ASMNet: a Lightweight Deep Neural Network for Face Alignment and Pose Estimation

Ali Pourramezan Fard, Hojjat Abdollahi, and Mohammad Mahoor
Department of Electrical and Computer Engineering
University of Denver, Denver, CO
{Ali.pourramezanfard, hojjat.abdollahi, mohammad.mahoor}@du.edu

Abstract

Active Shape Model (ASM) is a statistical model of object shapes that represents a target structure. ASM can guide machine learning algorithms to fit a set of points representing an object (e.g., face) onto an image. This paper presents a lightweight Convolutional Neural Network (CNN) architecture with a loss function being assisted by ASM for face alignment and estimating head pose in the wild. We use ASM to first guide the network towards learning a smoother distribution of the facial landmark points. Inspired by transfer learning, during the training process, we gradually harden the regression problem and guide the network towards learning the original landmark points distribution. We define multi-tasks in our loss function that are responsible for detecting facial landmark points as well as estimating the face pose. Learning multiple correlated tasks simultaneously builds synergy and improves the performance of individual tasks. We compare the performance of our proposed model called ASMNet with MobileNetV2 (which is about 2 times bigger than ASMNet) in both the face alignment and pose estimation tasks. Experimental results on challenging datasets show that by using the proposed ASM assisted loss function, the ASMNet performance is comparable with MobileNetV2 in the face alignment task. In addition, for face pose estimation, ASMNet performs much better than MobileNetV2. ASMNet achieves an acceptable performance for facial landmark points detection and pose estimation while having a significantly smaller number of parameters and floating-point operations compared to many CNN-based models.

1. Introduction

Facial Landmark Points Detection is an essential task in many facial image analyses and applications. It is crucial for facial image alignment, face recognition [18, 24, 35], pose estimation [41], and facial expression recognition [37, 56]. Active Shape Model (ASM) introduced by Tim Cootes [9] is among the first methods designed for facial landmark points detection. ASM is a statistical shape model made out of object samples. ASM and its variant, Active Appearance Models (AAM) [8, 25], can guide learning algorithms to fit a set of points (e.g., facial points) representing an object into an image containing the object instance. In better words, ASM guides the learning algorithm to iteratively deforms the model to find the best match po-

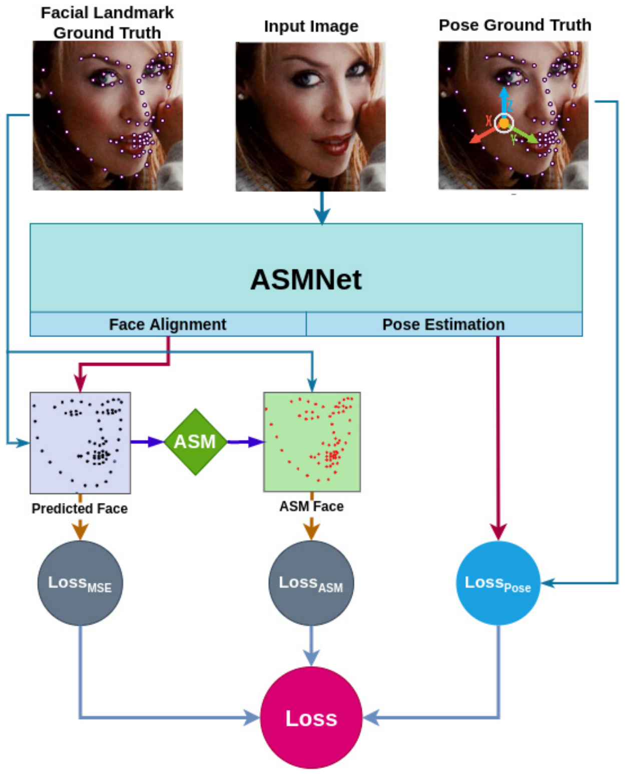
Facial Landmark Ground Truth
Input Image
Pose Ground Truth
ASMNet
Face Alignment
Pose Estimation
ASM
Predicted Face
ASM Face
Loss_{MSE}
Loss_{ASM}
$\text{Loss}_{\text{Pose}}$
Loss

Figure 1: The proposed loss function ($Loss_{total}$) learns two main tasks simultaneously and uses ASM as assistant loss.

sition between the model and the data in a new image. ASM/AAM and their predecessors' deformable models [34] have been studied well for landmark point detection in facial image analysis and human body joint tracking. We propose to use ASM in a deep convolutional neural network (CNN) for facial landmark point detection and head pose estimation.

Although most of the existing computer vision methods have focused on facial landmark points detection and pose estimation as separate tasks, some recent works [6, 28, 55, 56] show that learning correlated tasks simultaneously can improve the system accuracy. For instance, the authors of [52] explain that information contained in the features is distributed throughout deep neural networks hierarchically. More specifically, while lower layers contain information about edges, and corners and hence are more appropriate for localization tasks such as facial landmark points detection and pose estimation, deeper layers contain more abstract information which is more suitable for classification tasks [28]. Inspired by the idea of multi-task learning, we design our CNN model and hence the associated loss function to learn multiple correlated tasks simultaneously.

Several methods have been proposed for facial landmark points detection such as Constrained Local Model-Based Methods [3, 11], AAM [8, 25], part models [60], and Deep Learning (DL) based methods [53, 54]. Although DL-based methods are considered as state-of-the-art methods, facial landmark points detection is still considered a challenging task specifically for faces with large pose variations [12, 22, 44]. Accordingly, the price to pay to achieve a high accuracy is the rise in computational complexity and the fall in efficiency. Recent methods have focused on improving the accuracy and this is normally achieved by introducing new layers and consequently increasing the number of parameters as well as longer inference time. These methods prove to be accurate and successful in desktop and server applications, but with the growth of IoT, mobile devices, and robotics, there is a growing need for more accurate and efficient algorithms. There are a few networks that are designed for low-power devices. One of the most popular ones is MobileNetV2 [33] which has proven to be a good feature extractor [17].

In this paper, we propose a new network structure that is inspired by MobileNetV2 [33] and is specifically designed for facial landmark points detection with the focus on making the network shallow and small without losing much accuracy. To achieve this goal we propose a new loss function that employs ASM as an assistant loss and uses multi-task learning to improve the accuracy. Fig. 1 depicts a general framework of our proposed idea. We tested our proposed method with the challenging 300W [31] dataset, and the Wider Facial Landmarks in the Wild (WFLW) [44] dataset. Our experimental results show that the accuracy of facial landmark points detection and pose estimation is comparable with the state-of-the-art methods while the size of the network is 2 times smaller than MobileNetV2 [33].

The remainder of this paper is organized as follows. Sec. 2 reviews the related work in facial landmark points detection, pose detection, and small neural networks. Sec. 3 describes the architecture of our proposed network, the ASM assisted loss function and the training method. Experimental results are provided in Sec. 4. Finally, Sec. 5 concludes the paper with some discussions on the proposed method and future research directions.

2. Related Work

Automated facial landmark points detection has been studied extensively by the computer vision community. Zhou *etal.* [57] classified facial landmark points detection methods into two categories: regression-based and template fitting methods. Regression-based methods consider a facial image as a vector and use a transformation such as Principle Component Analysis (PCA), Discrete Cosine Transform (DCT) [32], or Gabor Wavelets [2, 43] to transform the image into another domain. Then a classification algorithm such as SVM [1, 14] or boosted cascade detector [42] is used to detect facial landmarks. In contrast, template fitting methods such as Active Shape Models (ASM) [9, 27] and Active Appearance Models (AAM) [10] constrain the search for landmark positions by using prior knowledge. Inspired by the ASM method, we define a loss term that applies a constraint to the shapes learned during the training process.

Recently, Deep Learning techniques have dominated state-of-the-art results in terms of performance and robustness. There have been several new CNN-based methods for facial landmark points detection. Sun Y. *etal.* [36] proposed a deep CNN cascade to extract the facial key-points back in 2013. Zhang Z. *etal.* [54] proposed a multi-task approach in which instead of solving FLP detection as an isolated task, they bundle it with similar tasks such as head pose estimation into a deep neural network to increase robustness and accuracy. Ranjan R. *etal.* [28] also uses deep multi-task learning and achieves high accuracy when detecting facial landmark. Several studies fit a 3D model onto the face and then infer the landmark positions [19, 20, 59].

One related task that can be trained simultaneously with facial landmark points detection, is head pose estimation. Detecting facial landmarks and estimating head pose can be made easier using 3D information [39, 59], however, this information is not always available. Wu *etal.* [46] propose a unified model for simultaneous facial landmark points detection, head pose estimation, and facial deformation analysis. This approach is robust to occlusion which is the result of the interaction between these tasks. One approach to estimate the head pose is to use the facial landmark point and head pose estimator sequentially [41]. We proposed a multi-task learning approach to tackle the problem of facial landmark points detection and pose estimation using a loss function assisted by ASM.

3. Proposed ASM Network

We first review the Active Shape Model (ASM) algorithm and then introduce our proposed network architecture for landmark point localization and pose estimation. Finally, we explain our customized loss function based on ASM that improves the accuracy of the network.

3.1. Active Shape Model Review

Active Shape Model is a statistical model of shape objects. Each shape is represented as n points, $S_{set} = \{(S_x^1, S_y^1), ..., (S_x^n, S_y^n)\}$ that are aligned into a common coordinate system. To simplify the problem and learn shape components, Principal Component Analysis (PCA) is applied to the covariance matrix calculated from a set of K training shape samples. Once the model is built, an approximation of any training sample (S) is calculated using Eq. 1:

$$S \approx \overline{S} + Pb \tag{1}$$

where $\overline{S}$ is the sample mean, $P = (p_1, p_2, ..., p_t)$ contains t eigenvectors of the covariance matrix and b is a t dimensional vector given by Eq. 2:

$$b = P^{\mathsf{T}}(S - \overline{S}) \tag{2}$$

Consequently, a set of parameters of a deformable model is defined by vector b, so that by varying the elements of the vector, the shape of the model is changed. Consider that the statistical variance (*i.e.*, eigenvalue) of the i^{th} parameter of b is λ_i. To make sure the generated image after applying ASM is relatively similar to the ground truth, the parameter b_i of vector b is usually limited to $\pm 3\sqrt{\lambda_i}$ [7]. This constraint ensures that the generated shape is similar to those in the original training set. Hence, we create a new shape S_{New} after applying this constraint, according to Eq. 3:

$$S_{New} = \overline{S} + P\tilde{b} \tag{3}$$

where $\tilde{b}$ is the constrained b. We also define $\mathcal{ASM}$ operator according to Eq. 4:

$$\mathcal{ASM} : (P_x^i, P_y^i) \mapsto (A_x^i, A_y^i) \tag{4}$$

$\mathcal{ASM}$ transforms each input point (P_x^i, P_y^i) to a new point (A_x^i, A_y^i) using Eqs. 1, 2, and 3.

In this paper, we propose a deep convolutional neural network architecture that utilizes ASM in the training loss function. Our proposed network (ASMNet) is significantly smaller than its predecessor, MobileNetV2 [33], while its performance is comparable with MobileNetV2 [33] in localizing landmark points and much better in pose estimation.

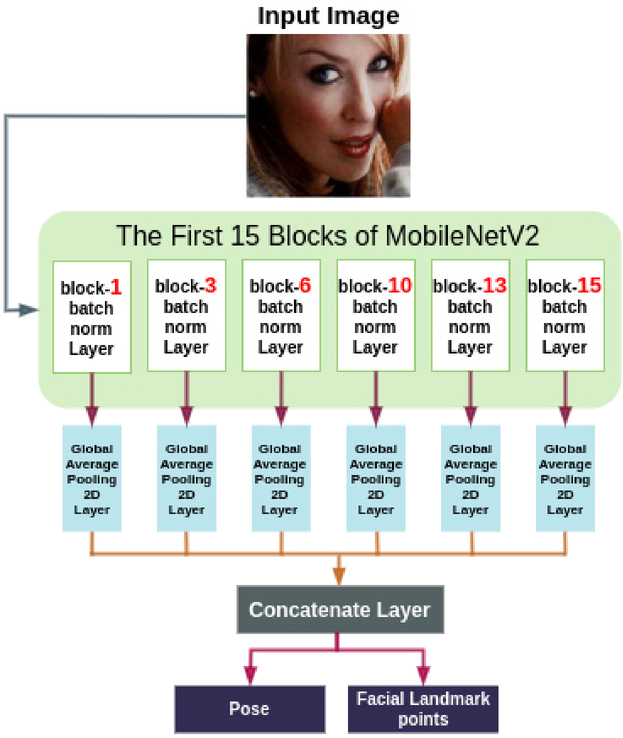
Input Image
The First 15 Blocks of MobileNetV2
block-1 batch norm Layer
block-3 batch norm Layer
block-6 batch norm Layer
block-10 batch norm Layer
block-13 batch norm Layer
block-15 batch norm Layer
Global Average Pooling 2D Layer
Global Average Pooling 2D Layer
Global Average Pooling 2D Layer
Global Average Pooling 2D Layer
Global Average Pooling 2D Layer
Global Average Pooling 2D Layer
Concatenate Layer
Pose
Facial Landmark points

Figure 2: The architecture of the ASMNet network. ASMNet designed using first 15 blocks of MobileNetV2 [33].

3.2. Proposed ASMNet Architecture

MobileNet and its variants [33] have received great attention as one of the most well-known deep neural networks for operation on embedded and mobile devices. Especially, MobileNetV2 [33] is shown to cope well with complex tasks such as image classification, object detection, and semantic segmentation. We have designed a network that is about two times smaller than MobileNetV2 [33], both in terms of the number of parameters, and FLOPs. In designing ASMNet, we only use the first 15 blocks of MobileNetV2 [33] while the main architecture which has 16 blocks. Nevertheless, creating a shallow network would eventually lower the final accuracy of the system. To avoid this problem we purposefully add a few new layers. Fig. 2 shows the architecture of ASMNet.

According to [52] the features in a Convolutional Neural Network (CNN) are distributed hierarchically. In other words, lower layers have features such as edges, and corners which are more suitable for tasks like landmark localization and pose estimation, and deeper layers contain more abstract features that are more suitable for tasks like image classification and image detection. Training a network for correlated tasks simultaneously builds a synergy that can improve the performance of each task [6, 54].

Motivated by the approach in [28], we designed a multi-task CNN to detect facial landmarks as well as estimating the pose of the faces (pitch, roll, and yaw) simultaneously. In order to use features from different layers, we have created shortcuts from *block-1-batch-normalization*, *block-3-batch-normalization*, *block-6-batch-normalization*, *block-10-batch-normalization*, and finally *block-13-batch-normalization*. We connect each of these shortcuts to the output of block 15 of MobileNetV2 [33], *block-15-add*, using a *global average pooling* layer. Finally, we concatenate all the *global average pooling* layers. Such architecture enables us to use features that are available in different layers of the network while keeping the number of the FLOPs small. In other words, since the original MobileNetV2 [33] is designed for image classification task - where the more abstract features are required - it might not be suitable for face alignment task - which needs both abstract features that are available in the deeper layers as well as features that are available in the lower layers such as edges and corners -.

We Designed ASMNet (see Fig. 2), by fusing the features that are available if different layers of the model. Furthermore, by concatenating the features that are collected after each *global average pooling* layer in the back-propagation process, it will be possible for the network to evaluate the effect of each shortcut path.

Moreover, we add another correlated task to the network. As Fig. 2 shows, the proposed network predicts 2 different outputs: the facial landmark points (the main output of the network), as well as the face pose. While the correlation and the synergy between these two tasks can result in more accurate results, we also wanted our lightweight ASMNet to be able to predict face pose as well so that it might be used in more applications.

3.3. ASM Assisted Loss Function

In the following, we describe the loss functions for two different tasks. These tasks are responsible for *facial landmark points detection*, and *pose estimation*.

Facial landmark points detection task: The common loss function for facial landmark points detection is Mean Square Error (MSE). We propose a new loss function that including MSE, as the main loss as well an the *assistant* loss which utilizes ASM to improve the accuracy of the network called *ASM-LOSS*.

The proposed ASM-LOSS guides the network to first learn the smoothed distribution of the facial landmark points. In other words, during the training process, the loss function compares the predicted facial landmark points with their corresponding ground truth as well as the smoothed version the ground truth which is generated using ASM. Given this, in the early stage of training, we set a bigger weight to the ASM-LOSS in comparison to the main loss – which is MSE –, since the variation of the smoothed facial landmark points are much lower than the original landmark points, and as a rule of thumb, easier to be learned by a CNN. Then, by gradually decrease the weight of the ASM-LOSS, we lead the network to focus more on the original landmark points. In practice, we figured out that this method, which is also can be taken to account as transfer learning, works out well and results in more accurate models.

We also discover that although face pose estimation has a heavy reliance on face alignment, it can achieve good accuracy with the assistant of smoothed facial landmark points as well. In other words, if the performance of facial landmark point detection task is *acceptable*, which means network can predict facial landmark such that the whole shape of the face is correct, the pose estimation can achieve a good accuracy. Accordingly, using smoothed landmark points and training network using ASM-LOSS will results in a more accuracy in pose estimation task.

Consider that for each image in the training set, there exists *n* landmark points in a set called *G* such that (G^i_x, G^i_y) is the coordinates for the i^{th} landmark point. Similarly, a predicted set *P* contains *n* points such that (P^i_x, P^i_y) is the predicted coordinates for the i^{th} landmark point.

$$\begin{aligned} G_{set} &= \{(G^1_x, G^1_y), ..., (G^n_x, G^n_y)\} \\ P_{set} &= \{(P^1_x, P^1_y), ..., (P^n_x, P^n_y)\} \end{aligned} \tag{5}$$

We apply PCA on the training set and calculate *eigenvectors* and *eigenvalues*. Then, we calculate set *A*, which contains *n* points and each point is the transformation of the corresponding point in *G*, by applying the $\mathcal{ASM}$ operator according to Eq. 4:

$$\begin{aligned} A_{set} &= \{(A^1_x, A^1_y), ..., (A^n_x, A^n_y)\} \\ \mathcal{ASM} &: (G^i_x, G^i_y) \mapsto (A^i_x, A^i_y) \end{aligned} \tag{6}$$

We define the *main* facial landmark point loss, Eq. 7, as the Mean Square Error between the ground truth (*G*) and the predicted landmark points (*P*).

$$\mathcal{L}_{mse} = \frac{1}{N}\frac{1}{n}\sum_{j=1}^{N}\sum_{i=1}^{n}\|G^i_j - P^i_j\|_2 \tag{7}$$

where N is the total number of images in the training set and $G^i_j = (G^i_x, G^i_y)$ shows the i^{th} landmark of the j^{th} sample in the training set. We calculate ASM-LOSS as the error between ASM points (A_{set}), and predicted landmark points (P_{set}) using Eq. 8:

$$\mathcal{L}_{asm} = \frac{1}{N}\frac{1}{n}\sum_{j=1}^{N}\sum_{i=1}^{n}\|A^i_j - P^i_j\|_2 \tag{8}$$

Finally, we calculate the *total* loss for the *facial landmark task* with according to Eq. 9:

$$\mathcal{L}_{facial} = \mathcal{L}_{mse} + \alpha \ \times \ \mathcal{L}_{asm} \tag{9}$$

The accuracy of PCA have a heavy reliance on the ASM points (A_{set}), which means that the more accurate the PCA, the less the discrepancy between the ground truth (G) and the ASM points (A_{set}). To be more detailed, by reducing the accuracy of PCA, the generated ASM points (A_{set}), will be more similar to the *average point set*, which is the average of all the ground truth face objects in the training sets. Consequently, predicting points in A_{set} is easier than the points in the G_{set} since the variation of latter is lower than the variation of the former. We use this feature to design our loss function such that we first guide the network towards learning the distribution of the smoothed landmark points – which is easier to be learned – and gradually harden the problem by decreasing the weight of ASM-LOSS.

We define α as ASM-LOSS weight using Eq. 10:

$$\alpha = \begin{cases} 2 & i < \frac{l}{3} \\ 1 & \frac{l}{3} < i < \frac{2l}{3} \\ 0.5 & i > \frac{2l}{3} \end{cases} \tag{10}$$

where i is the epoch number and l is the total number of training epochs. As shown in Eqs. 9, at the beginning of the training, the value of α is higher, which means we put more emphasize on ASM-LOSS. Hence, the network focuses more on predicting a simpler task and converges faster. Then after one-third of total epochs, we reduce α to 1, and put equal emphasis on the main MSE loss ASM-LOSS. Finally, after two-third of total epochs, by reducing α to 0.5, we direct the network toward predicting the main ground truths, while considering the smoothed points generated using ASM as an assistant. We also show experimentally in Sec. 4, that such technique leads to more accurate results, specifically when it comes to a lightweight network like ASMNet.

Pose estimation task: We use mean square error to calculate the loss for the head pose estimation task. Eq. 11 defines the loss function $\mathcal{L}_{pose}$, where yaw(y^p), pitch(p^p), and roll(r^p) are the predicted poses and y^t, p^t, and r^t are the corresponding ground truths.

$$\mathcal{L}_{pose} = \frac{1}{N}\sum_{j=1}^{N}\frac{(y_j^p - y_j^t)^2 + (p_j^p - p_j^t)^2 + (r_j^p - r_j^t)^2}{3} \tag{11}$$

Finally, we calculate the total loss as the total weighted loss of the 2 individual losses using Eq. 12:

$$\mathcal{L} = \sum_{i=1}^{2}\lambda_{task_i}\mathcal{L}_{task_i} \tag{12}$$

such that $task_i$ is the i^{th} element of the task set T = { $\mathcal{L}_{facial}$, $\mathcal{L}_{pose}$ } and the value of λ_{task_i} corresponds to the importance of the i^{th} task. Since we define facial landmark points detection task to be more important than pose estimation, we choose $\lambda_{task} = \{1, 0.5\}$. Fig. 1 illustrates the process of calculating the total loss value.

4. Experimental Results

4.1. Training Phase

300W.We followed the protocol described in [29] to train our networks on the 300W [31] dataset. We use 3,148 faces consisting of 2,000 images from the training subset of HELEN [23] dataset, 811 images from the training subset of LFPW [4] dataset, and 337 images from the full set of AFW [60] dataset with a 68-point annotation. For testing, 300W [31] has 3 subsets: Common subset with 554 images, Challenging subset with 135 images, and Full subset, including both Common and Challenging subsets, with 689 images. More specifically, the Challenging subset is the IBUG [31] dataset while the Common subset is a combination of the HELEN test subset (330 images) and LFPW test subset (224 images).

WFLW. WFLW [44], containing 7500 images for training and 2500 images for testing, is another widely used dataset, recently has been proposed based on WIDER FACE [51]. Each image in this dataset contains 98 manual annotated landmarks. In order to be able to evaluate the models under different circumstances, WFLW [44] provides 6 different subsets including 314 expression images, 326 large pose images, 206 make-up images, 736 occlusion images, 698 illumination images, and 773 blur images.

We use the method and algorithm in [30] to calculate the yaw, roll, and, pitch for each image in the dataset since to the best of our knowledge, no dataset provides the annotation for facial landmark points and face pose jointly.

4.2. Implementation Details

For the training set in each dataset, we crop all the images and extract the face region. Then the face images are scaled to 224 × 224 pixels. We augment the images (in terms of contrast, brightness, and color) to add robustness of data variation to the network. We use Adam optimizer for training the networks with learning rate 10^{-2}, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $decay = 10^{-5}$. Then we train networks for about 150 epochs with a batch size of 50. We implemented our codes using the TensorFlow library and run them on a NVidia 1080Ti GPU.

4.3. Evaluation Metrics

We follow the previous works and employ normalized mean error (NME) to measure the accuracy of our model. We define the normalising factor, followed by MDM [38] and [31] as "inter-ocular" distance (the distance between the outer-eye-corners). Furthermore, we calculate failure rate (FR), defined as the proportion of failed detected faces, for a maximum error of 0.1. Cumulative Errors Distribution (CED) curve as well as the area-under-the-curve (AUC) [49] is also reported. Besides, we use mean absolute error (MAE) for evaluating the pose estimation task.

4.4. Comparison with Other Models

We conducted four different experiments to evaluate the effectiveness of the proposed ASM assisted loss function. These experiments are designed to assess the performance of MobileNetV2 [33] and ASMNet, with and without the proposed ASM assisted loss function.

Table 1 shows the results of the experiments on *Full* subsets of 300W [31] (full), and WFLW [44], as well as the number of network parameters(#Params) and the sum of the FLOPs. For simplicity, we name our model as ”mnv2” (MobileNetV2 [33] trained using standard MSE loss function), ”mnv2_r” (MobileNetV2 [33] trained using our ASM assisted loss function), ”ASMNet_nr” (ASMNet trained using standard MSE loss function), and ”ASMNet” (ASMNet trained using our ASM assisted loss function).

Table 1 shows that the proposed ASM assisted loss function has a lower NME in both cases. Furthermore, while our proposed network architecture is about two times smaller than MobileNetV2 [33], its performance is comparable with it after applying our proposed ASM assisted loss function. It means that without sacrificing accuracy, we have created a network that is smaller and faster in comparison to MobileNetV2 [33]. Such characteristics make the ASMNet suitable for running on mobile and embedded devices.

Evaluation on 300W. The 300W [31] dataset is a very challenging benchmark in facial landmark detection task. Table 2 shows a comparison between ASMNet and the state-of-the-art methods. Although the performance of ASMNet does not outperform the state-of-the-art methods, comparing the number of the parameters, and FLOPs of the models (see Table 6), the accuracy of our proposed model is comparable and accurate in the context of small networks such as MobileNetV2 [33]. Furthermore, As the table 2 shows, the performance of the ASMNet with ASM assisted loss function on 300W [31] is better than the performance of ASMNet without the assisted loss. Fig. 3 shows the some example of facial landmark detection using ASMNet on the Challenging subset of 300W [31] dataset. As we can see, ASMNet performs well, even in challenging face images.

Table 1: Number of parameters in million (M) and FLOPs in billion (B), as well as Normalized Mean Error (NME in %) of landmarks localization on 300W [31], and WFLW [44] datasets.

Method	NME		Params (M)	FLOPs (B)
	300W	WFLW		
mnv2	4.70	9.57	2.42	0.60
mnv2_r	4.59	9.41		
ASMNet_nr	6.49	11.96	1.43	0.51
ASMNet	5.50	10.77		

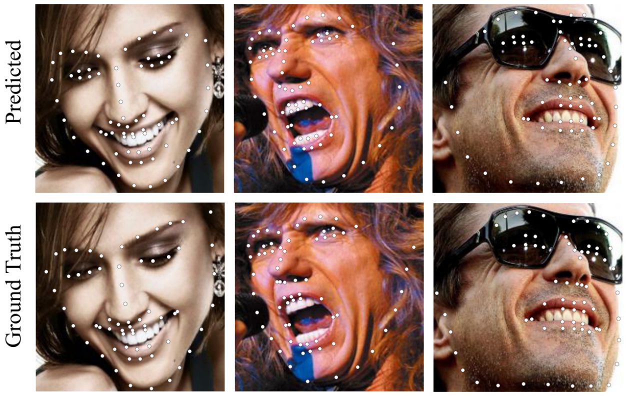

Figure 3: Facial Landmark detection using ASMNet over 300W [31] Challenging subset.

Table 2: Normalized Mean Error (in %) of 68-point landmarks localization on 300W [31] dataset.

Method	Normalized Mean Error		
	Common	Challenging	Fullset
RCN [16]	4.67	8.44	5.41
DAN [21]	3.19	5.24	3.59
PCD-CNN [22]	3.67	7.62	4.44
CPM [13]	3.39	8.14	4.36
DSRN [26]	4.12	9.68	5.21
SAN [12]	3.34	6.60	3.98
LAB [44]	2.98	5.19	3.49
DCFE [40]	2.76	5.22	3.24
mnv2	3.93	7.52	4.70
mnv2_r	3.88	7.35	4.59
ASMNet_nr	5.86	8.80	6.46
ASMNet	4.82	8.2	5.50

Evaluation on WFLW. Table 3 shows the performance of the state-of-the-art method and our proposed method over WFLW [44] and its 6 subsets. The performance of ASMNet is comparable to the performance of MobileNetV2 [33]. In other words, using the proposed ASM assisted loss function improves the model accuracy. Fig. 4 shows the some example of facial landmark detection using ASMNet on WFLW [44] dataset. While ASMNet can be taken as a very lightweight model, its performance is acceptable under different circumstances such as occlusion, extreme pose, expression, illumination, blur, and make-up.

Pose Evaluation. Neither 300W [31] nor WFLW [44] dataset do not provide the head pose information. Accordingly, we followed the method used by [48] and used another application to synthesizes the pose information. Although [48] used [3] for synthesizing the pose information, we used HopeNet [30] which is a state-of-the-art pose estimation method. Using HopeNet we acquired the *yaw*, *pitch*, and *roll* values of the 300W [31], and WFLW [44] images and used them as the ground truths for our network. Table 4 shows the mean absolute error (MAE) between HopeNet [30] results and our ASMNet.

Table 3: Normalized Mean Error (in %), failure rate (in %), and AUC of 98-point landmarks localization on WFLW [44] dataset.

Metric	Method	Test set	Pose	Expression	Illumination	Make-Up	Occlusion	Blur
Mean Error (%)	ESR [5]	11.13	25.88	11.47	10.49	11.05	13.75	12.20
	SDM [47]	10.29	24.10	11.45	9.32	9.38	13.03	11.28
	CFSS [58]	9.07	21.36	10.09	8.30	8.74	11.76	9.96
	DVLN [45]	6.08	11.54	6.78	5.73	5.98	7.33	6.88
	LAB [44]	5.27	10.24	5.51	5.23	5.15	6.79	6.32
	ResNet50(Wing+PDB) [15]	5.11	8.75	5.36	4.93	5.41	6.37	5.81
	mnv2	9.57	18.18	9.93	8.98	9.92	11.38	10.79
	mnv2_r	9.41	17.86	9.78	8.90	9.67	11.25	10.66
	ASMNet_nr	11.96	21.95	13.08	11.02	11.84	13.24	12.60
	ASMNet	10.77	21.11	12.02	9.93	10.55	12.34	11.62
Failure Rate	ESR [5]	35.24	90.18	42.04	30.80	38.84	47.28	41.40
	SDM [47]	29.40	84.36	33.44	26.22	27.67	41.85	35.32
	CFSS [58]	20.56	66.26	23.25	17.34	21.84	32.88	23.67
	DVLN [45]	10.84	46.93	11.15	7.31	11.65	16.30	13.71
	LAB [44]	7.56	28.83	6.37	6.73	7.77	13.72	10.74
	ResNet50(Wing+PDB) [15]	6.00	22.70	4.78	4.30	7.77	12.50	7.76
	mnv2	30.64	88.03	34.07	25.39	32.03	41.84	38.80
	mnv2_r	30.04	88.65	31.52	24.67	30.09	41.44	37.25
	ASMNet_nr	50.2	98.46	70.38	43.68	50.0	59.78	56.14
	ASMNet	39.12	98.41	59.87	33.38	38.34	48.64	46.31
AUC	ESR [5]	0.2774	0.0177	0.1981	0.2953	0.2485	0.1946	0.2204
	SDM [47]	0.3002	0.0226	0.2293	0.3237	0.3125	0.2060	0.2398
	CFSS [58]	0.3659	0.0632	0.3157	0.3854	0.3691	0.2688	0.3037
	DVLN [45]	0.4551	0.1474	0.3889	0.4743	0.4494	0.3794	0.3973
	LAB [44]	0.5323	0.2345	0.4951	0.5433	0.5394	0.4490	0.4630
	ResNet50(Wing+PDB) [15]	0.5504	0.3100	0.4959	0.5408	0.5582	0.4885	0.4918
	mnv2	0.2388	0.0096	0.1812	0.2510	0.2147	0.1719	0.1852
	mnv2_reg	0.2447	0.0099	0.1836	0.2563	0.2282	0.1779	0.1880
	ASMNet_nr	0.1024	0.0008	0.0414	0.1129	0.0941	0.0729	0.0797
	ASMNet	0.1637	0.0010	0.0714	0.1826	0.1653	0.1202	0.1268

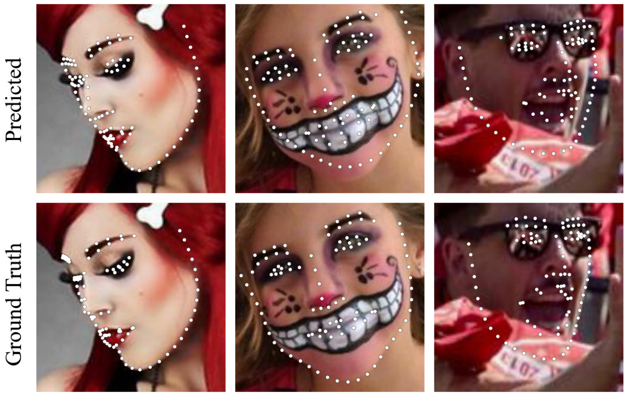

Figure 4: Facial Landmark detection using ASMNet over WFLW [44] dataset.

Table 4: Mean Absolute Error of pose estimation on 300W [31], WFLW [44] datasets compared to HopeNet[30].

Method		ASMNet_nr	ASMNet	mnv2	mnv2_r
300W [31]	yaw	2.41	1.62	1.75	1.71
	pitch	1.87	1.80	1.93	1.89
	roll	2.115	1.24	1.32	1.30
WFLW [44]	yaw	3.14	2.97	3.06	3.08
	pitch	2.99	2.93	3.03	2.94
	roll	2.23	2.21	2.26	2.22

Table 5: Mean Absolute Error of pose estimation on using ASMNet, JFA [48], and Yang*et. al* [50] on 300W [31].

Method	Pitch	Yaw	Roll
Yang*et. al* [50]	5.1	4.2	2.4
JFA [48]	3.0	2.5	2.6
ASMNet	1.80	1.62	1.24

In addition, we compare the performance of our proposed method with [48] as well as [50] in Table 5 using *Full* subset of 300W [31] dataset. As the results show, the performance of our lightweight ASMNet is comparable to HopeNet [30], which is a state-of-the-art method and outperforms the other methods as well. Besides, the performance of ASMNet is better than MobileNetV2 [33], even when it utilizes the ASM-LOSS function. Since in pose estimation task aligning the whole shape of the face is more crucial than aligning each landmark point, using ASM-LOSS function will lead to better performance.

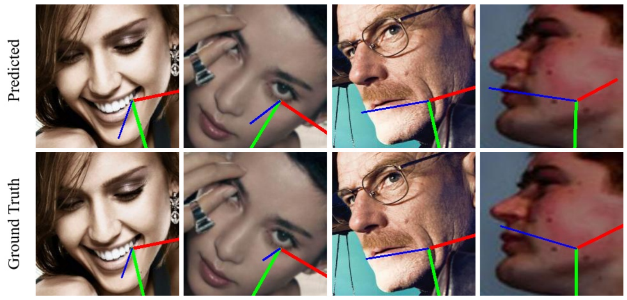

Figure 5: ASMNet can also estimate the head pose even in challenging conditions. The input images are from 300W [31] Challenging set.

Table 6: Model size (the number of model parameters) and computational cost (FLOPs) analysis of different networks.

Method	Backbone	#Params (M)	FLOPs (B)
DVLN [45]	VGG-16	132.0	14.4
SAN [12]	ResNet-152	57.4	10.7
LAB [44]	Hourglass	25.1	19.1
ResNet50 (Wing + PDB) [15]	ResNet-50	25	3.8
ASMNet	MobileNetV2 [33]	1.4	0.5
MobileNetV2 [33]	-	2.4	0.6

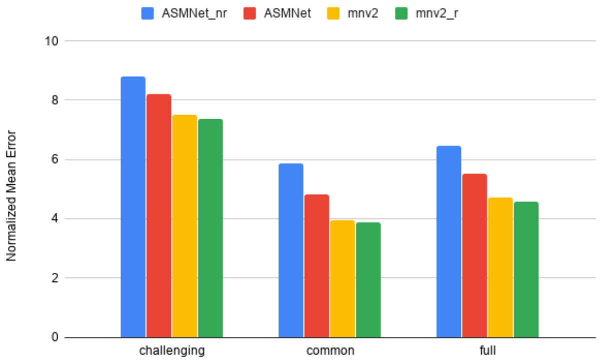

Figure 6: Comparing the performance of ASMNet, as well as MobileNetV2 [33] with and without the proposed ASM assisted loss function on 300W [31].

Table 7: Investigating the effect of using ASM assisted loss function both on MobileNetV2 [33] and ASMNet.

Method		NME reduction (in %)	
		ASMNet	mnv2
300W [31]	Full	0.96	0.11
	Common	1.58	0.05
	Challenging	0.60	0.17
WFLW [44]	Full	1.19	0.16
	Large pose	0.84	0.32
	Expression	1.06	0.15
	Illumination	1.09	0.08
	Makeup	1.29	0.25
	Occlusion	0.13	0.90
	Blur	0.98	0.13

Moreover, ASMNet is designed to use features generated in different layers of the neural network which enables it to outperforms MobileNetV2 [33] in pose estimation task. Fig. 5 shows the output of the pose estimation task.

Ablation Study. In Table 7 we study the ASM assisted loss by calculating the difference between normalized mean errors with and without ASM assisted loss both on ASMNet and MobileNetV2 [33]. As shown, using ASMNet utilized with ASM-LOSS function results in 0.96%, and 1.19% reduction in NME on 300W [31], and WFLW [44] respectively. Theses numbers are 0.11%, and 0.16% for MobileNetV2 [33]. According to the Table 7, and Fig. 6, using the ASM assisted loss function resulted in more accuracy improvement for ASMNet compared to MobileNetV2 [33]. Hence, it can be concluded that the ASM-LOSS function is capable of helping the lightweight CNN much more. In other words, when a lightweight network does not perform accurately enough, using the proposed ASM-LOSS function will play a vital role in improving the performance.

Model Size and Computational Cost Analysis. We calculate the number of network parameters as well as FLOPs to evaluate the model size and computational complexity. We calculate the FLOPs over the resolution of 224 × 224. As Table 6, although ASMNet is the smallest, its performance is comparable with MobileNetV2 [33], one of the best in *compact-class* models. Furthermore, since the idea behind ASMNet is to put a trade-off between accuracy and model performance, as we can see in Table 6, adding ASM assisted loss to a lightweight model such as ASMNet, and MobileNetV2 [33], results in the accuracy improvement.

5. Conclusion and Future Work

In this paper, we proposed ASMNet, a lightweight CNN architecture with multi-task learning for facial landmark points detection and pose estimation. We proposed a loss function that is assisted using ASM [9, 27] that increases the network accuracy. We built our network (called ASMNet) using a small portion of MobileNetV2 [33]. The proposed ASMNet architecture is about 2 times smaller than MobileNetV2 [33], while the accuracy remains at the same rate. The results of evaluating ASMNet and our proposed ASM assisted loss on widely used 300W [31], and WFLW [44] datasets show that the accuracy of ASMNet is acceptable in detecting facial landmark points and estimating head pose. The proposed method has the potential to be used in other computer vision tasks such as human body joint tracking or other shape objects that can be modeled using ASM. Hence, as a future research direction, we will investigate using ASMNet for such applications.

References

[1] G. Antonini, V. Popovici, and J.-P. Thiran. Independent component analysis and support vector machine for face feature extraction. In *International Conference on Audio-and Video-Based Biometric Person Authentication*, pages 111–118. Springer, 2003.

[2] S. Arca, P. Campadelli, and R. Lanzarotti. A face recognition system based on automatically determined facial fiducial points. *Pattern recognition*, 39(3):432–443, 2006.

[3] A. Asthana, S. Zafeiriou, S. Cheng, and M. Pantic. Robust discriminative response map fitting with constrained local models. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3444–3451, 2013.

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

[5] X. Cao, Y. Wei, F. Wen, and J. Sun. Face alignment by explicit shape regression. *International Journal of Computer Vision*, 107(2):177–190, 2014.

[6] D. Chen, S. Ren, Y. Wei, X. Cao, and J. Sun. Joint cascade face detection and alignment. In *European Conference on Computer Vision*, pages 109–122. Springer, 2014.

[7] T. Cootes, E. Baldock, and J. Graham. An introduction to active shape models. *Image processing and analysis*, pages 223–248, 2000.

[8] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. In *European conference on computer vision*, pages 484–498. Springer, 1998.

[9] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models-their training and application. *Computer vision and image understanding*, 61(1):38–59, 1995.

[10] T. F. Cootes, C. J. Taylor, et al. Statistical models of appearance for computer vision, 2004.

[11] D. Cristinacce and T. F. Cootes. Feature detection and tracking with constrained local models. In *Bmvc*, volume 1, page 3. Citeseer, 2006.

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

[13] X. Dong, S.-I. Yu, X. Weng, S.-E. Wei, Y. Yang, and Y. Sheikh. Supervision-by-registration: An unsupervised approach to improve the precision of facial landmark detectors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 360–368, 2018.

[14] C. Du, Q. Wu, J. Yang, and Z. Wu. Svm based asm for facial landmarks location. In *2008 8th IEEE International Conference on Computer and Information Technology*, pages 321–326. IEEE, 2008.

[15] Z.-H. Feng, J. Kittler, M. Awais, P. Huber, and X.-J. Wu. Wing loss for robust facial landmark localisation with convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2235–2245, 2018.

[16] S. Honari, J. Yosinski, P. Vincent, and C. Pal. Recombinator networks: Learning coarse-to-fine feature aggregation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5743–5752, 2016.

[17] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, et al. Speed/accuracy trade-offs for modern convolutional object detectors. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7310–7311, 2017.

[18] Z. Huang, X. Zhao, S. Shan, R. Wang, and X. Chen. Coupling alignments with recognition for still-to-video face recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3296–3303, 2013.

[19] A. Jourabloo and X. Liu. Pose-invariant 3d face alignment. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3694–3702, 2015.

[20] A. Jourabloo, M. Ye, X. Liu, and L. Ren. Pose-invariant face alignment with a single cnn. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3200–3209, 2017.

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

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

[23] V. Le, J. Brandt, Z. Lin, L. Bourdev, and T. S. Huang. Interactive facial feature localization. In *European conference on computer vision*, pages 679–692. Springer, 2012.

[24] C. Lu and X. Tang. Surpassing human-level face verification performance on lfw with gaussianface. In *Twenty-ninth AAAI conference on artificial intelligence*, 2015.

[25] P. Martins, R. Caseiro, and J. Batista. Generative face alignment through 2.5 d active appearance models. *Computer Vision and Image Understanding*, 117(3):250–268, 2013.

[26] X. Miao, X. Zhen, X. Liu, C. Deng, V. Athitsos, and H. Huang. Direct shape regression networks for end-to-end face alignment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5040–5049, 2018.

[27] S. Ordas, L. Boisrobert, M. Huguet, and A. Frangi. Active shape models with invariant optimal features (iof-asm) application to cardiac mri segmentation. In *Computers in Cardiology, 2003*, pages 633–636. IEEE, 2003.

[28] R. Ranjan, V. M. Patel, and R. Chellappa. Hyperface: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(1):121–135, 2017.

[29] S. Ren, X. Cao, Y. Wei, and J. Sun. Face alignment at 3000 fps via regressing local binary features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1685–1692, 2014.

[30] N. Ruiz, E. Chong, and J. M. Rehg. Fine-grained head pose estimation without keypoints. In *The IEEE Conference*

on Computer Vision and Pattern Recognition (CVPR) Workshops, June 2018.

[31] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic. 300 faces in-the-wild challenge: The first facial landmark localization challenge. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 397–403, 2013.

[32] A. A. Salah, H. Cinar, L. Akarun, and B. Sankur. Robust facial landmarking for registration. In *Annales des Télécommunications*. Springer, 2007.

[33] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4510–4520, 2018.

[34] J. M. Saragih, S. Lucey, and J. F. Cohn. Deformable model fitting by regularized landmark mean-shift. *International journal of computer vision*, 91(2):200–215, 2011.

[35] S. Soltanpour, B. Boufama, and Q. J. Wu. A survey of local feature methods for 3d face recognition. *Pattern Recognition*, 72:391–406, 2017.

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

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

[38] G. Trigeorgis, P. Snape, M. A. Nicolaou, E. Antonakos, and S. Zafeiriou. Mnemonic descent method: A recurrent process applied for end-to-end face alignment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4177–4187, 2016.

[39] S. Tulyakov and N. Sebe. Regressing a 3d face shape from a single image. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3748–3755, 2015.

[40] R. Valle, J. M. Buenaposada, A. Valdés, and L. Baumela. A deeply-initialized coarse-to-fine ensemble of regression trees for face alignment. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 585–601, 2018.

[41] F. Vicente, Z. Huang, X. Xiong, F. De la Torre, W. Zhang, and D. Levi. Driver gaze tracking and eyes off the road detection system. *IEEE Transactions on Intelligent Transportation Systems*, 16(4):2014–2027, 2015.

[42] P. Viola, M. Jones, et al. Robust real-time object detection. *International journal of computer vision*, 4(34-47):4, 2001.

[43] D. Vukadinovic and M. Pantic. Fully automatic facial feature point detection using gabor feature based boosted classifiers. In *2005 IEEE International Conference on Systems, Man and Cybernetics*, volume 2, pages 1692–1698. IEEE, 2005.

[44] W. Wu, C. Qian, S. Yang, Q. Wang, Y. Cai, and Q. Zhou. Look at boundary: A boundary-aware face alignment algorithm. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2129–2138, 2018.

[45] W. Wu and S. Yang. Leveraging intra and inter-dataset variations for robust face alignment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 150–159, 2017.

[46] Y. Wu, C. Gou, and Q. Ji. Simultaneous facial landmark detection, pose and deformation estimation under facial occlusion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3471–3480, 2017.

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

[48] X. Xu and I. A. Kakadiaris. Joint head pose estimation and face alignment framework using global and local cnn features. In *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, pages 642–649. IEEE, 2017.

[49] H. Yang, X. Jia, C. C. Loy, and P. Robinson. An empirical study of recent face alignment methods. *arXiv preprint arXiv:1511.05049*, 2015.

[50] H. Yang, W. Mou, Y. Zhang, I. Patras, H. Gunes, and P. Robinson. Face alignment assisted by head pose estimation. *arXiv preprint arXiv:1507.03148*, 2015.

[51] S. Yang, P. Luo, C.-C. Loy, and X. Tang. Wider face: A face detection benchmark. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5525–5533, 2016.

[52] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

[53] J. Zhang, S. Shan, M. Kan, and X. Chen. Coarse-to-fine autoencoder networks (cfan) for real-time face alignment. In *European conference on computer vision*, pages 1–16. Springer, 2014.

[54] Z. Zhang, P. Luo, C. C. Loy, and X. Tang. Facial landmark detection by deep multi-task learning. In *European conference on computer vision*, pages 94–108. Springer, 2014.

[55] Z. Zhang, W. Zhang, H. Ding, J. Liu, and X. Tang. Hierarchical facial landmark localization via cascaded random binary patterns. *Pattern Recognition*, 48(4):1277–1288, 2015.

[56] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM computing surveys (CSUR)*, 35(4):399–458, 2003.

[57] D. Zhou, D. Petrovska-Delacrétaz, and B. Dorizzi. Automatic landmark location with a combined active shape model. In *2009 IEEE 3rd International Conference on Biometrics: Theory, Applications, and Systems*, pages 1–7. IEEE, 2009.

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

[59] X. Zhu, Z. Lei, X. Liu, H. Shi, and S. Z. Li. Face alignment across large poses: A 3d solution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 146–155, 2016.

[60] X. Zhu and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2879–2886, June 2012.