deep learning based method to estimate 5 landmarks. While accurate results are obtained, all

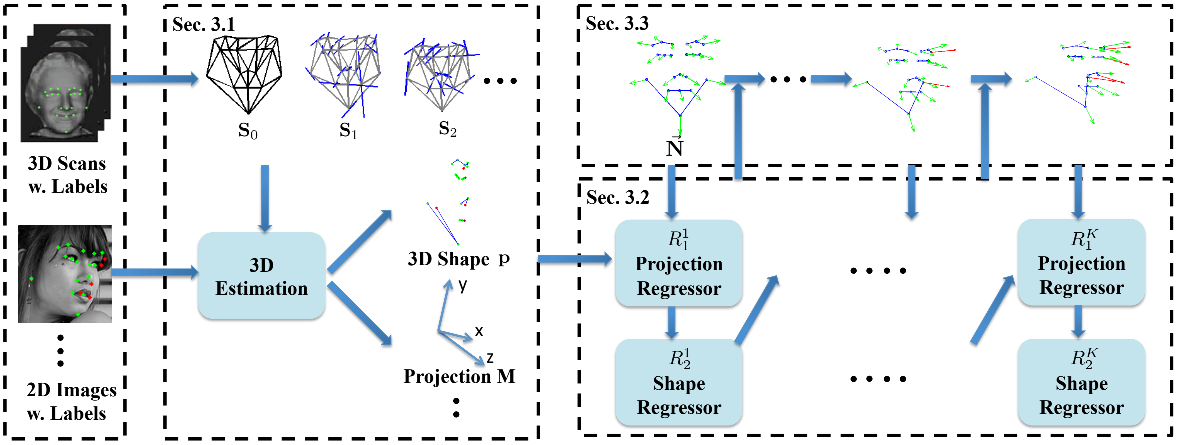

Figure 2: Overall architecture of our proposed PIFA method, with three main modules (3D modeling, cascaded coupled-regressor learning, and 3D surface-enabled visibility estimation). Green/red arrows indicate surface normals pointing toward/away from the camera.

testing images appear to be within ∼±60° so that all 5 landmarks are visible and there is no visibility estimation. The OSRD approach has the similar experimental constraint in that all images are within ±40° [35]. Other than these four works, the work on occlusion-invariant face alignment are also relevant since non-frontal faces can be considered as one type of occlusions, such as RCPR [3] and CoR [41]. Despite being able to estimate visibilities, neither method has been evaluated on faces with large pose variations. Finally, all aforementioned methods in this paragraph do not explicitly estimate the 3D locations of landmarks.

3D face alignment aims to recover the 3D locations of facial landmarks given a 2D image [11, 32]. There is also a very recently work on 3D face alignment from videos [12]. However, almost all methods take near-frontal-view face images as input, while our method can handle faces at all poses. A relevant but different problem is 3D face reconstruction, which recovers the *detailed 3D surface model* from one image, multiple images, or an image collection [9, 27]. Finally, 3D face model has been used in assisting 2D face alignment [34]. However, it has not been explicitly integrated into the powerful cascaded regressor framework, which is one of the main technical novelties of our approach.

3. Pose-Invariant 3D Face Alignment

This section presents the details of our proposed Pose-Invariant 3D Face Alignment (PIFA) algorithm, with emphasis on the training procedure. As shown in Fig. 2, we first learn a 3D Point Distribution Model (3DPDM) [7] from a set of labeled 3D scans, where a set of 2D landmarks on an image can be considered as a projection of a 3DPDM instance (i.e., 3D landmarks). For each 2D training face image, we assume that there exists the manual labeled 2D landmarks and their visibilities, as well as the corresponding *3D ground truth*– 3D landmarks and the camera projection matrix. Given the training images and 2D/3D ground truth, we train a cascaded coupled-regressor that is composed of two regressors at each cascade layer, for the estimation of the update of the 3DPDM coefficient and the projection matrix respectively. Finally, the visibilities of the projected 3D landmarks are automatically computed via the domain knowledge of the 3D surface normals, and incorporated into the regressor learning procedure.

3.1. 3D Face Modeling

Face alignment concerns the 2D face shape, represented by the locations of N 2D landmarks, i.e.,

$$\mathbf{U} = \begin{pmatrix} u_1 & u_2 & \cdots & u_N \\ v_1 & v_2 & \cdots & v_N \end{pmatrix}. \tag{1}$$

A 2D face shape $\mathbf{U}$ is a projection of a 3D face shape $\mathbf{S}$, similarly represented by the homogeneous coordinates of N 3D landmarks, i.e.,

$$\mathbf{S} = \begin{pmatrix} x_1 & x_2 & \cdots & x_N \\ y_1 & y_2 & \cdots & y_N \\ z_1 & z_2 & \cdots & z_N \\ 1 & 1 & \cdots & 1 \end{pmatrix}. \tag{2}$$

Similar to the prior work [34], a weak perspective model is assumed for the projection,

$$\mathbf{U} = \mathbf{M}\mathbf{S}, \tag{3}$$

where $\mathbf{M}$ is a 2×4 projection matrix with seven degrees of freedom (yaw, pitch, roll, two scales and 2D translations).

Following the basic idea of 3DPDM [7], we assume a 3D face shape is an instance of the 3DPDM,

$$\mathbf{S} = \mathbf{S}_0 + \sum_{i=1}^{N_s} p_i \mathbf{S}_i, \tag{4}$$

where $\mathbf{S}_0$ and $\mathbf{S}_i$ is the mean shape and ith shape basis of the 3DPDM respectively, N_s is the total number of shape bases, and p_i is the ith shape coefficient. Given a dataset of 3D scans with manual labels on N 3D landmarks per scan, we first perform procrustes analysis on the 3D scans to remove the global transformation, and then conduct Principal Component Analysis (PCA) to obtain the $\mathbf{S}_0$ and $\{\mathbf{S}_i\}$ (see the top-left part of Fig. 2).

The set of all shape coefficients $\mathbf{p} = (p_1, p_2, \cdots, p_{N_s})$ is termed as the *3D shape parameter* of an image. At this point, the face alignment for a testing image $\mathbf{I}$ has been converted from the estimation of $\mathbf{U}$ to the estimation of $\mathbf{P} = \{\mathbf{M}, \mathbf{p}\}$. The conversion is motivated by a few factors. First, without the 3D modeling, it is very difficult to model the out-of-plane rotation, which has a varying number of landmarks depending on the rotation angle and the individual 3D face shape. Second, as pointed out by [34], by only using $\frac{1}{6}$ of the number of the shape bases, 3DPDM can have an equivalent representation power as its 2D counterpart. Hence, using 3D model might lead to a more compact representation of unknown parameters.

Ground truth $\mathbf{P}$ Estimating $\mathbf{P}$ for a testing image implies the existence of ground truth $\mathbf{P}$ for each training image. However, while $\mathbf{U}$ can be manually labeled on a face image, $\mathbf{P}$ is normally unavailable unless a 3D scan is captured along with a face image. Therefore, in order to leverage the vast amount of existing 2D face alignment datasets, such as the AFLW dataset [15], it is desirable to estimate $\mathbf{P}$ for a face image and use it as the ground truth for learning.

Given a face image $\mathbf{I}$, we denote the manually labeled 2D landmarks as $\mathbf{U}$ and the landmark visibility as $\mathbf{v}$, an N-dim vector with binary elements indicating visible (1) or invisible (0) landmarks. Note that it is not necessary to label the 2D locations of invisible landmarks. We define the following objective function to estimate $\mathbf{M}$ and $\mathbf{p}$,

$$J(\mathbf{M}, \mathbf{p}) = \left|\left| \left(\mathbf{M} \left(\mathbf{S}_0 + \sum_{i=1}^{N_s} p_i \mathbf{S}_i \right) - \mathbf{U} \right) \odot \mathbf{V} \right|\right|^2, \quad (5)$$

where $\mathbf{V} = (\mathbf{v}^\intercal; \mathbf{v}^\intercal)$ is a $2 \times N$ visibility matrix, $\odot$ denotes the element-wise multiplication, and $||\cdot||^2$ is the sum of the squares of all matrix elements. Basically $J(\cdot,\cdot)$ computes the difference between the visible 2D landmarks and their 3D projections. An alternative estimation scheme is utilized, i.e., by assuming $\mathbf{p}^0 = 0$, we estimate $\mathbf{M}^k = \arg\min_{\mathbf{M}} J(\mathbf{M}, \mathbf{p}^{k-1})$, and then $\mathbf{p}^k = \arg\min_{\mathbf{p}} J(\mathbf{M}^k, \mathbf{p})$ iteratively until the changes of $\mathbf{M}$ and $\mathbf{p}$ are small enough. Both minimizations can be efficiently solved in closed forms via least-square error.

3.2. Cascaded Coupled-Regressor

For each training image $\mathbf{I}_i$, we now have its ground truth as $\mathbf{P}_i = \{\mathbf{M}_i, \mathbf{p}_i\}$, as well as their initialization, i.e., $\mathbf{M}_i^0 = g(\bar{\mathbf{M}}, \mathbf{b}_i)$, $\mathbf{p}_i^0 = \mathbf{0}$, and $\mathbf{v}_i^0 = \mathbf{1}$. Here $\bar{\mathbf{M}}$ is the average of ground truth projection matrices in the training set, $\mathbf{b}_i$ is a 4-dim vector indicating the bounding box location, and $g(\mathbf{M}, \mathbf{b})$ is a function that modifies the scale and translation of $\mathbf{M}$ based on $\mathbf{b}$. Given a dataset of N_d training images, the question is *how* to formulate an optimization problem to estimate $\mathbf{P}_i$. We decide to extend the successful cascaded regressors framework due to its accuracy and efficiency [4]. The general idea of cascaded regressors is to learn a series of regressors, where the kth regressor estimates the difference between the current parameter $\mathbf{P}_i^{k-1}$ and the ground truth $\mathbf{P}_i$, such that the estimated parameter gradually approximates the ground truth.

Motivated by this general idea, we adopt a cascaded coupled-regressor scheme where two regressors are learned at the kth cascade layer, for the estimation of $\mathbf{M}_i$ and $\mathbf{p}_i$ respectively. Specifically, the first learning task of the kth regressor is,

$$\Theta_1^k = \arg\min_{\Theta_1^k} \sum_{i=1}^{N_d} ||\Delta\mathbf{M}_i^k - R_1^k(\mathbf{I}_i, \mathbf{U}_i, \mathbf{v}_i^{k-1}; \Theta_1^k)||^2, \quad (6)$$

where

$$\mathbf{U}_i = \mathbf{M}_i^{k-1} \left(\mathbf{S}_0 + \sum_{i=1}^{N_s} p_i^{k-1} \mathbf{S}_i \right), \quad (7)$$

is the current estimated 2D landmarks, $\Delta\mathbf{M}_i^k = \mathbf{M}_i - \mathbf{M}_i^{k-1}$, and $R_1^k(\cdot; \Theta_1^k)$ is the desired regressor with the parameter of Θ_1^k. After Θ_1^k is estimated, we obtain $\Delta\hat{\mathbf{M}}_i = R_1^k(\cdot; \Theta_1^k)$ for all training images and update $\mathbf{M}_i^k = \mathbf{M}_i^{k-1} + \Delta\hat{\mathbf{M}}_i$. Note that this liner updating may potentially break the constraint of the projection matrix. Therefore, we estimate the scales and yaw, pitch, roll angles $(s_x, s_y, \alpha, \beta, \gamma)$ from $\mathbf{M}_i^k$ and compose a new $\mathbf{M}_i^k$ based on these five parameters.

Similarly the second learning task of the kth regressor is,

$$\Theta_2^k = \arg\min_{\Theta_2^k} \sum_{i=1}^{N_d} ||\Delta\mathbf{p}_i^k - R_2^k(\mathbf{I}_i, \mathbf{U}_i, \mathbf{v}_i^k; \Theta_2^k)||^2, \quad (8)$$

where $\mathbf{U}_i$ is computed via Eq 7 except $\mathbf{M}_i^{k-1}$ is replaced with $\mathbf{M}_i^k$. We also obtain $\Delta\hat{\mathbf{p}}_i = R_2^k(\cdot; \Theta_2^k)$ for all training images and update $\mathbf{p}_i^k = \mathbf{p}_i^{k-1} + \Delta\hat{\mathbf{p}}_i$. This iterative learning procedure continues for K cascade layers.

Learning $R^k(\cdot)$ Our cascaded coupled-regressor scheme does not depend on the particular feature representation or the type of regressors. Therefore, we may define them based on the prior work or any future development in features and regressors. Specifically, in this work we adopt the HOG-based linear regressor [37] and the fern regressor [3].

For the linear regressor, we denote a function $f(\mathbf{I}, \mathbf{U})$ to extract HOG features around a small rectangular region of each one of N landmarks, which returns a $32N$-dim feature vector. Thus, we define the regressor function as

$$R(\cdot) = \Theta^\intercal \cdot \mathrm{Diag}^*(\mathbf{v}_i) f(\mathbf{I}_i, \mathbf{U}_i), \quad (9)$$

where Diag$^*(\mathbf{v})$ is a function that duplicates each element of $\mathbf{v}$ 32 times and converts into a diagonal matrix of size $32N$. Note that we also add a constraint, $\lambda||\Theta||^2$, to Eq 6 or Eq 8 for a more robust least-square solution. By plugging Eq 9 to Eq 6 or Eq 8, the regressor parameter Θ (e.g., a $N_s \times 32N$ matrix for R_2^k) can be easily estimated in the closed form.

For the fern regressor, we follow the training procedure of [3]. That is, we divide the face region into a 3×3 grid. At each cascade layer, we choose 3 out of 9 zones with the least occlusion, computed based on the $\{\mathbf{v}_i^k\}$. For each selected zone, a depth 5 random fern regressor is learned from the interpolated shape-indexed features selected by the correlation-based method [4] from that zone only. Finally the learned $R(\cdot)$ is a weighted mean voting from the 3 fern regressors, where the weight is inversely proportional to the average amount of occlusion in that zone.

3.3. 3D Surface-Enabled Visibility

Up to now the only thing that has not been explained in the training procedure is how to estimate the visibility of the projected 2D landmarks, $\mathbf{v}_i$. It is obvious that during the testing we have to estimate $\mathbf{v}$ at each cascade layer for a testing image, since there is no visibility information given. As a result, during the training procedure, we also have to *estimate* $\mathbf{v}$ per cascade layer for each *training image*, rather than using the manually labeled ground truth visibility that is useful for estimating ground truth $\mathbf{P}$ as shown in Eq 5.

Depending on the camera projection matrix $\mathbf{M}$, the visibility of each projected 2D landmark may dynamically change along different layers of the cascade (see the top-right part of Fig. 2). In order to estimate $\mathbf{v}$, we decide to use the 3D face surface information. We start by assuming every individual has a similar 3D surface normal vector at each of its 3D landmarks. Then, by rotating the surface normal according to the rotation angle indicated by the projection matrix, we know that whether the rotated surface normal is pointing toward the camera (i.e., visible) or away from the camera (i.e., invisible). In other words, the sign of the z-axis coordinates indicates visibility.

By taking a set of 3D scans with manually labeled 3D landmarks, we can compute the landmarks' average 3D surface normals, denoted as a $3 \times N$ matrix $\vec{\mathbf{N}}$. Then we use the following equation to compute the visibility vector,

$$\mathbf{v} = \vec{\mathbf{N}}^\intercal \cdot \left(\frac{\mathbf{m}_1}{||\mathbf{m}_1||} \times \frac{\mathbf{m}_2}{||\mathbf{m}_2||} \right), \tag{10}$$

where $\mathbf{m}_1$ and $\mathbf{m}_2$ are the left-most three elements at the first and second row of $\mathbf{M}$ respectively, and $|| \cdot ||$ denotes the L_2 norm. For fern regressors, $\mathbf{v}$ is a soft visibility within ± 1. For linear regressors, we further compute $\mathbf{v} = \frac{1}{2}(1 + \text{sign}(\mathbf{v}))$, which results in a hard visibility of either 1 or 0.

In summary, we present the detailed training procedure in Algorithm 1.

Algorithm 1: The training procedure of PIFA.

```
Data: 3D model $\{\{\mathbf{S}\}_{i=0}^{N_s}, \vec{\mathbf{N}}\}$, labeled data $\{\mathbf{I}_i, \mathbf{U}_i, \mathbf{b}_i\}_{i=1}^{N_d}$
Result: Cascaded regressor parameters $\{\Theta_1^k, \Theta_2^k\}_{k=1}^{K}$
/* 3D modeling                                              */
foreach $i = 1, \cdots, N_d$ do
    Estimate $\mathbf{M}_i$ and $\mathbf{p}_i$ via Eq. 5;
/* Initialization                                           */
foreach $i = 1, \cdots, N_d$ do
    $\mathbf{p}_i^0 = \mathbf{0}$ ;            ▷ Assuming the mean 3D shape
    $\mathbf{v}_i^0 = \mathbf{1}$ ;          ▷ Assuming all landmarks visible
    $\mathbf{M}_i^0 = g(\bar{\mathbf{M}}, \mathbf{b}_i)$ and $\mathbf{U}_i = \mathbf{M}_i^0\mathbf{S}_0$ ;
/* Regressor learning                                       */
foreach $k = 1, \cdots, K$ do
    Estimate $\Theta_1^k$ via Eq 6 ;
    Update $\mathbf{M}_i^k$ and $\mathbf{U}_i$ for all images ;
    Compute $\mathbf{v}_i^k$ via Eq 10 for all images ;
    Estimate $\Theta_2^k$ via Eq 8 ;
    Update $\mathbf{p}_i^k$ and $\mathbf{U}_i$ for all images .
```

Model fitting Given a testing image $\mathbf{I}$ with bounding box $\mathbf{b}$ and its initial parameter $\mathbf{M}^0 = g(\bar{\mathbf{M}}, \mathbf{b})$ and $\mathbf{p}^0 = \mathbf{0}$, we can apply the learned cascaded coupled-regressor for face alignment. Basically we iteratively use $R_1^k(\cdot; \Theta_1^k)$ to compute $\Delta\hat{\mathbf{M}}$, update $\mathbf{M}^k$, compute $\mathbf{v}^k$, use $R_2^k(\cdot; \Theta_2^k)$ to compute $\Delta\hat{\mathbf{p}}$, and update $\mathbf{p}^k$. Finally the estimated 3D landmarks are $\hat{\mathbf{S}} = \mathbf{S}_0 + \sum_i p_i^K \mathbf{S}_i$, and the estimated 2D landmarks are $\hat{\mathbf{U}} = \mathbf{M}^K \hat{\mathbf{S}}$. Note that $\hat{\mathbf{S}}$ carries the individual 3D shape information of the subject, but not necessary in the same pose as the 2D testing image.

4. Experimental Results

Datasets The goal of this work is to advance the capability of face alignment on **in-the-wild faces with all possible view angles**, which is the type of images we desire when selecting experimental datasets. However, very few publicly available datasets satisfy this characteristic, or have been extensively evaluated in prior work (see Tab. 1). Nevertheless, we identify three datasets for our experiments.

AFLW dataset [15] contains $\sim$25,000 in-the-wild face images, each image annotated with the *visible* landmarks (up to 21 landmarks), and a bounding box. Based on our estimated $\mathbf{M}$ for each image, we select a subset of 5,200 images where the numbers of images whose absolute yaw angles within $[0°, 30°]$, $[30°, 60°]$, $[60°, 90°]$ are roughly $\frac{1}{3}$ each. To have a more *balanced distribution* of the left vs. right view faces, we take the odd indexed images among 5,200 (i.e., 1st, 3rd), flip them horizontally, and use them to replace the original images. Finally, a random partition leads to 3,901 and 1,299 images for training and testing respectively. As shown in Tab. 1, among the methods that test on all poses, we have the largest number of testing images.

AFW dataset [48] contains 205 images and in total 468 faces with different poses within ±90°. Each image is labeled with *visible* landmarks (up to 6), and a face bounding box. We only use AFW for testing.

Since we are also estimating 3D landmarks, it is important to test on a dataset with *ground truth*, rather than estimated, 3D landmark locations. We find BP4D-S database [45] to be the best for this purpose, which contains pairs of 2D images and 3D scans of spontaneous facial expressions from 41 subjects. Each pair has semi-automatically generated 83 2D and 83 3D landmarks, and the pose. We apply a random perturbation on 2D landmarks (to mimic imprecise face detection) and generate their enclosed bounding box. With the goal of selecting as many non-frontal view faces as possible, we choose a subset where the numbers of faces whose yaw angle within $[0°, 10°]$, $[10°, 20°]$, $[20°, 30°]$ are 100, 500, and 500 respectively. We randomly select half of $1,100$ images for training and the rest for testing, with disjoint subjects.

Experiment setup Our PIFA approach needs a 3D model of $\{\mathbf{S}\}_{i=0}^{N_s}$ and $\vec{\mathbf{N}}$. Using the BU-4DFE database [39] that contains 606 3D facial expression sequences from 101 subjects, we evenly sample 72 scans from each sequence and gather a total of 72×606 scans. Based on the method in Sec. 3.1, the resultant model has $N_s = 30$ for AFLW and AFW, and $N_s = 200$ for BP4D-S.

During the training and testing, for each image with a bounding box, we place the mean 2D landmarks (learned from the training set) on the image such that the landmarks on the boundary are within the four edges of the box. For training with linear regressors, we set $K = 10$, $\lambda = 120$, while $K = 75$ for fern regressors.

Evaluation metric Given the ground truth 2D landmarks $\mathbf{U}_i$, their visibility $\mathbf{v}_i$, and estimated landmarks $\hat{\mathbf{U}}_i$ of N_t testing images, we have two ways of computing the landmark estimation errors: 1) Mean Average Pixel Error (MAPE) [40], which is the average of the estimation errors for visible landmarks, i.e.,

$$\text{MAPE} = \frac{1}{\sum_i^{N_t} |\mathbf{v}_i|_1} \sum_{i,j}^{N_t,N} \mathbf{v}_i(j)||\hat{\mathbf{U}}_i(:,j) - \mathbf{U}_i(:,j)||, \tag{11}$$

where $|\mathbf{v}_i|_1$ is the number of visible landmarks of image $\mathbf{I}_i$, and $\mathbf{U}_i(:,j)$ is the jth column of $\mathbf{U}_i$. 2) Normalized Mean Error (NME), which is the average of the normalized estimation error of visible landmarks, i.e.,

$$\text{NME} = \frac{1}{N_t} \sum_i^{N_t} (\frac{1}{d_i |\mathbf{v}_i|_1} \sum_j^{N} \mathbf{v}_i(j)||\hat{\mathbf{U}}_i(:,j) - \mathbf{U}_i(:,j)||), \tag{12}$$

where d_i is the square root of the face bounding box size, as used by [40]. Note that normally d_i is the inter-eye distance in prior face alignment work dealing with near-frontal faces.

Table 2: The NME(%) of three methods on AFLW.

N_t	PIFA	CDM	RCPR
$1,299$	**6.52**		7.15
783	**6.08**	8.65	

Given the ground truth 3D landmarks $\mathbf{S}_i$ and estimated landmarks $\hat{\mathbf{S}}_i$, we first estimate the global rotation, translation and scale transformation so that the transformed $\mathbf{S}_i$, denoted as $\mathbf{S}'_i$, has the minimum distance to $\hat{\mathbf{S}}_i$. We then compute the MAPE via Eq 11 except replacing $\mathbf{U}$ and $\hat{\mathbf{U}}_i$ with $\mathbf{S}'_i$ and $\hat{\mathbf{S}}_i$, and $\mathbf{v}_i = \mathbf{1}$. Thus the MAPE only measures the error due to non-rigid shape deformation, rather than the pose estimation.

Choice of baseline methods Given the explosion of face alignment work in recent years, it is important to choose appropriate baseline methods so as to make sure the proposed method advances the state of the art. In this work, we select three recent works as baseline methods: 1) CDM [40] is a CLM-type method and the first one claimed to perform pose-free face alignment, which has exactly the same objective as ours. On AFW it also outperforms the other well-known TSPM method [48] that can handle all pose faces. 2) TCDCN [46] is a powerful deep learning-based method published in the most recent ECCV. Although it only estimates 5 landmarks for up to ~60° yaw, it represents the recent development in face alignment. 3) RCPR [3] is a regression-type method that represents the occlusion-invariant face alignment. Although it is an earlier work than CoR [41], we choose it due to its superior performance on the large COFW dataset (see Tab. 1 of [41]). It can be seen that these three baselines not only are most relevant to our focus on pose-invariant face alignment, but also well represent the major categories of existing face alignment algorithms based on [33].

Comparison on AFLW Since the source code of RCPR is publicly available, we are able to perform the training and testing of RCPR on our specific AFLW partition. We use the available executable of CDM to compute its performance on our test set. We strive to provide the same setup to the baselines as ours, such as the initial bounding box, regressor learning, etc. For our PIFA method, we use the fern regressor. Because CDM integrates face detection and pose-free face alignment, no bounding box was given to CDM and it successfully detects and aligns 783 out of $1,299$ testing images. Therefore, to compare with CDM, we evaluate the NME on the *same* 783 testing images. As shown in Tab. 2, our PIFA shows superior performance to both baselines. Although TCDCN also reports performance on a subset of $3,000$ AFLW images within ±60° yaw, it is evaluated with 5 landmarks, based on NME when d_i is the inter-eye distance. Hence, without the source code of TCDCN, it is difficult to have a fair comparison on our subset of AFLW images (e.g., we can not define d_i as the inter-

Table 3: The comparison of four methods on AFW.

N_t	N	Metric	PIFA	CDM	RCPR	TCDCN
468	6	MAPE	**8.61**	9.13		
313	5	NME	9.42		9.30	**8.20**

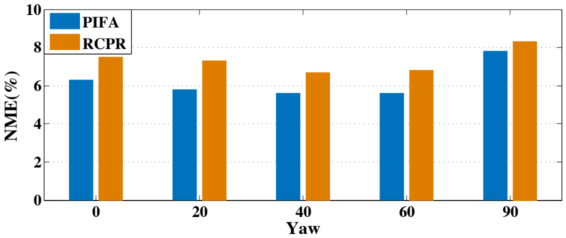

Figure 3: The NME of five pose groups for two methods.

eye distance due to profile view faces). On the $1,299$ testing images, we also test our method with linear regressors, and achieve a NME of 7.50, which shows the strength of fern regressors.

Comparison on AFW Unlike our specific subset of AFLW, the AFW dataset has been evaluated by all three baselines, but different metrics are used. Therefore, the results of the baselines in Tab. 3 are from the published papers, instead of executing the testing code. One note is that from the TCDCN paper [46], it appears that all 5 landmarks are visible on all displayed images and no visibility estimation is shown, which might suggest that TCDCN was evaluated on a subset of AFW with up to $\pm 60^\circ$ yaw. Hence, we select the total of 313 out of 468 faces within this pose range and test our algorithm. Since it is likely that our subset could differ to [46], please take this into consideration while comparing with TCDCN. Overall, our PIFA method still performs comparably among the four methods. This is especially encouraging given the fact that TCDCN utilizes a substantially larger training set of $10,000$ images - more than two times of our training set. Note that in addition to Tab. 2 and 3, our PIFA also has other benefits as shown in Tab. 1. E.g., we have 3D and visibility estimation, while RCPR has no 3D estimation and TCDCN does not have visibility estimation.

Estimation error across poses Just like pose-invariant face recognition studies the recognition rate across poses [18, 19], we also like to study the performance of face alignment across poses. As shown in Fig. 3, based on the estimated projection matrix $\mathbf{M}$ and its yaw angles, we partition all testing images of AFLW into five bins, each around a specific yaw angle. Then we compute the NME of testing images within each bin, for our method and RCPR. We can observe that the profile view images have in general larger NME than near-frontal images, which shows the challenge of pose-invariant face alignment. Further, the improvement of PIFA over RCPR is consistent across most of the poses.

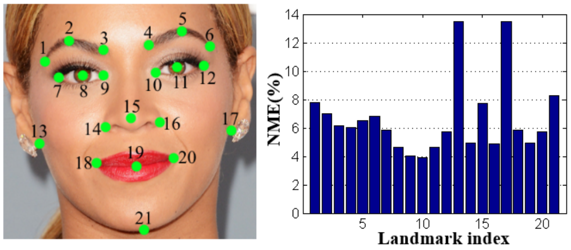

Figure 4: The NME of each landmark for PIFA.

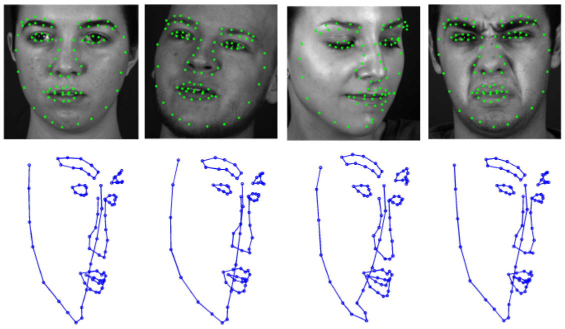

Figure 5: 2D and 3D alignment results of the BP4D-S dataset.

Table 4: Efficiency of four methods in FPS.

PIFA	CDM	RCPR	TCDCN
3.0	0.2	3.0	**58.8**

Estimation error across landmarks We are also interested in the estimation error across various landmarks, under a wide range of poses. Hence, for the AFLW test set, we compute the NME of each landmark for our method. As shown in Fig. 4, the two eye regions have the least amount of error. The two landmarks under the ears have the most error, which is consistent with the intuition. These observations also align well with prior face alignment study on near-frontal faces.

3D landmark estimation By performing the training and testing on the BP4D-S dataset, we can evaluate the MAPE of 3D landmark estimation, with exemplar results shown in Fig. 5. Since there are limited 3D alignment work and many of which do not perform quantitative evaluation, such as [11], we are not able to find another method as the baseline. Instead, we use the 3D mean shape, $\mathbf{S}_0$, as a baseline and compute its MAPE with respect to the ground truth 3D landmarks $\mathbf{S}_i$ (after global transformation). We find that the MAPE of $\mathbf{S}_0$ baseline is 5.02, while our method has 4.75. Although our method offers a better estimation than the mean shape, this shows that 3D face alignment is still a very challenging problem. We hope the effort to quantitatively measure the 3D estimation error, which is more difficult than its 2D counterpart, will encourage more research activities to address this challenge.

Computational efficiency Based on the efficiency reported in the publications of baseline methods, we compare the

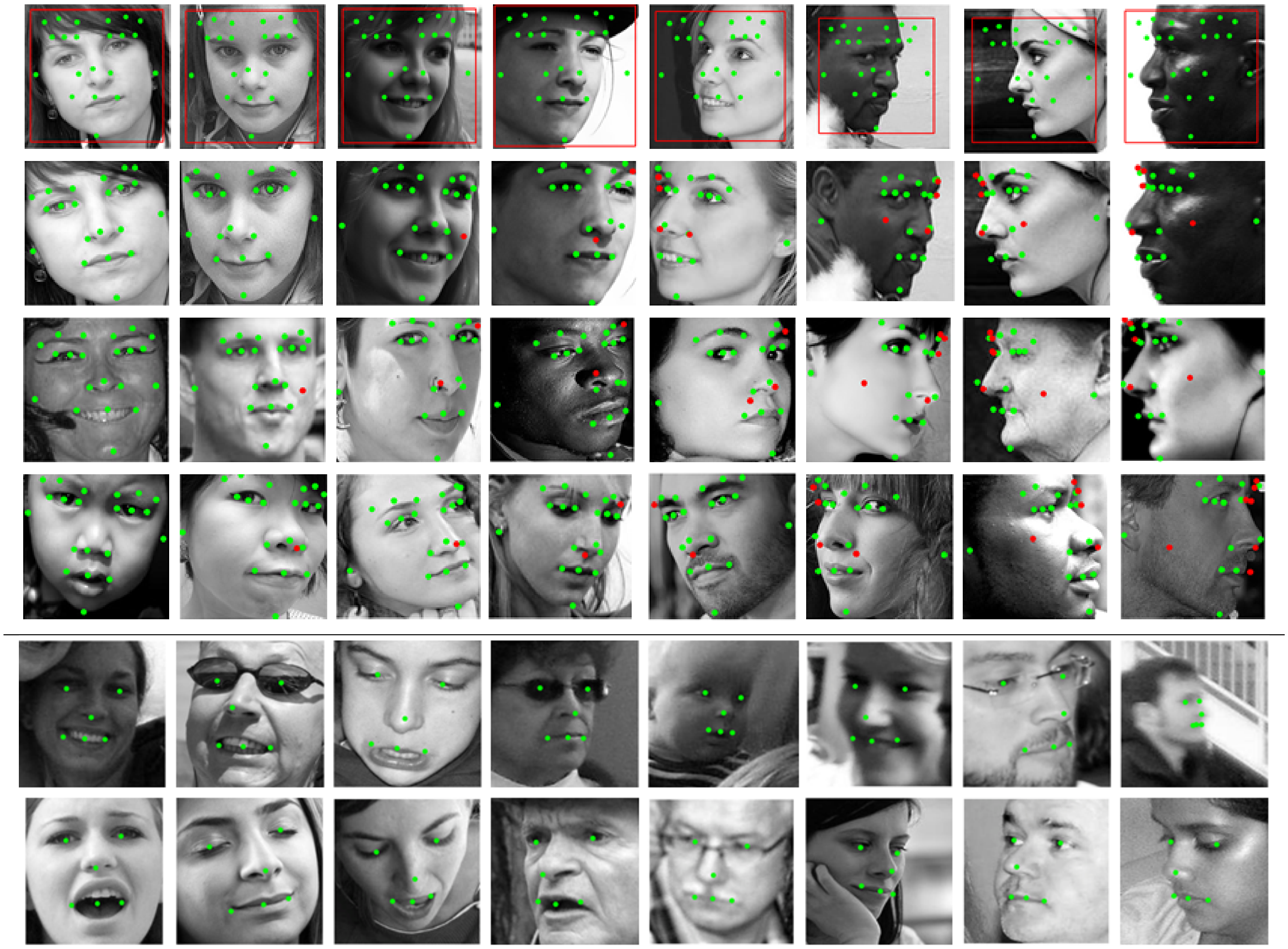

Figure 6: Testing results of AFLW (top) and AFW (bottom). As shown in the top row, we initialize face alignment by placing a 2D mean shape in the given bounding box of each image. Note the *disparity* between the initial landmarks and the final estimated ones, as well as the diversity in pose, illumination and resolution among the images. Green/red points indicate visible/invisible estimated landmarks.

computational efficiency of four methods in Tab. 4. Only TCDCN is measured based on the C implementation while other three are all based on Matlab implementation. It can be observed that TCDCN is the most efficient one. Consider that we estimate both 2D and 3D landmarks, at 3 FPS our unoptimized implementation is reasonably efficient. In our algorithm, the most computational demanding part is feature extraction, while estimating the updates for the projection matrix and 3D shape parameter has closed-form solutions and is very efficient.

Qualitative results We now show the qualitative face alignment results for images in two datasets. As shown in Fig. 6, despite the large pose range of ±90° yaw, our algorithm does a good job of aligning the landmarks, and correctly predict the landmark visibilities. These results are especially impressive if you consider the same mean shape (2D landmarks) is used as the initialization of all testing images, which has very large deformations with respect to their final landmark estimation.

5. Conclusions

Motivated by the fast progress of face alignment technologies and the need to align faces at all poses, this paper draws attention to a relatively less explored problem of face alignment robust to poses variation. To this end, we propose a novel approach to tightly integrate the powerful cascaded regressor scheme and the 3D face model. The 3D model not only serves as a compact constraint, but also offers an automatic and convenient way to estimate the visibilities of 2D landmarks - a key for successful pose-invariant face alignment. As a result, for a 2D image, our approach estimates the locations of 2D and 3D landmarks, as well as their 2D visibilities. We conduct an extensive experiment on a large collection of all-pose face images and compare with three state-of-the-art methods. While superior 2D landmark estimation has been shown, the performance on 3D landmark estimation indicates the future direction to improve this line of research.

References

[1] A. Asthana, S. Zafeiriou, S. Cheng, and M. Pantic. Robust discriminative response map fitting with constrained local models. In *CVPR*, pages 3444–3451. IEEE, 2013.

[2] V. Bettadapura. Face expression recognition and analysis: the state of the art. *arXiv preprint arXiv:1203.6722*, 2012.

[3] X. P. Burgos-Artizzu, P. Perona, and P. Dollár. Robust face landmark estimation under occlusion. In *ICCV*, pages 1513–1520. IEEE, 2013.

[4] X. Cao, Y. Wei, F. Wen, and J. Sun. Face alignment by explicit shape regression. *IJCV*, 107(2):177–190, 2014.

[5] T. Cootes, G. Edwards, and C. Taylor. Active appearance models. *IEEE T-PAMI*, 23(6):681–685, June 2001.

[6] T. Cootes, C. Taylor, and A. Lanitis. Active shape models: Evaluation of a multi-resolution method for improving image search. In *BMVC*, volume 1, pages 327–336, 1994.

[7] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models — their training and application. *CVIU*, 61(1):38–59, Jan 1995.

[8] D. Cristinacce and T. Cootes. Boosted regression active shape models. In *BMVC*, volume 2, pages 880–889, 2007.

[9] J. Gonzalez-Mora, F. De la Torre, N. Guil, and E. L. Zapata. Learning a generic 3D face model from 2D image databases using incremental structure-from-motion. *Image and Vision Computing*, 28(7):1117–1129, 2010.

[10] R. Gross, I. Matthews, and S. Baker. Generic vs. person specific active appearance models. *Image and Vision Computing*, 23(11):1080–1093, Nov. 2005.

[11] L. Gu and T. Kanade. 3D alignment of face in a single image. In *CVPR*, volume 1, pages 1305–1312, 2006.

[12] L. A. Jeni, J. F. Cohn, and T. Kanade. Dense 3D face alignment from 2D videos in real-time. In *FG*, 2015.

[13] A. Jourabloo, X. Yin, and X. Liu. Attribute preserved face de-identification. In *ICB*, 2015.

[14] V. Kazemi and J. Sullivan. One millisecond face alignment with an ensemble of regression trees. In *CVPR*, pages 1867–1874. IEEE, 2014.

[15] M. Koestinger, P. Wohlhart, P. M. Roth, and H. Bischof. Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. In *First IEEE International Workshop on Benchmarking Facial Image Analysis Technologies*, 2011.

[16] X. Liu. Discriminative face alignment. *IEEE T-PAMI*, 31(11):1941–1954, 2009.

[17] X. Liu. Video-based face model fitting using adaptive active appearance model. *Image and Vision Computing*, 28(7):1162–1172, 2010.

[18] X. Liu and T. Chen. Pose-robust face recognition using geometry assisted probabilistic modeling. In *CVPR*, volume 1, pages 502–509, 2005.

[19] X. Liu, J. Rittscher, and T. Chen. Optimal pose for face recognition. In *CVPR*, volume 2, pages 1439–1446, 2006.

[20] S. Lucey, R. Navarathna, A. B. Ashraf, and S. Sridharan. Fourier Lucas-Kanade algorithm. *IEEE T-PAMI*, 35(6):1383–1396, 2013.

[21] P. Luo, X. Wang, and X. Tang. Hierarchical face parsing via deep learning. In *CVPR*, pages 2480–2487. IEEE, 2012.

[22] I. Matthews and S. Baker. Active appearance models revisited. *IJCV*, 60(2):135–164, 2004.

[23] S. Ren, X. Cao, Y. Wei, and J. Sun. Face alignment at 3000 FPS via regressing local binary features. In *CVPR*, 2014.

[24] J. Roth, Y. Tong, and X. Liu. Unconstrained 3D face reconstruction. In *CVPR*, 2015.

[25] E. Sánchez-Lozano, F. De la Torre, and D. González-Jiménez. Continuous regression for non-rigid image alignment. In *ECCV*, pages 250–263. Springer, 2012.

[26] J. M. Saragih, S. Lucey, and J. Cohn. Face alignment through subspace constrained mean-shifts. In *ICCV*, 2009.

[27] G. Stylianou and A. Lanitis. Image based 3D face reconstruction: A survey. *Int. J. of Image and Graphics*, 9(2):217–250, 2009.

[28] Y. Tong, X. Liu, F. W. Wheeler, and P. Tu. Automatic facial landmark labeling with minimal supervision. In *CVPR*, 2009.

[29] G. Tzimiropoulos and M. Pantic. Optimization problems for fast AAM fitting in-the-wild. In *ICCV*, pages 593–600. IEEE, 2013.

[30] M. Valstar, B. Martinez, X. Binefa, and M. Pantic. Facial point detection using boosted regression and graph models. In *CVPR*, pages 2729–2736. IEEE, 2010.

[31] A. Wagner, J. Wright, A. Ganesh, Z. Zhou, H. Mobahi, and Y. Ma. Toward a practical face recognition system: Robust alignment and illumination by sparse representation. *IEEE T-PAMI*, 34(2):372–386, 2012.

[32] C. Wang, Y. Zeng, L. Simon, I. Kakadiaris, D. Samaras, and N. Paragios. Viewpoint invariant 3D landmark model inference from monocular 2D images using higher-order priors. In *ICCV*, pages 319–326. IEEE, 2011.

[33] N. Wang, X. Gao, D. Tao, and X. Li. Facial feature point detection: A comprehensive survey. *arXiv preprint arXiv:1410.1037*, 2014.

[34] J. Xiao, S. Baker, I. Matthews, and T. Kanade. Real-time combined 2D+3D active appearance models. In *CVPR*, volume 2, pages 535–542, 2004.

[35] J. Xing, Z. Niu, J. Huang, W. Hu, and S. Yan. Towards multi-view and partially-occluded face alignment. In *CVPR*, pages 1829–1836. IEEE, 2014.

[36] X. Xiong and F. De la Torre. Supervised descent method and its applications to face alignment. In *CVPR*, pages 532–539. IEEE, 2013.

[37] J. Yan, Z. Lei, D. Yi, and S. Z. Li. Learn to combine multiple hypotheses for accurate face alignment. In *ICCVW*, pages 392–396. IEEE, 2013.

[38] H. Yang and I. Patras. Sieving regression forest votes for facial feature detection in the wild. In *ICCV*, pages 1936–1943. IEEE, 2013.

[39] L. Yin, X. Chen, Y. Sun, T. Worm, and M. Reale. A high-resolution 3D dynamic facial expression database. In *FG*, 2008.

[40] X. Yu, J. Huang, S. Zhang, W. Yan, and D. N. Metaxas. Pose-free facial landmark fitting via optimized part mixtures and cascaded deformable shape model. In *ICCV*, pages 1944–1951. IEEE, 2013.

[41] X. Yu, Z. Lin, J. Brandt, and D. N. Metaxas. Consensus of regression for occlusion-robust facial feature localization. In *ECCV*, pages 105–118. Springer, 2014.

[42] C. Zhang and Z. Zhang. A survey of recent advances in face detection. Technical report, Tech. rep., Microsoft Research, 2010.

[43] J. Zhang, S. Shan, M. Kan, and X. Chen. Coarse-to-fine auto-encoder networks (CFAN) for real-time face alignment. In *ECCV*, pages 1–16. Springer, 2014.

[44] J. Zhang, S. Zhou, D. Comaniciu, and L. McMillan. Conditional density learning via regression with application to deformable shape segmentation. In *CVPR*, 2008.

[45] X. Zhang, L. Yin, J. F. Cohn, S. Canavan, M. Reale, A. Horowitz, P. Liu, and J. M. Girard. BP4D-spontaneous: a high-resolution spontaneous 3D dynamic facial expression database. *Image and Vision Computing*, 32(10):692 – 706, 2014.

[46] Z. Zhang, P. Luo, C. C. Loy, and X. Tang. Facial landmark detection by deep multi-task learning. In *ECCV*, pages 94–108. Springer, 2014.

[47] S. Zhou and D. Comaniciu. Shape regression machine. In *IPMI*, pages 13–25, 2007.

[48] X. Zhu and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *CVPR*, pages 2879–2886. IEEE, 2012.