Joint Face Detection and Facial Motion Retargeting for Multiple Faces

Bindita Chaudhuri[1]*, Noranart Vesdapunt[2], Baoyuan Wang[2]
[1]University of Washington, [2]Microsoft Corporation
[1]bindita@cs.washington.edu, [2]{noves,baoyuanw}@microsoft.com

Abstract

Facial motion retargeting is an important problem in both computer graphics and vision, which involves capturing the performance of a human face and transferring it to another 3D character. Learning 3D morphable model (3DMM) parameters from 2D face images using convolutional neural networks is common in 2D face alignment, 3D face reconstruction etc. However, existing methods either require an additional face detection step before retargeting or use a cascade of separate networks to perform detection followed by retargeting in a sequence. In this paper, we present a single end-to-end network to jointly predict the bounding box locations and 3DMM parameters for multiple faces. First, we design a novel multitask learning framework that learns a disentangled representation of 3DMM parameters for a single face. Then, we leverage the trained single face model to generate ground truth 3DMM parameters for multiple faces to train another network that performs joint face detection and motion retargeting for images with multiple faces. Experimental results show that our joint detection and retargeting network has high face detection accuracy and is robust to extreme expressions and poses while being faster than state-of-the-art methods.

1. Introduction

Facial gestures are an effective medium of non-verbal communication, and communication becomes more appealing through 3D animated characters. This has led to extensive research [8, 3, 20] in developing techniques to retarget human facial motion to 3D animated characters. The standard approach is to model human face by a 3D morphable model (3DMM)[5] and learn the weights of a linear combination of blendshapes that fits to the input face image. The learned "expression" weights and "head pose" angles are then directly mapped to semantically equivalent blendshapes of the target 3D character rig to drive the desired facial animation. Previous methods, such as [8], formulate 3DMM fitting as an optimization problem of regressing the 3DMM parameters from the input image. However, these methods require significant pre-processing or post-processing operations to get the final output.

Using deep convolution neural networks, recent works have shown remarkable accuracy in regressing 3DMM parameters from a 2D image. However, while 3DMM fitting with deep learning is frequently used in related domains like 2D face alignment[61, 7], 3D face reconstruction[38, 16, 25, 13] etc., it hasn't been proven yet as an effective approach for facial motion retargeting. This is because 1) face alignment methods focus more on accurate facial landmark localization while face reconstruction methods focus more on accurate 3D shape and texture reconstruction to capture the fine geometric details. In contrast, facial retargeting to an arbitrary 3D character only requires accurate transfer of facial expression and head pose. However, due to the ambiguous nature of this ill-posed problem of extracting 3D face information from 2D image, both facial expression and head pose learned by those methods are generally *sub-optimal* as they are not well decoupled from other information like identity. 2) Unlike alignment and reconstruction, retargeting often requires real-time tracking and transfer of the facial motion. However, existing methods for alignment and reconstruction are highly memory intensive and often involve complex rendering of the 3DMM as intermediate steps, thereby making these methods difficult to deploy on light-weight hardware like mobile phones.

It is important to note that all previous deep learning based 3DMM fitting methods work on a single face image assuming face is already detected and cropped. To support multiple faces in a single image, a straightforward approach is to run a face detector on the image first to detect the all face regions and then perform the retargeting operations on each face individually. Such an approach, however, requires additional execution time for face detection and the computational complexity increases linearly with the number of faces in the input image. Additionally, tracking multiple faces with this approach becomes difficult when people move in and out from the frame or occlude each other. In the literature of joint face detection and alignment, existing methods [12, 55, 15] either use a random forest to predict the face bounding boxes and landmarks or adopt an iterative

*Work primarily done during an internship at Microsoft.

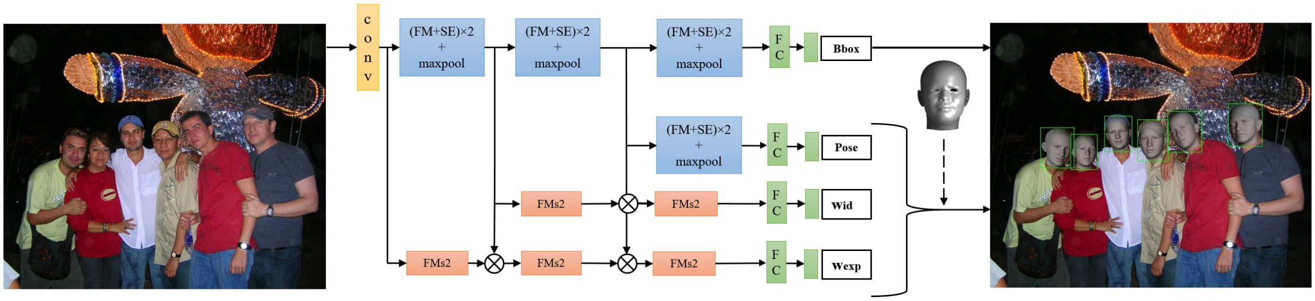

Figure 1: Network architecture. Our end-to-end joint face detection and retargeting network is specifically tailored to incorporate multi-scale representation disentangling. The building blocks are Fire Modules (FM) [23] and squeeze-and-excitation (SE) blocks [21] which are designed for real-time application. The multi-scale branches use FM with stride 2 (FMs2) to allow concatenation. The $Pose$, w_{id} (identity parameters) and w_{exp} (expression parameters) together with 3DMM generate the 3D mesh for every face bounding box.

two-step approach to generate region proposals and predict the landmark locations in the proposed regions. However, these methods are primarily optimized for regressing accurate landmark locations rather than 3DMM parameters.

To this end, we divide our work into two parts. In the first part, we propose a multitask learning network to directly regress the 3DMM parameters from a well-cropped 2D image with a single face; we call this as Single Face Network (SFN). Our 3DMM parameters are grouped into: a) identity parameters that contain the face shape information, b) expression parameters that captures the facial expression, c) pose parameters that include the 3D rotation and 3D translation of the head and d) scale parameters that links the 3D face with the 2D image. We have observed that pose and scale parameters require global information while identity and expression parameters require different level of information, so we propose to emphasize on high level image features for pose and scale and the multi-scale features for identity and expression. Our network architecture is designed such that different layers embed image features at different resolutions, and these multi-scale features help in disentangling the parameter groups from each other. In the second part, we propose a single end-to-end trainable network to jointly detect the face bounding boxes and regress the 3DMM parameters for multiple faces in a single image. Inspired by YOLO[33] and its variants[34, 35], we design our Multiple Face Network (MFN) architecture that takes a 2D image as input and predicts the centroid position and dimensions of the bounding box as well as the 3DMM parameters for each face in the image. Unfortunately, existing publicly available multi-face image datasets provide ground truth for face bounding boxes only and not 3DMM parameters. Hence, we leverage our SFN to generate the weakly labelled "ground truth" for 3DMM parameters for each face to train our MFN. Experimental results show that our MFN not only performs well for multi-face retargeting but also improves the accuracy of face detection. Our main contributions can be summarized as follows:

1. We design a multitask learning network, specifically tailored for facial motion retargeting by casting the scale prior into a novel network topology to disentangle the representation learning. Such network has been proven to be crucial for both single face and multiple face 3DMM parameters estimation.
2. We present a novel top-down approach using an end-to-end trainable network to jointly learn the face bounding box locations and the 3DMM parameters from an image having multiple faces with different poses and expressions.
3. Our system is easy to deploy into practical applications without requiring separate face detection for pose and expression retargeting. Our joint network can be run in real-time on mobile devices without engineering level optimization, e.g. only 39ms on Google Pixel 2.

2. Related Work

2.1. 2D Face Alignment and 3D Face Reconstruction

Early methods like [27] used a cascade of decision trees or other regressors to directly regress the facial landmark locations from a face image. Recently, the approach of regressing 3DMM parameters using CNNs and fitting 3DMM to the 2D image has become popular. While Jourabloo et al. [26] use a cascade of CNNs to alternately regress the shape (identity and expression) and pose parameters, Zhu et al. [61, 60] perform multiple iterations of a single CNN to regress the shape and pose parameters together. These methods use large networks and require 3DMM in the network during testing, thereby requiring large memory and execution time. Regressing 3DMM parameters using CNNs is also popular in face reconstruction [46, 38, 18, 45]. Richardson et al. [39] uses a coarse-to-fine approach to capture fine details in addition to face geometry. However, reconstruction methods also regress texture and focus more on capturing fine geometric details. For joint face alignment and reconstruction, [17] regresses a position regression map from the image and [47] regresses the parameters of a nonlinear 3DMM using an unsupervised encoder-decoder network. For joint face detection and alignment, recent methods either use a mixture of trees [31] or a cascade of CNNs [12, 55]. In [15], separate networks are trained to perform different tasks like proposing regions, classifying and regressing the bounding boxes from the regions, predicting the landmark locations in those regions etc. In [32], region proposals are first generated with selective search algorithm and bounding box and landmark locations are regressed for each proposal using a multitask learning network. In contrast, we use a single end-to-end network to do join face detection and 3DMM fitting for face retargeting purposes.

2.2. Performance-Based Animation

Traditional performance capture systems (using either depth cameras or 3D scanners for direct mesh registration with depth data) [43, 6, 48] require complex hardware setup that is not readily available. Among the methods which use 2D images as input, the blendshape interpolation technique [8, 41] is most popular. However, these methods require dense correspondence of facial points [37] or user-specific adaptations [30, 9] to estimate the blendshape weights. Recent CNN based approaches either require depth input [29, 19] or regress character-specific parameters with several constraints [3]. Commercial software products like Faceshift [1], Faceware [2] etc. perform realtime retargeting but with poor expression accuracy [3].

2.3. Object Detection and Keypoint Localization

In the literature of multiple object detection and classification, Fast RCNN [36] and YOLO [33] are the two most popular methods with state-of-the-art performance. While [36] uses a region proposal network to get candidate regions before classification, [33] performs joint object location regression and classification. Keypoint localization for multiple objects is popularly used for human pose estimation [28, 11] or object pose estimation [44]. In case of faces, landmark localization for multiple faces can be done in two approaches: *top-down approach* where landmark locations are detected after detecting face regions and *bottom-up approach* where the facial landmarks are initially predicted individually and then grouped together into face regions. In our method, we adopt the top-down approach.

3. Methodology

3.1. 3D Morphable Model

The 3D mesh of a human face can be represented by a multilinear 3D Morphable Model (3DMM) as

$$\mathcal{M} = \mathcal{V} \times \mathrm{b}_{\mathrm{id}} \times \mathrm{b}_{\mathrm{exp}} \tag{1}$$

where $\mathcal{V}$ is the mean neutral face, b_{id} are the identity bases and $\mathrm{b}_{\mathrm{exp}}$ are the expression bases. We use the face tensor provided by FacewareHouse [10] as 3DMM, where $\mathcal{V} \in \mathbb{R}^{11510 \times 3}$ denotes $11,510$ 3D co-ordinates of the mesh vertices, b_{id} denotes 50 shape bases obtained by taking PCA over 150 identities and $\mathrm{b}_{\mathrm{exp}}$ denotes 47 bases corresponding to 47 blendshapes (1 neutral and 46 micro-expressions). To reduce the computational complexity, we manually mark 68 vertices in $\mathcal{V}$ as the facial landmark points based on [31] and create a reduced face tensor $\hat{\mathcal{M}} \in \mathbb{R}^{204 \times 50 \times 47}$ for use in our networks. Given a set of identity parameters $w_{\mathrm{id}} \in \mathbb{R}^{50 \times 1}$, expression parameters $w_{\mathrm{exp}} \in \mathbb{R}^{47 \times 1}$, 3D rotation matrix $\mathbf{R} \in \mathbb{R}^{3 \times 3}$, 3D translation parameters $\mathbf{t} \in \mathbb{R}^{3 \times 1}$ and a scale parameter (focal length) f, we use weak perspective projection to get the 2D landmarks $\mathbf{P_{lm}} \in \mathbb{R}^{68 \times 2}$ as:

$$\mathbf{P_{lm}} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \end{bmatrix} [\mathbf{R} * (\hat{\mathcal{M}} * w_{\mathrm{id}} * w_{\mathrm{exp}}) + \mathbf{t}] \tag{2}$$

where $w_{\mathrm{exp}}[1] = 1 - \sum_{i=2}^{47} w_{\mathrm{exp}}[i]$ and $w_{\mathrm{exp}}[i] \in [0,1], i = 2, \ldots, 47$. We use a unit quaternion $\mathbf{q} \in \mathbb{R}^{4 \times 1}$ [60] to represent 3D rotation and convert it into rotation matrix for use in equation 2. Please note that, for retargeting purposes, we omit the learning of texture and lighting in the 3DMM.

3.2. Multi-scale Representation Disentangling

A straightforward way of holistically regressing all the 3DMM parameters together through a fully connected layer on top of one shared representation will not be optimal particularly for our problem where each group of parameters has strong semantic meanings. Intuitively speaking, head pose learning does not require detailed local face representations since it is fundamentally independent of skin texture and subtle facial expressions, which has also been observed in recent work on pose estimation [58]. However, for identity learning, a combination of both local and global representations would be necessary to differentiate among different persons. For example, some persons have relatively small eyes but fat cheek while others have big eyes and thin cheek, so both the local features around the eyes and the overall face silhouette would be important to approximate the face shape. Similarly, expression learning possibly requires even fine-grained granularity of different scales of representations. Single eye wink, mouth grin and big laugh clearly require three different levels of representations to differentiate them from other expressions.

Another observation is, given the 2D landmarks of an image, there exist multiple combinations of 3DMM parameters that can minimize the 2D landmark loss. This ambiguity would cause additional challenges to the learning to favor the semantically more meaningful combinations. For example, as shown in Fig. 2, we can still minimize the 2D landmark loss by rotating the head and using different

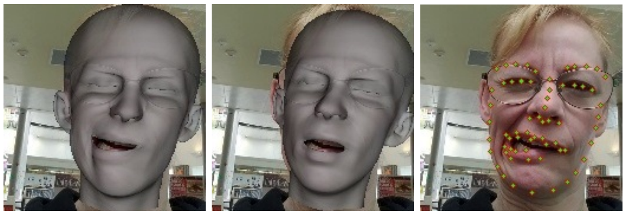

Figure 2: Importance of representation disentangling. Left: correctly fitted mesh with jaw left; Middle: incorrectly fitted mesh with large roll angle; Right: projected landmarks from both the meshes are still the same.

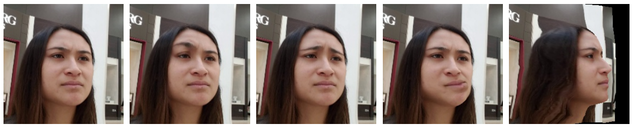

Figure 3: Synthesized images for disentangled regularization.

identity coefficients to accommodate the jaw left even without a correct jaw left expression coefficient. Motivated by both the multi-scale prior and the ambiguous nature of this problem, we designed a novel network architecture that is specifically tailored for facial retargeting applications as illustrated in Fig. 1, where pose is only learned through the final global features while expression learning depends on the concatenation of multi-scale representations.

Disentangled Regularization In addition to the above network design, we add regularization during the training to further enforce the disentangled representation learning. We augment each face image by random translation/rotation/scale to generate multiple different images with the same identity and expression coefficients. In addition, we edit the images using image warping techniques to slightly change the facial expression without changing the pose and identity. Fig. 3 shows a few such synthesized examples for the same identity.

3.3. Single Face Retargeting Network

When the face bounding box is given, we can train a single face retargeting network to output 3DMM parameters for each cropped face image using the network architecture proposed above. Fortunately, many public datasets [40, 10, 22, 61] already provide bounding boxes along with 68 2D facial landmark points. To encourage disentangling, we fit 3DMM parameters for each cropped single face image using the optimization method of [7] and treat them as ground truth for our network in addition to the landmarks. Although individual optimization may result in over-fitting and noisy ground truth, our network can intrinsically focus more on the common global patterns from the training data. To achieve this, we initially train with a large weight on the L1 loss with respect to the ground truth (g) parameters, and then gradually decay this weight to trust more on the 2D landmark loss, as shown in the following loss function:

$$\tau * \left\{ \frac{1}{50}\sum_{i=1}^{50}|w_{\text{id}_i} - w^g_{\text{id}_i}| + \frac{1}{46}\sum_{i=1}^{46}|w_{\text{exp}_i} - w^g_{\text{exp}_i}| \right. \\ \left. + \frac{1}{4}\sum_{i=1}^{4}|\mathbf{R}_i - \mathbf{R}^g_i| \right\} + \sqrt{\frac{1}{68}\sum_{i=1}^{68}(\mathbf{P}_{\text{lm}_i} - \mathbf{P}^g_{\text{lm}_i})^2} \quad (3)$$

where τ denotes decay parameter with respect to epoch. We choose τ = 10/epoch across all experiments. Note that, although we drop the 3D translation and scale ground truth loss to allow 2D translation and scaling augmentation, the translation and scale parameters can still be learned by the 2D landmark loss.

3.4. Joint Face Detection and Retargeting

Our goal is to save computation cost by performing both face detection and 3DMM parameter estimation simultaneously instead of sequentially running a separate face detector and then single face retargeting network on each face separately. The network could potentially also benefit from the cross domain knowledge, especially for detection task, where introducing 3DMM gives the prior on how the face should look like in 3D space which complements the 2D features in separate face detection framework.

Inspired by YOLO [33], our joint network is designed to predict 3DMM parameters for each anchor point in additional to bounding box displacement and objectness. We divide the input image into 9×9 grid and predict a vector of length $4 + 1 + (50 + 46 + 4 + 3 + 1) = 109$ for a bounding box in each grid cell. Here 4 denotes 2D co-ordinates of the centroid, width and height of the face bounding box, 1 denotes the confidence score for the presence of a face in that cell and the rest denote the 3DMM parameters for the face in the cell. We also adopt the method of starting with 5 anchor boxes as bounding box priors. Our final loss function is the summation of Eq. 3 across all grids and anchors, as shown in the following:

$$\tau * \left\{ \frac{1}{50}\sum_{j=1}^{9^2}\sum_{k=1}^{5}\sum_{i=1}^{50}\mathbb{1}_{ijk}|w_{\text{id}_{ijk}} - w^g_{\text{id}_{ijk}}| \right. \\ + \frac{1}{46}\sum_{j=1}^{9^2}\sum_{k=1}^{5}\sum_{i=1}^{46}\mathbb{1}_{ijk}|w_{\text{exp}_{ijk}} - w^g_{\text{exp}_{ijk}}| \\ \left. + \frac{1}{4}\sum_{j=1}^{9^2}\sum_{k=1}^{5}\sum_{i=1}^{4}\mathbb{1}_{ijk}|\mathbf{R}_{ijk} - \mathbf{R}^g_{ijk}| \right\} \\ + \sqrt{\frac{1}{68}\sum_{j=1}^{9^2}\sum_{k=1}^{5}\sum_{i=1}^{68}\mathbb{1}_{ijk}(\mathbf{P}_{\text{lm}_{ijk}} - \mathbf{P}^g_{\text{lm}_{ijk}})^2} \quad (4)$$

where $\mathbb{1}_{ijk}$ denotes whether a kth bounding box predictor in cell j contains a face. Since there are no publicly available multi-face datasets that provide both bounding box location and 3DMM parameters for each face, for proof-of-concept, we obtain the 3DMM ground truth by running our single face retargeting network on each face separately. The centroid co-ordinates and the bounding box dimensions are calculated in the same manner as in [33] and we use the same loss functions for these values (see supplementary).

4. Experimental Setup

4.1. Datasets

For single face retargeting, we combine multiple datasets to have a good training set for accurate prediction of each group of 3DMM parameters. 300W-LP contains many large poses and Facewarehouse is a rich dataset for expressions. The ground truth 68 2D landmarks provided by these datasets are used to obtain 3DMM ground truth by [7]. LFW and AFLW2000-3D are used as test sets for static images and 300VW is used as test set for tracking on videos. For multiple face retargeting, AFW has ground truth bounding boxes, pose angles and 6 landmarks and is used as a test set for static images, while FDDB and WIDER only provide bounding box ground truth and are therefore used for training (WIDER test set is kept separate for testing). Music videos dataset is used to test our MFN performance on videos. We remove all images with more than 20 faces and also remove faces whose bounding box dimensions are <2% of the image dimensions from both the training and test sets. This mainly includes faces in the background crowd with size less than 5×5 pixels. The reason is that determining the facial expressions for such small faces is ambiguous even for human eyes and hence retargeting is not meaningful. More dataset details are summarized in Table 1. We use an 80-20 split of the training set for training and validation. To measure the performance of expression accuracy, we manually collect an expression test set by selecting some extreme expression images (Fig. 7). The number of images in each of the expression categories are: eye close: 185, eye wide: 70, brow raise: 124, brow anger: 100, mouth open: 81, jaw left/right: 136, lip roll: 64, smile: 105, kiss: 143, total: 1008 images.

Dataset		**#images**	**#faces**
SFN	300W-LP [40, 61]	61225	61225
	FacewareHouse [10]	5000	500
	LFW [22]	12639	12639
	AFLW2000-3D [61]	2000	2000
	300VW [42]	114 (videos)	218K
MFN	FDDB [24]	2845	5171
	WIDER [53]	11905	56525
	AFW [31]	205	1000
	Music videos [57]	8 (videos)	-

Table 1: Number of images or videos and faces for each dataset used in training and testing of our networks.

4.2. Evaluation Metrics

We use 4 metrics: 1) Average Precision (AP) with different intersection-over-union thresholds as defined in [35] to evaluate our MFN performance for face detection, 2) Normalized Mean Error (NME) defined as the Euclidean distance between the predicted and ground truth 2D landmarks averaged over 68 landmarks and normalized by the bounding box dimensions, 3) Area under the Curve (AUC) of the Cumulative Error Distribution curve for landmark error normalized by the diagonal distance of ground truth bounding box [15], and 4) expression metric defined as the mean absolute distance between the predicted expression parameters with respect to the ground truth. The value of each expression parameter lies between 0 and 1 as in [10].

4.3. Implementation Details

4.3.1 Training Configuration

Our networks are implemented in Keras [14] with Tensorflow backend and trained using Adam optimizer with batch size 32. The initial learning rate (10^{-3} for SFN and 10^{-4} for MFN) is decreased by 10 times (up to 10^{-6}) when the validation loss does not change over 5 consecutive epochs. Training takes about a day on a Nvidia GTX 1080 for each network. For data augmentation, we use random scaling in the range [0.8,1.2], random translation of 0-10%, color jitter and in-plane rotation. These augmentation techniques improve the performance of SFN and also help in generating more accurate ground truth for individual faces for MFN.

4.3.2 Single Face Retargeting Architecture

Our network takes 128x128 resized image as input. In the first layer, we use a 7 × 7 convolution layer with 64 filters and stride 2 followed by a 2 × 2 maxpooling layer to capture the fine details in the image. The following layers are made up of Fire Modules(FM) of SqueezeNet [23] (with 16 and 64 filters in squeeze and expand layers respectively) and squeeze-and-excite modules(SE) of [21] in order to compress the model size and reduce the model execution time without compromising the accuracy. At the end of network, we use a global average pooling layer followed by fully connected (FC) layers to generate the parameters. The penultimate FC layers each has 64 units with ReLU activation and sigmoid activation is used for the last FC layer of the expression branch to restrict the values between 0 and 1. To realize the multiscale prior and the disentangled learning, we concatenate the features at different scales and form separate branches for each group of parameters. The extra branches are built with the same blocks as the main branch,

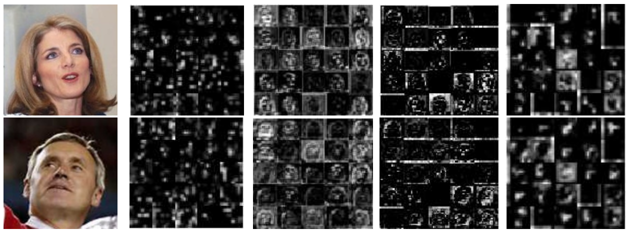

Figure 4: Visualization of learned features. From left to right in each row: input image, features for single scale SFN, features for expression branch of multi-scale SFN, features for identity branch of multi-scale SFN, features for pose branch of multi-scale SFN.

but we reduce the channel size by half to restrict the extra computation cost.

4.3.3 Joint Detection and Retargeting Architecture

Our joint detection and retargeting network architecture is similar to Tiny DarkNet [33] with the final layer changed to predict a tensor of size $9 \times 9 \times 5 \times 109$. However, since we only have one object class (face) in our problem, we reduce the number of filters in each layer to a quarter of their original values. For multi-scale version, we change the input image size to 288×288 and extend the multi-scale backbone for single face retargeting by changing the output of each branch to accommodate grid output (Fig. 1). The pose branch outputs change from 4 (R) + 3 (T) + 1 (f) = 8 to $9 \times 9 \times 5 \times 8$. The expression branch outputs change from 46 to $9 \times 9 \times 5 \times 46$, and identity branch outputs change from 50 to $9 \times 9 \times 5 \times 50$. One extra branch is also added to output objectness and bounding box location ($9 \times 9 \times 5 \times (4+1)$). In total, multi-scale version outputs the same dimension as single-scale, but the output channels are split with respect to each type of branch.

5. Results

5.1. Importance of Multi-Scale Representation

Our multi-scale network design, unlike the single scale design (refer to the supplementary material), reduces the load on the network to learn complex features by allowing the network to concentrate on different image features to learn different parameters. In Fig. 4, we see that single scale network learns generic facial features that combine the representations for identity, expression and pose. On the other hand, multi-scale network learns different levels of representation (pixel-level detailed features for expression, region level features for identity and global aggregate features for pose). We have randomly chosen only 25 filter outputs at level 3 of our SFN for clearer visualization. Table 2 shows that our multi-scale design not only reduces NME for single face images using SFN but also improves the performance of MFN in terms of both NME (by generating a better weakly supervised ground truth) and AP for detection. Clearly, different feature representations are crucial to accurately learn different groups of parameters. By reducing the network load, this design also allows model compression so that multi-scale networks can be of comparable size with respect to single scale networks while having better accuracy. Fig. 5 shows that the multi-scale design predicts more accurate expression parameters (first row has correct landmarks for closed eyes) and identity parameters (second row has correct landmarks that fit the face shape) while being robust to large poses (second row), illumination (first row) and occlusion (third row).

Model	Evaluation			
	NME	Multi Face		
	(%)	AP	AP50	AP75
(1) MFN (detection only)	-	92.1	99.2	94.3
(2) Single scale SFN	2.16	-	-	-
(3) Multi-scale SFN	1.91	-	-	-
(4) SS-MFN + GT from (2)	2.89	97.5	99.8	98.1
(5) SS-MFN + GT from (3)	2.65	98.2	100	98.9
(6) MS-MFN + GT from (3)	2.23	98.8	100	99.3

Table 2: Quantitative evaluation of our SFN and MFN. SS-MFN and MS-MFN denote single scale and multi-scale MFN respectively. NME values are calculated for LFW (single faces) and AP values are calculated for AFW.

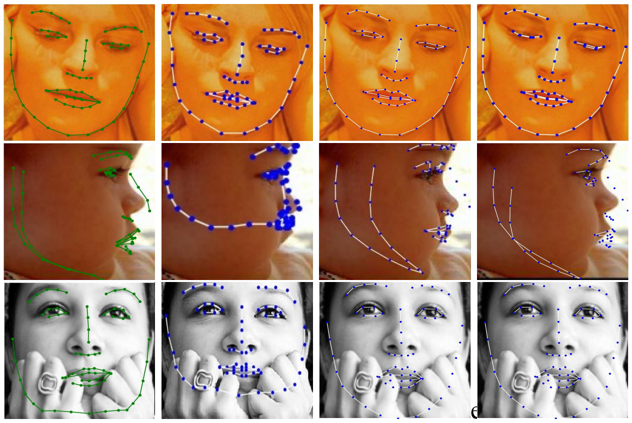

Figure 5: 2D face alignment results for AFLW2000-3D. Column 1: original image with ground truth landmarks; Column 2: results using [7]; Column 3: our single scale SFN; Column 4: our multi-scale SFN.

5.2. Comparison with 2D Alignment Methods

Even though we aim to predict the 3DMM parameters for retargeting applications, our model can naturally serve the purpose for 2D face alignment (via 3D). Therefore, we can evaluate our model from the performance of 2D alignment perspective. Table 3 compares the performance of our single scale and multi-scale SFN with state-of-the-art 2D face alignment methods (compared under the same settings). As can be seen, our model achieves much smaller er-

Method	[0°,30°]	[30°,60°]	[60°,90°]	Mean
SDM [50]	3.67	4.94	9.67	6.12
3DDFA [61]	3.78	4.54	7.93	5.42
3DDFA2 [61]	3.43	4.24	7.17	4.94
Yu et al. [54]	3.62	6.06	9.56	6.41
3DSTN [4]	3.15	4.33	5.98	4.49
DFF [25]	3.20	4.68	6.28	4.72
PRN [16]	2.75	3.51	4.61	3.62
SS-SFN (ours)	3.09	4.27	5.59	4.31
MS-SFN (ours)	2.91	3.83	4.94	3.89

Table 3: Comparison of NME(%) for 68 landmarks for AFLW2000-3D (divided into 3 groups based on yaw angles). 3DDFA2 refers to 3DDFA+SDM [61].

Method	Scenario 1	Scenario 2	Scenario 3
Yang et al. [51]	0.791	0.788	0.710
Xiao et al. [49]	0.760	0.782	0.695
CFSS [59]	0.784	0.783	0.713
MTCNN [56]	0.748	0.760	0.726
MHM [15]	0.847	0.838	0.769
MS-SFN (ours)	0.901	0.884	0.842

Table 4: Landmark localization performance of our method on videos in comparison to state-of-the-art face tracking methods. The values are reported in terms of Area under the Curve (AUC) for Cumulative Error Distribution of the 2D landmark error for 300VW test set.

rors compared to most of the methods that are dedicated for precise landmark localization. While PRN [16] has lower NME, its network size is 80 times bigger than ours and takes 9.8ms on GPU compared to <1ms required by our network. In addition to evaluations on static images, we also measure the face tracking performance in a video using our SFN. We set the bounding box of the current frame using the boundaries of the 2D landmarks detected in the previous frame and perform retargeting on a frame-by-frame basis. Table 4 compares the AUC values on 300VW dataset for three scenarios categorized by the dataset (compared under the same settings). Our method performs significantly better than other methods (about 9% improvement over the second best method for Scenario 3) with negligible failure rate because extensive data augmentation helps our tracking algorithm to quickly recover from failures.

5.3. Importance of Joint Training

Joint regression of both face bounding box locations and 3DMM parameters forces the network to learn exclusive facial features that characterize face shape, expression and pose in addition to differentiating face regions from the background. This helps in more precise face detection in-the-wild by leveraging both 2D information from bounding boxes and 3D information from 3DMM parameters. Table 2 shows that Average Precision (AP) is improved by a large margin with joint training compared to when the same network is trained to only regress bounding box locations. The retargeting accuracy for MFN is also comparable to that of SFN and the slight decrease in NME is because of training MFN on multi-face images and testing on single face images. Nevertheless, we observe improved performance in terms of both NME and AP by using better ground truth generated by multi-scale model. Our detection accuracy (mAP: 98.8%) outperforms Hyperface [32] (mAP: 97.9%) and Faceness-Net [52] (mAP: 97.2%) on the entire AFW dataset when compared under the same settings. Results of our MFN on multi-face images are illustrated in Fig. 6.

5.4. Evaluation of Expressions

Our expression evaluation results in Table 5 emphasize the improvement of multi-scale design on SFN. MS-MFN performs better than SS-SFN for all expressions except the eye expressions. This is because eye patches are small compared to the entire image for MFN whereas they are zoomed in on cropped images for SFN. Attention network for emphasizing small eye regions could be a future work for our MFN. However, MS-MFN shows less accuracy compared to MS-SFN because it is being tested on single face images while being trained on multi-face images. For the multi-person test set images, we found similar visual results by applying MS-MFN on the whole image and by applying MS-SFN on each face individually cropped from the image. This is expected because MFN is trained with ground truth from SFN. The performance of MS-SFN on our expression test set is shown in Fig. 7. The face shape fitting can be improved by using more landmarks or identity parameters, but we are limited by the available 3DMM and ground truth annotations. We also conducted live performance capture experiments to evaluate the efficiency our system in retargeting facial motion from face(s) to 3D character(s). Fig. 8 shows some screenshots recorded during the experiments.

5.5. Computational Complexity

Excluding the IO time, SFN can run at 15ms/frame on Google Pixel 2 (assuming single face and excluding face detector runtime). Face detection with our compressed detector model is 34ms, so separate face detection and retargeting requires 49ms for 1 face, 109ms for 5 faces and 184ms for 10 faces. On the other hand, our proposed MFN performs joint face detection and retargeting at 39ms on any number of faces. The model sizes for compressed face detector is 11.5MB and SFN is 2MB, so the combination is 13.5MB, while our MFN is only 13MB. Hence our joint network reduces both memory requirement and execution time.

6. Conclusion

We propose a lightweight multitask learning network for joint face detection and facial motion retargeting on mobile

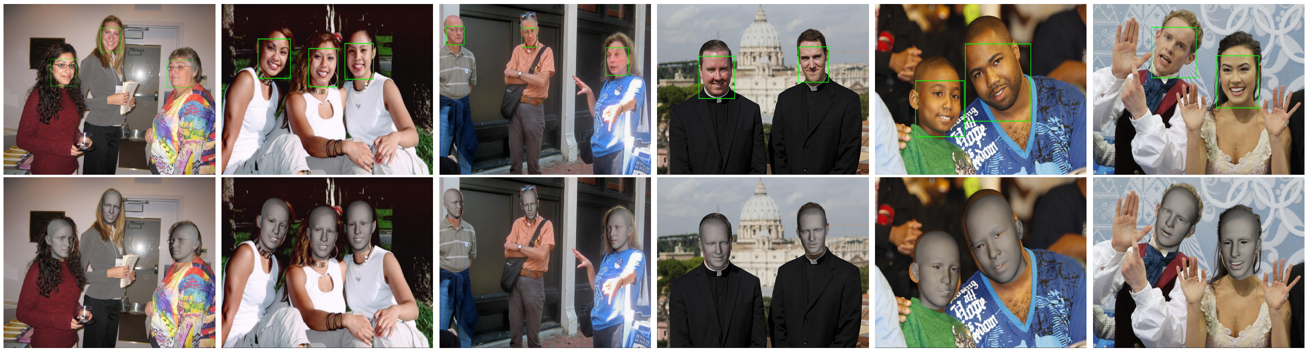

Figure 6: Testing results of our joint detection and retargeting model. Columns 1-3: Sampled from AFW; Columns 4-6: Sampled from WIDER. We show both the predicted bounding boxes and the 3D meshes constructed from 3DMM parameters.

Model	Eye Close	Eye Wide	Brow Raise	Brow Anger	Mouth Open	Jaw L/R	Lip Roll	Smile	Kiss	Avg
(1) Single scale SFN	0.082	0.265	0.36	0.451	0.373	0.331	0.359	0.223	0.299	0.305
(2) Multi-scale SFN	0.016	0.257	0.216	0.381	0.334	0.131	0.204	0.245	0.277	0.229
(3) MS-MFN + GT from (2)	0.117	0.407	0.284	0.405	0.284	0.173	0.325	0.248	0.349	0.288

Table 5: Quantitative evaluation of expression accuracy (measured by the expression metric) on our expression test set. Lower error means the model performs better for extreme expressions when required.

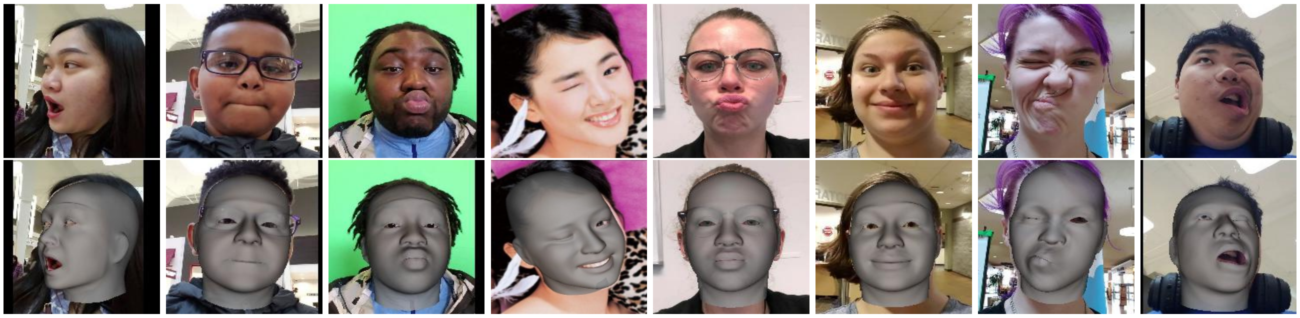

Figure 7: Results by applying MS-SFN on our expression test set.

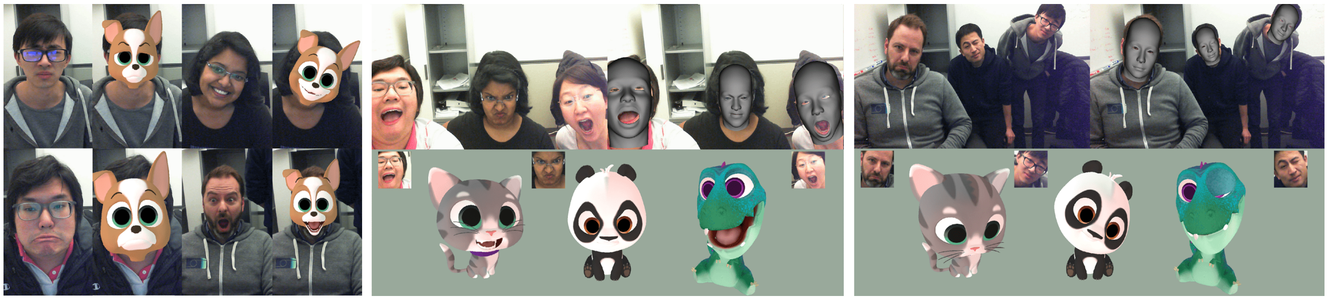

Figure 8: Retargeting from face(s) to 3D character(s).

devices in real time. The lack of 3DMM training data for multiple faces is tackled by generating weakly supervised ground truth from a network trained on images with single faces. We carefully design the network architecture and regularization to enforce disentangled representation learning inspired by key observations. Extensive results have demonstrated the effectiveness of our design.

Acknowledgements We thank the anonymous reviewers for their constructive feedback, Muscle Wu, Xiang Yan, Zeyu Chen and Pai Zhang for helping, and Linda Shapiro, Alex Colburn and Barbara Mones for valuable discussions.

References

[1] Faceshift. http://faceshift.com/.

[2] Faceware. http://facewaretech.com/.

[3] Deepali Aneja, Bindita Chaudhuri, Alex Colburn, Gary Faigin, Barbara Mones, and Linda Shapiro. Learning to generate 3D stylized character expressions from humans. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2018.

[4] Chandraskehar Bhagavatula, Chenchen Zhu, Khoa Luu, and Marios Savvides. Faster than real-time facial alignment: A 3D spatial transformer network approach in unconstrained poses. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[5] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3D faces. In *Proceedings SIGGRAPH*, pages 187–194, 1999.

[6] Sofien Bouaziz, Yangang Wang, and Mark Pauly. Online modeling for realtime facial animation. *ACM Transactions on Graphics*, 32(4), July 2013.

[7] Adrian Bulat and Georgios Tzimiropoulos. How far are we from solving the 2D & 3D face alignment problem? (and a dataset of 230,000 3D facial landmarks). In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[8] Chen Cao, Qiming Hou, and Kun Zhou. Displaced dynamic expression regression for real-time facial tracking and animation. *ACM Transactions on Graphics*, 33(4), July 2014.

[9] Chen Cao, Yanlin Weng, Stephen Lin, and Kun Zhou. 3D shape regression for real-time facial animation. *ACM Transactions on Graphics*, 32(4), July 2013.

[10] Chen Cao, Yanlin Weng, Shun Zhou, Yiying Tong, and Kun Zhou. Facewarehouse: A 3d facial expression database for visual computing. *IEEE Transactions on Visualization and Computer Graphics*, 20(3):413–425, March 2014.

[11] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2D pose estimation using part affinity fields. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[12] Dong Chen, Shaoqing Ren, Yichen Wei, Xudong Cao, and Jian Sun. Joint cascade face detection and alignment. In *European Conference on Computer Vision (ECCV)*, 2014.

[13] Nikolai Chinaev, Alexander Chigorin, and Ivan Laptev. Mobileface: 3D face reconstruction with efficient cnn regression. In *European Conference on Computer Vision Workshops (ECCVW)*, 2018.

[14] François Chollet et al. Keras. https://keras.io, 2015.

[15] Jiankang Deng, George Trigeorgis, Yuxiang Zhou, and Stefanos Zafeiriou. Joint multi-view face alignment in the wild. *arXiv preprint arXiv:1708.06023*, 2017.

[16] Yao Feng, Fan Wu, Xiaohu Shao, Yanfeng Wang, and Xi Zhou. Joint 3D face reconstruction and dense alignment with position map regression network. In *European Conference on Computer Vision (ECCV)*, 2018.

[17] Yao Feng, Fan Wu, Xiaohu Shao, Yanfeng Wang, and Xi Zhou. Joint 3D face reconstruction and dense alignment with position map regression network. In *European Conference on Computer Vision (ECCV)*, 2018.

[18] Yudong Guo, Juyong Zhang, Jianfei Cai, Boyi Jiang, and Jianmin Zheng. Cnn-based real-time dense face reconstruction with inverse-rendered photo-realistic face images. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2018.

[19] Yudong Guo, Juyong Zhang, Lin Cai, Jianfei Cai, and Jianmin Zheng. Self-supervised cnn for unconstrained 3D facial performance capture from a single RGB-D camera. *arXiv preprint arXiv:1808.05323*, 2018.

[20] Pei-Lun Hsieh, Chongyang Ma, Jihun Yu, and Hao Li. Unconstrained realtime facial performance capture. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[21] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[22] Gary B. Huang, Manu Ramesh, Tamara Berg, and Erik Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.

[23] Forrest N. Iandola, Song Han, Matthew W. Moskewicz, Khalid Ashraf, William J. Dally, and Kurt Keutzer. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and <0.5mb model size. *arXiv preprint arXiv:1602.07360*, 2016.

[24] Vidit Jain and Erik Learned-Miller. Fddb: A benchmark for face detection in unconstrained settings. Technical Report UM-CS-2010-009, University of Massachusetts, Amherst, 2010.

[25] Boyi Jiang, Juyong Zhang, Bailin Deng, Yudong Guo, and Ligang Liu. Deep face feature for face alignment and reconstruction. *arXiv preprint arXiv:1708.02721*, 2017.

[26] Amin Jourabloo and Xiaoming Liu. Large-pose face alignment via cnn-based dense 3D model fitting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[27] Vahid Kazemi and Josephine Sullivan. One millisecond face alignment with an ensemble of regression trees. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[28] Muhammed Kocabas, Salih Karagoz, and Emre Akbas. MultiPoseNet: Fast multi-person pose estimation using pose residual network. In *European Conference on Computer Vision (ECCV)*, 2018.

[29] Samuli Laine, Tero Karras, Timo Aila, Antti Herva, Shunsuke Saito, Ronald Yu, Hao Li, and Jaakko Lehtinen. Production-level facial performance capture using deep convolutional neural networks. In *Eurographics Symposium on Computer Animation*, 2017.

[30] Hao Li, Jihun Yu, Yuting Ye, and Chris Bregler. Realtime facial animation with on-the-fly correctives. *ACM Transactions on Graphics (Proceedings SIGGRAPH)*, 32(4), July 2013.

[31] Deva Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[32] Rajeev Ranjan, Vishal M. Patel, and Rama Chellappa. Hyperface: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition. *arXiv preprint arXiv:1603.01249*, 2016.

[33] Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[34] Joseph Redmon and Ali Farhadi. Yolo9000: Better, faster, stronger. *arXiv preprint arXiv:1612.08242*, 2016.

[35] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[36] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2015.

[37] Roger Blanco i Ribera, Eduard Zell, J. P. Lewis, Junyong Noh, and Mario Botsch. Facial retargeting with automatic range of motion alignment. *ACM Transactions on Graphics*, 36(4), 2017.

[38] Elad Richardson, Matan Sela, and Ron Kimmel. 3D face reconstruction by learning from synthetic data. In *International Conference on 3D Vision (3DV)*, 2016.

[39] Elad Richardson, Matan Sela, Roy Or-El, and Ron Kimmel. Learning detailed face reconstruction from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[40] Christos Sagonas, Georgios Tzimiropoulos, Stefanos Zafeiriou, and Maja Pantic. 300 faces in-the-wild challenge: The first facial landmark localization challenge. In *IEEE International Conference on Computer Vision Workshops (ICCVW)*, 2013.

[41] Jason M. Saragih, Simon Lucey, and Jeffrey F. Cohn. Realtime avatar animation from a single image. In *Face and Gesture*, 2011.

[42] Jie Shen, Stefanos Zafeiriou, Grigorios G. Chrysos, Jean Kossaifi, Georgios Tzimiropoulos, and Maja Pantic. The first facial landmark tracking in-the-wild challenge: Benchmark and results. In *IEEE International Conference on Computer Vision Workshops (ICCVW)*, 2015.

[43] Nikolai Smolyanskiy, Christian Huitema, Lin Liang, and Sean Eron Anderson. Real-time 3D face tracking based on active appearance model constrained by depth data. *Image and Vision Computing*, 32(11):860 – 869, 2014.

[44] Bugra Tekin, Sudipta N. Sinha, and Pascal Fua. Real-time seamless single shot 6d object pose prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[45] Ayush Tewari, Michael Zollhöfer, Hyeongwoo Kim, Pablo Garrido, Florian Bernard, Patrick Pèrez, and Christian Theobalt. MoFA: Model-based deep convolutional face autoencoder for unsupervised monocular reconstruction. In *IEEE International Conference on Computer Vision Workshops (ICCVW)*, 2017.

[46] Anh Tuan Tran, Tal Hassner, Iacopo Masi, and Gerard Medioni. Regressing robust and discriminative 3d morphable models with a very deep neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[47] Luan Tran and Xiaoming Liu. Nonlinear 3D face morphable model. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[48] Thibaut Weise, Sofien Bouaziz, Hao Li, and Mark Pauly. Realtime performance-based facial animation. In *SIGGRAPH*, 2011.

[49] Shengtao Xiao, Shuicheng Yan, and Ashraf A. Kassim. Facial landmark detection via progressive initialization. In *IEEE International Conference on Computer Vision Workshops (ICCVW)*, 2015.

[50] X. Xiong and F. De la Torre. Global supervised descent method. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[51] Jing Yang, Jiankang Deng, Kaihua Zhang, and Qingshan Liu. Facial shape tracking via spatio-temporal cascade shape regression. In *IEEE International Conference on Computer Vision Workshops (ICCVW)*, 2015.

[52] Shuo Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. From facial parts responses to face detection: A deep learning approach. In *IEEE International Conference on Computer Vision (ICCV)*, 2015.

[53] Shuo Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. Wider face: A face detection benchmark. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[54] Ronald Yu, Shunsuke Saito, Haoxiang Li, Duygu Ceylan, and Hao Li. Learning dense facial correspondences in unconstrained images. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[55] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10):1499–1503, Oct 2016.

[56] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10):1499–1503, Oct 2016.

[57] Shun Zhang, Yihong Gong, Jia-Bin Huang, Jongwoo Lim, Jinjun Wang, Narendra Ahuja, and Ming-Hsuan Yang. Tracking persons-of-interest via adaptive discriminative features. In *European Conference on Computer Vision (ECCV)*, 2016.

[58] Zhanpeng Zhang, Ping Luo, Chen Change Loy, and Xiaoou Tang. Facial landmark detection by deep multi-task learning. In *European Conference on Computer Vision (ECCV)*, 2014.

[59] Shizhan Zhu, Cheng Li, Chen Change Loy, and Xiaoou Tang. Face alignment by coarse-to-fine shape searching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[60] Xiangyu Zhu, Zhen Lei, Stan Z Li, et al. Face alignment in full pose range: A 3D total solution. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2017.

[61] Xiangyu Zhu, Zhen Lei, Xiaoming Liu, Hailin Shi, and Stan Z. Li. Face alignment across large poses: A 3D solution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.