IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received November 14, 2019, accepted December 1, 2019, date of publication December 17, 2019,
date of current version January 10, 2020.

Digital Object Identifier 10.1109/ACCESS.2019.2960325

Edge-Embedded Multi-Dropout Framework for Real-Time Face Alignment

GEE-SERN HSU[1], (Senior Member, IEEE), WEN-FONG HUANG[1], AND MOI HOON YAP[2]
[1]Department of Mechanical Engineering, National Taiwan University of Science and Technology, Taipei 10607, Taiwan
[2]School of Computing, Mathematics and Digital Technology, Manchester Metropolitan University, Manchester M15 6BH, U.K.

Corresponding author: Gee-Sern Hsu (jison@mail.ntust.edu.tw)

This work was supported in part by the MOST Taiwan under Grant 106-2221-E-011-144-MY3 and Grant 108-2923-E-011-003-MY3, and in part by the CPSI Center, Higher Education Sprout Project, MOE Taiwan.

ABSTRACT We propose the Edge-Embedded Multi-Dropout (EEMD) framework for real-time face alignment. The EEMD framework extracts facial edge features and explores multiple dropout architecture for locating facial landmarks. It consists of two major component networks, namely the Contour Detection Network (CDN) and the Multi-Dropout Network (MDN); and two supplementary networks, one for face detection and the other for pose regression. When a face is detected by the face detector, its pose will be classified by the pose classifier, then the associated facial edges be detected by the CDN, and then the landmarks be located by the MDN. The embedding of the CDN into the EEMD framework describes the observation that most landmarks are located on the contours/edges of the facial components and of the whole face. We revise a state-of-the-art edge detector as part of the base network for the CDN. The MDN is proposed to better design the regression architecture with appropriate dropout settings for better preventing overfitting and enhancing regression accuracy. Unlike most of the 2D approaches unable to locate landmarks in extreme poses, the proposed framework can detect landmarks on profile faces, i.e., $\pm 90°$ in yaw, *in real time*. Evaluated on benchmark databases, the EEMD demonstrates a competitive performance to other state-of-the-art approaches with a satisfying runtime speed.

INDEX TERMS Face alignment, facial landmark, dropout.

I. INTRODUCTION

Face alignment refers to the method that depicts the geometric structure of a face, including the contours of the face and of the facial components, such as eyes, nose and mouth. The depiction is generally performed by automatically locating the fiducial points, commonly known as facial landmarks, along the presumably unknown contours on a given facial image. The face alignment problem is therefore often cast as the detection of facial landmarks. The facial landmark detection must be accurate and robust to all poses, expressions and illumination variations.

Many approaches have been proposed in recent years [1]–[14]. Some of these approaches can only handle poses with yaw up to 45° [1], [2], and some recent methods can handle full pose, i.e., up to full profile [4], [7]–[14]. The methods able to locate full-pose landmarks can be split into 2D and 3D. Although part of the 2D landmarks can be considered as the 2D projection of the 3D landmarks, the *ground-truth* landmarks defined for the 2D and 3D approaches are different. The number of the 2D landmarks changes across pose, and it is common that more 2D landmarks are defined for yaw $<45°$ than those defined for yaw $\geq 45°$. However, the number of the 3D landmarks is a constant, and the landmarks can be split into visible and invisible (or self-occluded). The framework proposed in this paper is for detecting the full-pose 2D landmarks. In addition to landmark accuracy, we are also concerned about the runtime speed as an objective of this study is to develop a *real-time* face alignment solution.

We propose the Edge-Embedded Multi-Dropout (EEMD) framework for real-time full-pose face alignment. The architecture of the EEMD is shown in Figure 1. It consists of two major component networks, namely the Contour Detection Network (CDN) and the Multiple Dropout Network (MDN). Given a training image with landmarks annotated, we first connect the neighboring landmarks to form the landmarked edges, and define the binary landmarked edge image as the

The associate editor coordinating the review of this manuscript and approving it for publication was Bo Shen.

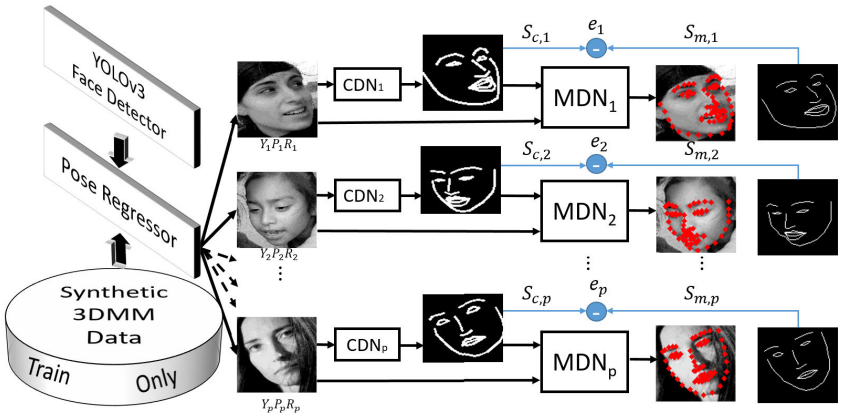

FIGURE 1. The proposed Edge-Embedded Multi-Dropout (EEMD) framework consists of two primary networks, namely the Contour Detection Network (CDN) and Multiple Dropout Network (MDN), and two supplementary networks, the (YOLO3) face detector and the (MDN) pose classifier.

output of the CDN with the given face image as input. The CDN is trained to estimate the landmarked edges for an input facial image. The trained CDN is then exploited as a preprocessor for the MDN. The MDN takes the facial image combined with the associated landmarked edge image, generated by the CDN, as the 2-channel input and the landmark locations as the output. The MDN has multiple dropouts implemented at convolutional layers, instead of the common fully-connected layers, to enhance the robustness against overfitting and improve the landmark accuracy.

The motivation to the design of the EEMD is the observation that most facial landmarks are located on the edges of the facial contour and facial components, for example, eyes, mouth and nose. The accuracy of the facial landmark detection can be improved if those edges are considered as part of the input and serve as a constraint for landmark regression. The network for landmark regression advances the Multi-Dropout Network (MDN) that we proposed in [15] with an additional loss function proposed in this paper. The study in [15] shows that multiple dropouts implemented at the convolution layers can better prevent overfitting than the regular way of a single dropout implemented at the fully-connected layers. Because of the overfitting prevention, the regression error can be further decreased, and in turn, improving the accuracy of the landmark localization. The recommended architecture has double dropouts implemented at the convolution layers.

In addition to the aforementioned CDN and MDN, the EEMD has two supplementary networks, the face detector and the pose classifier. The pose classifier aims at grouping the pose of a 2D face into similar orientations, so that the pose variation in each pose group can be constrained to a limited range. As the pose variation is reduced within each pose group, the pose-oriented landmark regressor can be better trained, and the landmarks can be more accurately located. The pose classifier is trained on the synthetic 300W-LPA data generated by the 3DMM (3-Dimensional Morphable Model). The pipeline of the EEMD works as follows: it first detects a face, then the pose classifier classifies the face into one of the pose clusters, then the pose-oriented CDN estimates the landmarked edge map, and then the pose-oriented MDN estimates the locations of the landmarks. The CDN and MDN work in series for each pose cluster, as shown in Figure 1.

This paper presents an extension to our previous work on the multiple dropout network [15] and the edge coupled framework [16] for facial landmark detection. In [15], we report the influences of different dropout settings on the performance of classification and regression networks. The settings that we have investigated include the number of dropout layers and the locations to implement dropout considered appropriate for stabilizing the training of a deep convolutional neural network (CNN). In [16], we report the combination of the multiple dropout network and an edge detection network which gives a preliminary version of the EEMD framework presented in this extended version. The extended parts include the following:

1) In [16], the loss function used for locating the facial landmarks is the common Euclidean distance between the predicted and ground-truth landmarks, which is known as the between-landmark loss. The loss considered in this extension adds in the L_1 distance between the CDN-generated contour S_c and the model-predicted contour S_m. This additional loss function considers the between-shape error and effectively improves the accuracy of the landmark localization.
2) In [16], the CDN follows the HED (Holistically-Nested Edge Detector) [17]. In this extension, we redesign the CDN based on the state-of-the-art edge detector, RCF (Richer Convolutional Feature) network [18], so that the inter-layer convolutional features can be explored for better extraction of the facial contour.
3) In [16], we used the 300W-LP [7] for training. In this extension, we augment the 300W-LP with pitch data and compose the 300W-LPA (Large Pose Augmented) dataset for training, which enhances the learning of

the multi-pose facial contour and landmark localization. Due to the additional pitch data included in the training set, the pose classifier in this extension is also redesigned accordingly, making it different from that in the previous work [16]. We have released the pitch-augmented 300W-LPA dataset with this paper.

The contributions of this paper can be summarized as follows.

1) It is verified that facial edges are important clues for locating facial landmarks. Justified on the benchmark databases, the EEMD framework that considers both the facial edges and RGB images demonstrates a competitive performance with a satisfying runtime speed among state-of-the-art approaches.
2) It is verified in this extended study that the loss function can be better designed to include both the between-landmark and between-shape losses, leading to better landmark accuracy than the common loss solely based on the between-landmark distance.
3) Multiple dropout architecture is better understood through this study with extensive experiments. It is verified that dual dropouts at the convolution layers can better prevent overfitting than the common way of implementing one single dropout at the fully-connected layers. Multiple dropouts have different influences on the CNN training phases for classification and regression, but this important issue has never been discussed until this study.
4) Cross-pose face alignment can be better solved by incorporating pose clustering that transforms a highly nonlinear regression problem into a set of weakly nonlinear regression problems. The EEMD framework explores a pose classifier to classify the pose of a given face, and then processes the face by the pose-oriented CDN and MDN networks for landmark detection. The pose-oriented CDN and MDN are benefited by faster training and lower validation loss due to the limited in-class variation in each pose cluster.

The contents of this paper are organized as follows. We first give a brief review in Sec. II. The development of the proposed EEMD framework is presented in Sec. III, followed by the training data preparation in Sec. IV. In Sec.V-B, we present the experimental evaluation of the EEMD framework. A conclusion is finally given in Sec.VI.

II. PREVIOUS WORK

An extensive review on the face alignment challenges, methods and databases can be found in [19]. We only highlight the recent (<6 years) approaches in this brief review. As described in Sec.I, we split the methods into 2D and 3D for better description on the approaches and performance.

A. 2D APPROACHES

The Supervised Descent Method (SDM) [1] minimizes a nonlinear least squares cost function formed by the initial and target landmark locations. The core part of SDM learns a sequence of descent directions that minimizes the mean of the cost functions sampled from the training set. The Regressing Local Binary Features (RLBF) [2] explores a better learning-based locality principle which learns the discriminant characteristics of local binary features, and achieves better accuracy than SDM. Although both SDM and RLBF attain high accuracy with good runtime speed, they can only handle pose up to 45° in yaw. To handle large-pose face alignment, Hsu et al. [4] propose the Regressive Tree Structured Model (RTSM), which is composed of a coarse TSM (c-TSM) and a refined TSM (r-TSM). The c-TSM works with fewer parts on a low-resolution image, and the r-TSM works with more parts on a high-resolution image. The c-TSM acts as a fast but coarse face detector that searches for facial candidates which are processed by the r-TSM for removing the false positives and locating the landmarks. It takes 0.7 sec to locate the landmarks on a face from the AFW database.

Many approaches have been built on the deep CNNs after 2013. The three-level cascade CNN [20] extracts the global features by the first level for initializing the landmark locations, and refines the initial predictions by the next two levels. However, it only locates 5 sparse landmarks within a face without considering any landmarks on the facial boundary/contour, and also works for limited poses (<45°) only. The Cascade Multi-Channel CNN (CMC-CNN) [5] locates the landmarks by performing bottom-up detection and top-down correction via a cascade of CNNs. Zhang et al. [6] propose the Tasks-Constrained Deep Convolutional Network (TCDCN), which not only learns the inter-task correlation but also employs the dynamic task coefficients to facilitate the multi-task optimization. Although the above CNN-centric approaches attain good accuracy and a few can work at high speed, they *cannot handle extreme poses*, i.e., >60° in yaw. A boundary-aware algorithm is proposed in [21] that uses stacked hourglass to estimate facial boundary heatmap and model the structure between facial boundaries through message passing for better robustness to occlusion. It is, however, unclear about how this approach handles large poses. Furthermore, although this approach also utilizes the facial contours to improve the landmark learning, the complex framework with a discriminator slows down the runtime (60 ms/image).

To detect full-pose landmarks, the HyperFace [9] fuses the intermediate layers of a deep CNN using a separate CNN followed by a multi-task learning algorithm that operates on the fused features. It exploits the synergy among multiple tasks which boosts up the individual performances, including the full-pose landmark localization. The study reported in [10] covers both 2D and 3D landmark localization using the Face Alignment Network (FAN) architecture built on four Hour-Glass (HG) networks. Three FANs are considered, the 2D-FAN, 3D-FAN, and 2D-to-3D FAN. The synthetic datasets 300W-LP-2D and 300W-LP-3D [22] are used to train the 2D-FAN and 3D-FAN, respectively, for locating the 2D and 3D landmarks. The 2D-to-3D FAN is trained on the 300W-LP with both 2D and 3D landmark annotations so that after training the network can convert the 2D landmark

annotations to 3D landmark annotations. Most of the above reviewed methods and other state-of-the-art algorithms are included in the performance comparison with the proposed approach reported in Sec. V-B3.

B. 3D APPROACHES

To handle large-pose face alignment, Zhu et al. propose the 3D Dense Face Alignment (3DDFA) that combines a cascaded CNN regressor and the 3D Morphable Model (3DMM), and formulates the alignment as 3DMM fitting problem [7]. The 3DMM fitting result can incorporate 2D landmark detectors to locate the landmarks. In the 3DDFA framework, the HOG features at landmarks are extracted to train a linear regressor to refine the landmark locations. To tackle the issue of limited landmark labeling for 3D dense alignment, Liu *et al.* [8] propose the Dense Face Alignment (DeFA) to employ contour and local feature constraints to the 3D dense fitting. These constraints are integrated into the CNN training as additional loss terms, and hence enhance the CNN for the 3D face fitting. Zhang et al. [11] propose the Joint Voxel and Coordinate Regression (JVCR) method for 3D landmark localization. The JVCR uses a volumetric representation to encode the per-voxel likelihood of landmark positions, and a stacked hourglass network to estimate the volumetric representation from coarse to fine, followed by a 3D convolution network that takes the estimated volume as input and regresses the 3D landmark coordinates. Deng et al. [12] propose the Cascade Multi-view Hourglass Model (CMHM) made of two Hourglass models for 3D face alignment. One Hourglass model aims to predict semi-frontal and profile 2D landmarks, the other is used to estimate the 3D facial shapes. A 3D reconstruction-based method with two multitask CNNs (MTCNNs) embedded is proposed in [13] for 3D shape and landmark estimation. One MTCNN handles pose estimation and 3D shape reconstruction and the other extracts the modified shape indexed features for more precise estimation of the 3D shape. The shape-aware heatmap is proposed in [14] for large-pose face alignment. The shape-aware heatmap is built on a Gaussian mixture model that considers adjacent landmarks to reconstruct the shape of local regions with a probability measure for the goodness of fit. Most of the above reviewed and other 3D approaches consider 3D landmark databases for validation, such as the AFLW2000-3D [7] and Menpo-3D [10], and only a few also consider 2D large-pose databases. The performance on the AFLW 2D database [23] reported in [7] and [8] is included in the performance comparison in Sec. V-B3.

III. PROPOSED APPROACH

The EEMD (Edge-Embedded Multi-Dropout) framework is shown in Figure 1. It has two major component networks, the CDN (Contour Detection Network) and MDN (Multiple Dropout Network) landmark regressor; and two supplementary component networks, the face detector and the MDN pose classifier. When a face is detected by the face detector, its pose will be classified by the MDN pose classifier, then its edges will be detected by the corresponding pose-oriented CDN, and then the landmarks will be located by the paired pose-oriented MDN landmark detector.

In the following, we first present the CDN (Contour Detection Network) in Sec. III-A, followed by the MDN (Multiple Dropout Network) in Sec. III-B, including both the MDN pose classifier and MDN landmark detector (or called the MDN landmark regressor). The supplementary face detector is briefed in Sec. III-C.

A. CONTOUR DETECTION NETWORK

The CDN network explores the state-of-the-art RCF (Richer Convolutional Feature) network [18] as the base net. We improve the RCF architecture with two modifications: 1) Shallower convolution layers for better scaled and leveled features and 2) Local window enhancement for extracting fine scaled features from component regions. The first modification with shallower convolution layers is due to the fact that our targets, including the eyes, mouth and the whole face, are in the same scale as the input is a face cropped by the face detector, instead of the multi-scaled objects considered in the general edge detection as in [18]. The second modification makes the network focus more on the components (or the regions of interests) and extracts features from up-scaled component regions.

As the RCF is the base net of our CDN, and the VGG-16 [24] is the base net of the RCF, the CDN can be better explained by looking into the architecture of the VGG-16. The VGG-16 demonstrates outstanding performance in various computer vision tasks, e.g., image classification [25] and face recognition [24]. It consists of two double-convolution blocks, three triple-convolution blocks and three fully-connected layers. The fully-connected layers are all removed in the RCF, and the five convolution blocks (called ConvBlocks 1∼5) with 13 convolution layers are kept. The convolution layers are commonly denoted as conv-1-1, conv-1-2, conv-2-1, ...conv-5-2 and conv-5-3, where conv-*i*-*k* denotes the *k*-th convolution layer in ConvBlock-*i*. A pooling layer with 2×2 window is implemented between the convolution blocks, as shown by the leftmost column of processors in Figure 2.

The modifications made by the RCF include the following: (1) Each conv layer is connected to a conv layer with kernel size 1×1 and channel depth 21 (denoted as $1 \times$ 1-21). The resulting layers in each block are accumulated using an *eltwise* layer to form hybrid features. (2) Each eltwise layer is connected to a $1 \times$ 1-1 conv layer, followed by a deconv layer for feature map upsampling. The deconv layer is connected to a sigmoid layer for minimizing the cross-entropy loss from the target. (3) All upsampling layers are concatenated and followed by a $1 \times$ 1-1 conv layer for fusing the feature maps from each block. The fused feature is connected to a sigmoid layer for minimizing the cross-entropy loss.

For building the CDN, we further modify the RCF network by (1) removing the 5th convolution block, i.e., ConvBlock 5, and (2) removing the last convolution

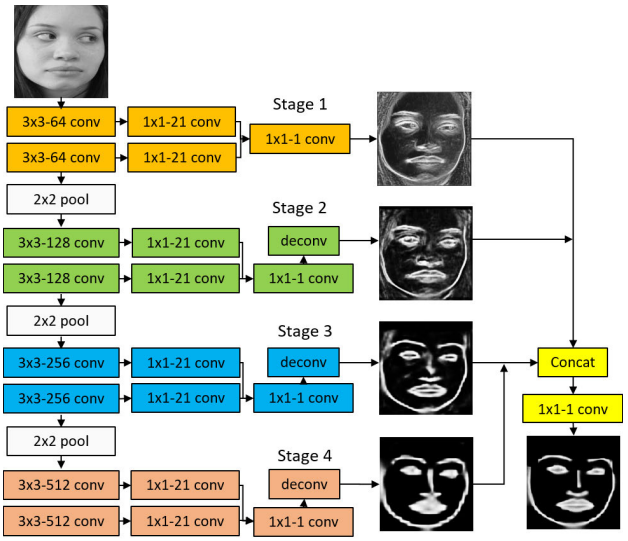

FIGURE 2. Architecture and settings of the proposed CDN (Contour Detection Network), the four stage edge outputs are extracted from ConvBlocks 1∼4.

layer in ConvBlocks 3 and 4, i.e., conv-4-3 and conv-5-3. Since edges are of high spatial frequencies which can be weakened by deeper convolution, removal of deeper convolution blocks and layers can better preserve such high frequency features. The CDN is therefore configured as the four double-convolution blocks shown in Figure 2.

To prepare the training data, we connect each neighboring landmark pair by a straight line, forming the landmarked contours over the face. The ground-truth edge probability y_i at pixel i is defined as 1 if it is on the contour, and 0 otherwise. Given the network with parameters lumped into the parameter vector W_c, the contour loss $L(X_i^k; W_c)$ considered in ConvBlock-k can be computed as follows:

$$L(X_i^k; W_c) = \begin{cases} \alpha \cdot \log(1 - P(X_i^k); W_c), & \text{if } y_i = 0 \\ \beta \cdot \log P(X_i^k; W_c), & \text{if } y_i = 1 \end{cases} \quad (1)$$

where X_i^k is the feature output extracted from ConvBlock-k, and

$$\alpha = \gamma \cdot \frac{|Y^+|}{|Y^+| + |Y^-|}, \quad \beta = \frac{|Y^-|}{|Y^+| + |Y^-|}$$

Y^+ and Y^- denote the set of edge pixels (as positive data) and the set of non-edge or background pixels (as negative data), respectively. γ is a hyper-parameter weight chosen to balance the positive and negative sets. $P(\cdot)$ is the sigmoid function. In most cases, the non-edge pixels substantially outnumber the edge pixels, i.e., $|Y^-| \gg |Y^+|$, making α and β weight the loss much more on the edge pixels.

Summing up the above loss from each ConvBlock with the fused loss contributed by all ConvBlocks, the total loss considered in the proposed CDN can be written as the following.

$$\mathcal{L}_T = \sum_{i=1}^{|I|} \left(\sum_{k=1}^{K} L(X_i^k; W_c) + L(X_i^{fuse}; W_c) \right) \quad (2)$$

where $|I|$ is the number of pixels in image I, K is the number of convolution blocks (4, in this case), and X_i^{fuse} is the concatenated output feature from all ConvBlocks.

B. MULTIPLE DROPOUT NETWORK

Consider a typical neural network of M layers with input $u^{(m)} = [u_i^{(m)}]_i$, output $v^{(m)} = [v_i^{(m)}]_i$ at Layer m, $m = 1, \ldots, M$, with the activation function $f(\cdot)$, the layer operation can be described as

$$u_i^{(m)} = w_i^{(m)} v^{(m-1)} \quad (3)$$

$$v_i^{(m)} = f(u_i^{(m)}) \quad (4)$$

where $w_i^{(m)}$ is the layer parameter vector that maps $v^{(m-1)}$ to $u_i^{(m)}$ ($v^{(0)}$ is the input to Layer 1). The activation $f(\cdot)$ can be followed by a max-pooling operation. With the dropout operation added in, (3) and (4) can be written as

$$\widehat{v}^{(m-1)} = q^{(m-1)} .* v^{(m-1)} \quad (5)$$

$$u_i^{(m)} = w_i^{(m)} \widehat{v}^{(m-1)} \quad (6)$$

$$v_i^{(m)} = f(u_i^{(m)}) \quad (7)$$

where $q^{(m-1)} = [q_j^{(m-1)}]_j$, $q_j^{(m-1)} \sim B(p)$, and $B(p)$ is the Bernoulli distribution with probability p of being 1; $.*$ denotes element-wise multiplication. According to the implementation in [26], the vector $q^{(m-1)}$ is first sampled and then multiplied element-wise with the output vector $v^{(m-1)}$ of the activation at Layer $m-1$ to create the *thinned* output $\widehat{v}^{(m-1)}$. The thinned output $\widehat{v}^{(m-1)}$ is then used as the input to Layer m. This process can be repeated at each layer, amounting to sampling a sub-network from a larger network. At the learning phase, the derivatives of the loss function are back-propagated through the sub-network. At the validation and testing phases, the weights are scaled as $\widehat{W}^{(m)} = pW^{(m)}$, and the network is exploited without the dropout. These computations are undertaken when we compute the test and validation errors of the proposed multiple dropout architectures with various dropout settings.

As our framework consists of a classifier, namely the MDN pose classifier, and a regressor, namely the MDN landmark detector, we are able to study the influences of implementing multiple dropouts on both types of networks. The structures of the MDN pose classifier and of the pose-oriented MDN landmark detector determined by this study are shown in Figure 3, with parameter settings in Table 1. The MDN pose classifier is made of three single-convolution blocks and three fully-connected layers. Each single-convolution block is composed of a convolution layer connected to a max-pooling layer. The dropouts are implemented at the last two convolution blocks, next to the max-pooling layers. The loss considered is the empirical softmax function. As we use a synthetic dataset for training which allows grouping of the training data according to preferred pose ranges, we have experimented 2 groups for yaw $\leq 45°$ and $>45°$, and 14 groups for both yaw and pitch variations considered. See Sections IV and V for details.

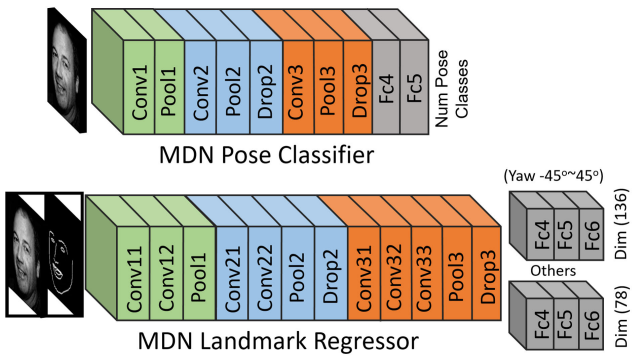

FIGURE 3. Top: The MDN pose classifier is trained to classify a face into one of the pose classes. Bottom: The MDN landmark regressor (or detector) is trained to locate the facial landmarks.

TABLE 1. Input, output and network settings for the MDN pose classifier and MDN landmark regressor.

Filter Size: 3, Stride: 1, Dropout Rate 0.5 for all				
Landmark Regressor			Pose Classifier	
Layer	Tag	Output Dim	Tag	Output Dim
0	input	2x96x96	input	1x96x96
1	conv-1-1	64x94x94	conv-1	32x94x94
2	conv-1-2	128x92x92	pool-1	32x47x47
3	pool-1	128x46x46	conv-2	64x46x46
4	conv-2	64x44x44	pool-2	64x23x23
5	conv-2-2	128x42x42	**dropout**2	64x23x23
6	pool-2	128x21x21	conv-3	128x22x22
7	**dropout**2	128x21x21	pool-3	128x11x11
8	conv-3-1	128x19x19	**dropout**3	128x11x11
9	conv-3-2	128x17x17	fc4	512
10	conv-3-3	128x15x15	fc5	512
11	pool-3	128x7x7	output	16
12	**dropout**3	128x7x7	-	-
13	fc-4	512	-	-
14	fc-5	512	-	-
14	fc-6	512	-	-
15	output	136 or 78	-	-

The pose-oriented MDN landmark detector is made of two double-convolution blocks, one triple-convolution block and three fully-connected layers. The dropouts are implemented at the second double-convolution block and at the triple-convolution block, also next to the max-pooling layers. Note that the output dimension of the MDN landmark detector is 136 for yaw $\leq$ 45° and 78 for $>$ 45°, i.e., for 68 and 39 landmarks, respectively. Denoting N as the number of landmarks, the loss considered in the MDN landmark detector is the Normalized Mean Error (NME) between the predicted landmarks $L_i^{(p)}$ and the ground-truth landmarks $L_i^{(g)}$, normalized to the size of the ground-truth bounding box. The NME is known as the between-landmark loss $\mathcal{L}_l$, computed as follows:

$$\mathcal{L}_l = \frac{1}{N}\sum_{i=1}^{N}\frac{||L_i^{(g)} - L_i^{(p)}||_2}{d} \tag{8}$$

where $d = \sqrt{h_b \cdot w_b}$ is the size of the ground-truth bounding box, computed as the square root of the area $h_b \cdot w_b$.

Both of the above MDN pose classifier and MDN landmark detector networks are determined from extensive experiments for comparing the training stability and network performance with different ways of applying the dropouts, including different numbers of dropouts and different locations for implementing the dropouts. Note that in contrary to the general way of implementing dropouts at fully-connected layers, our study reveals that *dropouts can be better implemented at convolution layers, especially for regression networks*. Additionally, our experiments also show that multiple dropouts are required for better stabilizing the training of the landmark regression network, and can improve the stability when training the pose classifier network. See the experiments in Sec. V for more details.

The between-landmark loss $\mathcal{L}_l$ in (8) is considered with the 2-group pose classifier when we were experimenting for the determination of the most appropriate settings for the multiple dropout network, and when we were locating the landmarks in our previous work [15]. To further improve the landmark accuracy, in this extended study we also consider the between-shape (or shape-to-shape) loss $\mathcal{L}_s$. Given a training image I_j, $\mathcal{L}_s$ accounts for the difference between $S_{m,j}$, the contour formed by the estimated landmarks, and $S_{c,j}$, the contour rendered by the CDN, denoted by e_j as shown in Figure1. e_j is computed as the L_1 norm between $S_{m,j}$ and $S_{c,j}$, i.e.,

$$e_j = ||S_{m,j} - S_{c,j}||_1 = \sum_{i}^{|I|}|s_{m,j}^i - s_{c,j}^i| \tag{9}$$

C. FACE DETECTOR

For face detection, we revise the YOLO3 detector [27] and train it on the WIDER FACE database [28] that offers 393,703 labeled faces in 32,203 images with a large variability in pose, illumination, expression, scale and occlusion. Following the data partition specified in [28], the WIDER FACE is split into a training and validation set with 199k faces in 16,106 images and a test set with 194k faces in 16,097 images. Instead of following the default settings of YOLO3, we customize the anchor boxes with the facial bboxs of the training set so that we can enhance the performance. Compared with the state-of-the-art approaches on the AFW database [29], the YOLO3 achieves AP (Average Precision) 99.6%, better than the DPM (97.2%) [30], the HeadHunter (97.1%) [30] and the Faster RCNN (95.3%) [31]. Note that the Faster RCNN is proposed in [31] for object detection, we retrained it for face detection the same way as we did for the YOLO3.

IV. TRAINING DATA GENERATION

To train the MDN pose classifier and landmark detectors, a large dataset is needed with each sample labeled in pose and landmarks. Such a database collected *in the wild* is hardly available so far [29], [32]. However, the face profiling approach [7] that takes a nearly frontal face as an input and

TABLE 2. Number of images in each pose group.

Pitch \ Yaw	-90° ∼ −70°	-70° ∼ −45°	-45° ∼ −15°	-15° ∼ 15°	15° ∼ 45°	45° ∼ 70°	70° ∼ 90°
0° ∼+20°	10782	14319	16580	9891	16580	14319	10782
0° ∼-20°	9719	12751	13870	9305	13870	12751	9719

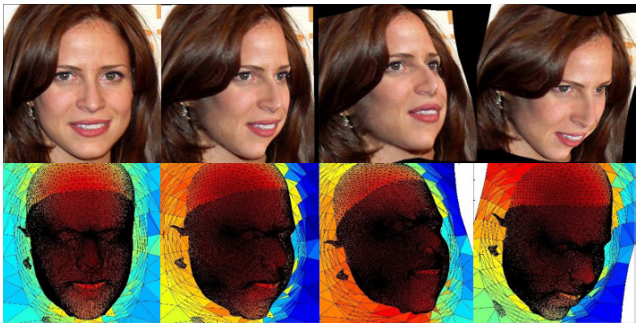

FIGURE 4. Face profiling with the original placed on the bottom left. The top row shows the rotated faces and background, the bottom row shows the 3DMM fitted face with 3D meshed background.

generates its rotated counterparts offers an effective solution to this issue.

The core part of the face profiling is the 3D image meshing on a given 2D face and its background. The 3D meshing begins with a 3D Morphable Model (3DMM) fitted to the 2D face by following the Multi-Features Fitting (MFF) [33]. This approach can be directly applied to faces labeled with landmarks. Given a landmark labeled 2D face, the MMF fitting will be appropriately constrained by the landmarks and deliver a well fitted 3D model. To include expression variation to the 3DMM, Zhu et al. [22] combine the identity shape from the Basel Face Model (BFM) [34] and the expression shape from the Face Warehouse [35]. In addition, they propose the *landmark marching* technique for fitting the 3DMM to a face with pose variation, allowing an accurate estimate of the pose for the face in terms of orientations in yaw, pitch and roll. The 3D facial model and its 3D meshed background form a 3D object that can be rotated to a specified orientation. We use the code downloaded via the link provided in [22].

Using the face profiling approach, Zhu et al. offer the 300W-LP (Large-Pose) dataset that contains faces made from the 300W database [32] with each face rotated *in yaw* at 5° each step up to 90° [7]. On top of this dataset, we augment it with additional data made with rotation *in pitch* with the same 5° step up to ±20°. We call this dataset the 300W-LPA (Large-Pose Augmented). Figure 4 shows a sample with its 3D meshed face and background model rotated in yaw and pitch, and the rendered 2D images. To show the differences between the original 300W, 300W-LP and our augmented 300W-LPA databases, Figure 5 shows 3 subjects from the 300W original dataset, the yaw-augmented samples, and our pitch-augmented samples. As the pitch augmentation is performed for different yaw, we only show one subject instead of all three. To obtain the 300W-LPA, please visit https://sites.google.com/view/300w-lpa-database for details.

As the faces generated by 3DMM face profiling are tagged with 3D pose, we classify them into 7 intervals in yaw: $[-90°, -70°]$, $[-70°, -45°]$, $[-45°, -15°]$, $[-15°, 15°]$, $[15°, 45°]$, $[45°, 70°]$ and $[70°, 90°]$; and further into 2 intervals in pitch in each yaw interval, namely chin-up ($-20° \sim 0°$) and chin-down ($0° \sim 20°$). Note that the yaw classes are the horizontal mirrors of the other sides, e.g., $[-90°, -70°]$ is the mirror of $[70°, 90°]$. Therefore, the data in the 300W-LPA are actually grouped into 4 yaw classes, in which 3 yaw classes are flipped to the horizontal mirrors, forming 7 yaw classes in total. It gives 14 pose classes with the additional split in pitch. The number of images in each group is given in Table 2. An ablation study reported in Sec.V-B2 reveals that the landmark accuracy improves when the number of pose classes increases. More pose classes make the pose variation in one pose class decreased, reducing the difficulty for landmark regression and thus improving the accuracy. See Sec.V-B2 for more details.

V. EXPERIMENTS

The experiments are split into two parts. The first part aims to determine the most appropriate settings for multiple dropout network, including the location(s) and number of dropouts to better stabilize training. Without loss of generality, we only consider a simplified network with the 2-group MDN pose classifier and MDN landmark regressor, as shown in Figure 3, i.e., without the CDN in the framework. The second part aims to demonstrate the strength of embedding the CDN into the EEMD framework, study the influences of using the 300W-LPA and with different numbers of pose classes, and compare with other contemporary approaches. The NME (Normalized Mean Error) in (8) is used to compute the landmark error.

Two benchmark databases are considered in our evaluation, 300-W [32] and AFLW [23]. The 300-W dose not offer data with large/extreme poses, but the AFLW does. The AFLW can be further split into different pose ranges in yaw to better understand the performance for large poses. Because the median pose samples in the AFW [29] are merged to the 300W training set, the rest data with large poses are used as testing set in our experiments. To clarify the datasets used for training and testing, Table 3 gives the settings for evaluating the MDN pose classifier and the proposed EEMD landmark detector. All experiments were run on a Ubuntu 14.04 with Titan X GPU, and CUDA 7.5 with cuDNN 4.0 on Caffe.

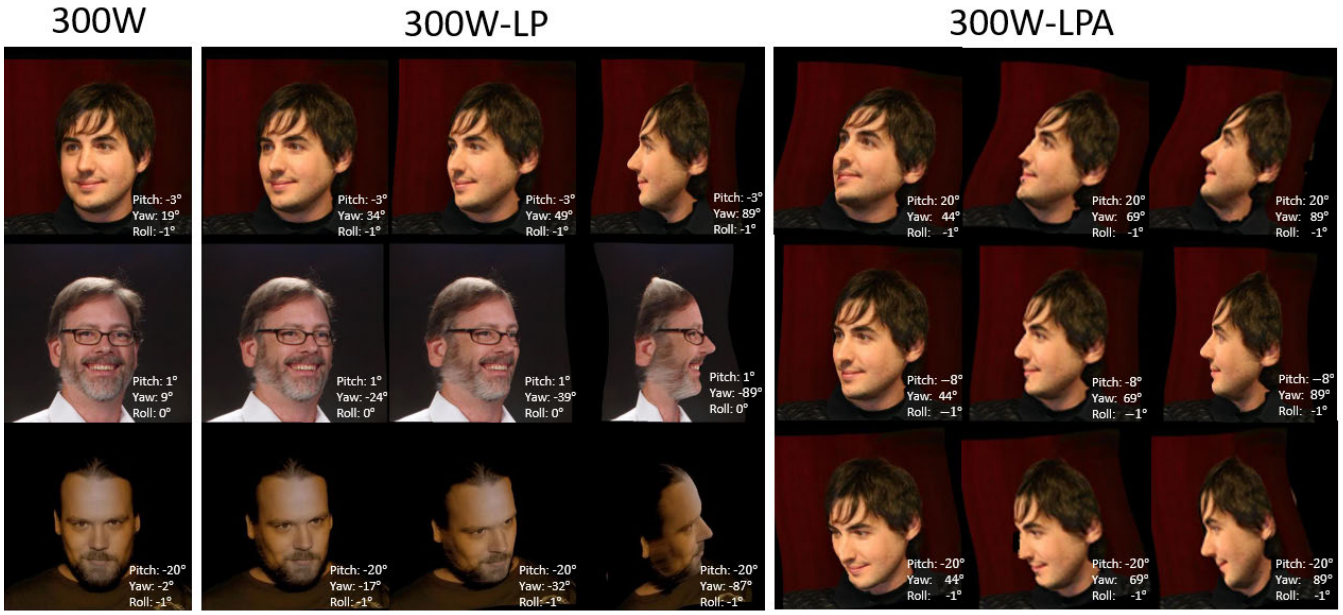

FIGURE 5. Data generated by using 3DMM face profiling. Left: the original 300W; Center: 300W-LP from [7]; Right: 300W-LPA (augmented with pitch samples added to 300W-LP, proposed in this paper).

TABLE 3. Training and testing setups.

	Training	Testing
MDN Pose Classifier	300W-LPA Training Set	300W-LPA Test Set
CDN + MDN Landmark Detector	300W-LPA Training Set	AFLW, 300W Test Set
300W-LPA training and testing splits follow the original 300W splits		

A. ARCHITECTURE OF MULTI-DROPOUT NETWORK (MDN)

The network considered in this section is a simplified network without the contour detection module, as the MDN landmark regressor shown in Figure 3 without the contour image input. When training the MDN landmark detectors, we adopt a two-phase scheme. We train the network for locating a sparse set of landmarks in Phase 1, and then *fine tune* the last fully-connected layer for locating the desired dense set of landmarks in Phase 2.

In Phase 1, we train the frontal-pose MDN landmark detector using the Multi-Task Facial Landmark (MTFL) dataset [36] for 5-landmark localization with 5 location outputs. The MTFL contains 10,000 face images labeled with 5 landmarks each, including two pupils, nose and two corners of the mouth. In Phase 2, the network is fine tuned by training on the 300W training set for locating the 68 landmarks. We augment different reception field images to overcome deviation of face detection by random shifting and scaling. When training the profile-pose MDN landmark detector, we also train the model for locating a sparse set of 5 landmarks first (eye corner, mouth corner, eyebrow corner, nose tip and chin), and fine tune it for the dense set of 39 landmarks.

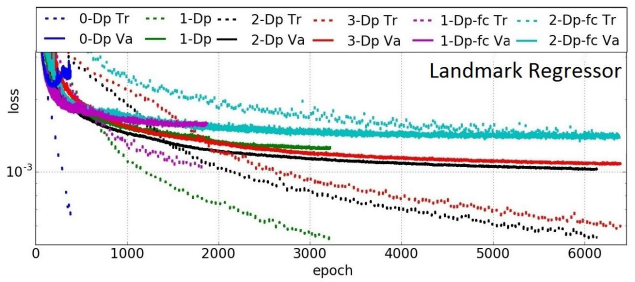

FIGURE 6. Training and validation errors of the MDN landmark detector in Figure3 with different numbers of dropouts. Tr denotes training error and Va is validation error. 0-Dp refers to no dropout, 1-Dp is one dropout only next to Pool3, 2-Dp is 1-Dp with one more dropout next to Pool2, 3-Dp is 2-Dp with one more dropout next to Pool1, 1-Dp-fc is one dropout only next to Fc5, and 2-Dp-fc is 1-Dp-fc with one more dropout added next to Fc4.

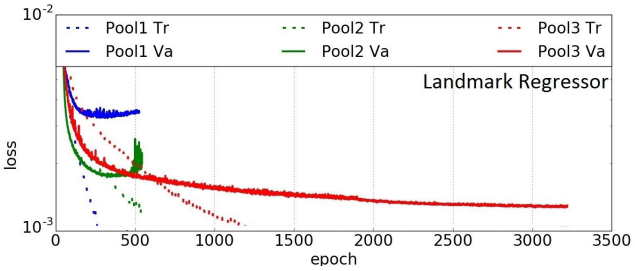

FIGURE 7. Training and validation errors of the MDN landmark detector (or regressor) with one dropout implemented next to different pooling layers. Pool1 (Pool2, Pool3) is the dropout applied next to "Pool1" ("Pool2", "Pool3").

Figure 6 shows the training and validation errors of the MDN landmark detector with different numbers of dropouts applied at different layers. The experiments are based on the aforementioned Phase 1 setup on the MTFL dataset. 0-Dp

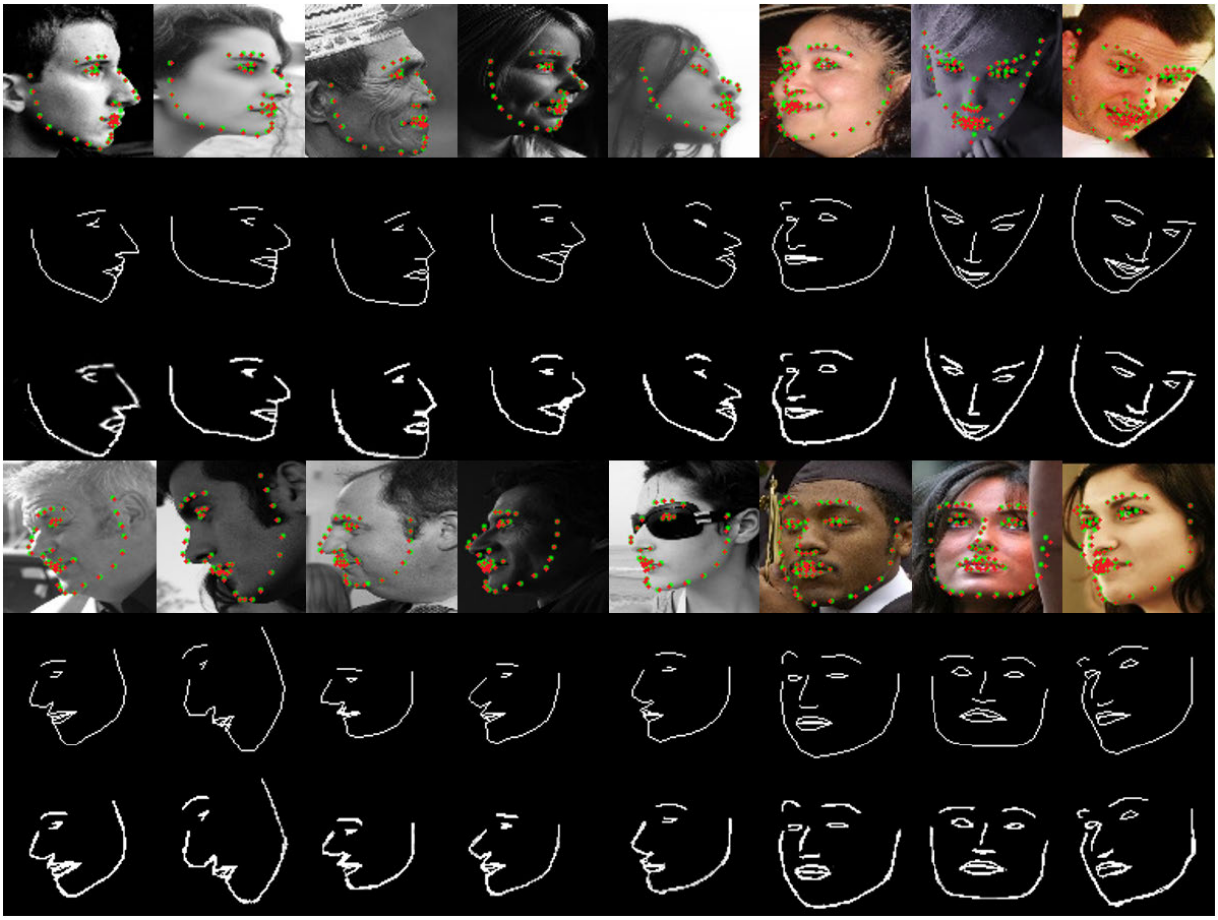

FIGURE 8. Top and the 4th rows: the landmarks located by the EEMD network denoted in red and ground truth in green; the 2nd and 5th rows: the ground-truth landmarked contours; the 3rd and bottom rows: the CDN estimated contours.

denotes the network without dropout. 1-Dp, 2-Dp and 3-Dp denote respectively the networks with one to three dropouts connected to the max-pooling layers in Figure 3. 1-Dp-fc and 2-Dp-fc denote the networks with one dropout implemented next to the fully-connected layer Fc5 and two dropouts next to Fc4 and Fc5, respectively. The tendency toward overfit can be evaluated by the gap between the training and validation errors as the epoch increases. We have applied a stopping criterion that the training ends when the validation error no longer decreases for more than 200 epochs (otherwise, the training for each setup would have taken much longer time to complete). The 0-Dp shows severe overfitting as the training error decreases rapidly while the validation error oscillates within a clear bound, and all take place shortly after the training begins. 2-Dp and 3-Dp perform almost equally well, and both appear better than 1-Dp in terms of the stabilized gap and the validation error at large epochs. 1-Dp-fc appears better than 2-Dp-fc, as the latter shows some tendency toward underfit, which refers to the observation that the validation error decreases faster and lower than the training error. When underfit takes place, it can be very difficult to further reduce the validation error.

In addition to the comparison on different numbers of dropouts in Figure 6, the comparison on a dropout implemented at different layers is shown in Figure 7. Pool*n* refers to a dropout implemented next to the layer "Pool*n*" in the MDN landmark detector. Apparently, Pool3 is the best and Pool1 is the worse, showing that dropout is better implemented relatively deep in the network.

B. PERFORMANCE OF EDGE-EMBEDDED MULTI-DROPOUT NETWORK

1) FACIAL CONTOUR DETECTION

We followed most of the parameter settings, including the receptive field sizes, in the RCF [18] for setting up the contour detection module CDN. Figure 8 shows the facial contours obtained by the CDN (the 3rd and bottom rows), the ground-truth landmarked contours (the 2nd and 5th rows) and the landmarks detected (the top and 4th rows) in red with the ground-truth landmarks in green. The CDN estimated contours on the face and facial components are all close to the ground-truth contours. As most of the detected landmarks overlap with the ground truth, we compute the

TABLE 4. Landmark accuracy (in % NME) for different settings on the pose classes and training databases (300W-LP v.s. 300W-LPA). *k*p refers to the additional pitch grouping to the *k* yaw classes.

Trained on	No. PCs	300-W Test			AFLW Test				
		Common Set	Challenge Set	Full Set	[0, 30]	[30, 60]	[60, 90]	Mean	Std
300W-LPA	3	4.81	9.45	5.68	4.05	4.53	6.09	4.89	0.87
	6 (3p)	4.72	9.34	5.58	3.93	4.38	5.81	4.66	1.06
	5	4.72	9.26	5.52	3.91	4.40	5.73	4.60	0.81
	10 (5p)	4.62	8.98	5.40	3.81	4.29	5.57	4.48	0.81
	7	4.65	8.97	5.45	3.85	4.25	5.34	4.45	0.93
	14 (7p)	4.51	8.72	5.29	3.79	4.14	5.15	4.29	0.74
300W-LP	3	4.97	9.81	5.91	4.18	4.72	6.26	5.15	1.16
	5	4.92	9.63	5.79	4.08	4.68	6.21	5.11	0.81
	7	4.89	9.47	5.72	4.02	4.64	6.17	5.07	1.11

TABLE 5. Landmark accuracy (in NME %) comparison with SOTA approaches. The best three in each category column are in **boldface**. PCs stands for Pose Classes.

	300-W			AFLW					Speed
Method	Common Set	Challenge Set	Full Set	[0, 30]	[30, 60]	[60, 90]	Mean	Std	Time(s)
CFSS [3]	**4.73**	9.98	5.76	-	-	-	-	-	0.040
RTSM [4]	6.02	16.52	8.06	7.97	12.67	16.31	12.32	4.18	0.250
CMC-CNN [5]	4.91	12.03	6.30	-	-	-	-	-	0.150
RLBF [2]	4.95	11.98	6.32	4.68	-	-	-	-	**0.003**
TCDCN [6]	**4.80**	**8.60**	**5.54**	-	-	-	-	-	-
3DDFA [7]	6.15	10.59	7.01	5.00	5.06	6.74	5.60	0.99	0.076
3DDFA-SDM [7]	5.53	9.56	6.31	4.75	4.83	6.38	5.32	0.92	0.105
DeFA [8]	5.37	**9.38**	6.10	-	-	-	-	-	-
HyperFace [9]	-	-	-	**3.93**	**4.14**	**4.71**	**4.26**	**0.41**	0.200
MDN [15]	4.95	10.52	6.01	4.57	5.01	7.17	5.58	1.39	**0.016**
ERN [16]	4.71	9.81	5.94	4.32	4.79	6.97	5.36	1.42	**0.034**
EEMD (3 PCs)	4.81	9.45	**5.68**	**4.05**	**4.53**	**6.09**	**4.89**	**0.87**	**0.038**
EEMD (14 PCs)	**4.51**	**8.72**	**5.29**	**3.79**	**4.14**	**5.15**	**4.29**	**0.74**	**0.038**

landmark location errors and compare with other contemporary approaches in the next section.

2) ABLATION STUDY ON TRAINING DATABASES AND POSE CLASSES

To better understand the effectiveness of the pitch-augmented 300W-LPA and the influence of different pose classes, we compare the performance of the same network trained on the 300W-LP dataset, and the performance with various numbers of pose classes. This comparison study is designed with the following settings:

1) As the 300W-LPA contains both the yaw and pitch augmented data, the pose classes (PCs) are primarily defined in yaw, and each yaw class can be further split in pitch. We first segment the data into 3, 5 and 7 PCs in yaw, and each is further split into 2 pitch groups (chin up and down) denoted as 3p, 5p and 7p, associated with 6, 10 and 14 PCs, respectively.
   a) 3 PCs: $[-90°, 45°]$, $[-45°, 45°]$ and $[45°, 90°]$ in yaw;
   b) 6 PCs (3p): Additional split of each PC in the above a) into 2 pitch classes $[20°, 0°]$ and $[0°, -20°]$ in pitch, 3p refers to the additional pitch grouping to the 3 yaw classes;
   c) 5 PCs: $[-90°,-70°]$, $[-70°,-45°]$, $[-45°, 45°]$, $[45°, 70°]$ and $[70°, 90°]$ in yaw;
   d) 10 PCs (5p): Additional split of each PC in c) into 2 pitch classes in the same way as in b);
   e) 7 PCs: $[-90°,-70°]$, $[-70°,-45°]$, $[-45°,-15°]$, $[-15°, 15°]$, $[15°, 45°]$, $[45°, 70°]$ and $[70°, 90°]$ in yaw;
   f) 14 PCs (7p): Additional split of each PC in e) into 2 pitch classes in the same way as in b);
2) The 300W-LP is the 300W-LPA without the pitch-augmented data, and can thus be grouped in yaw only, i.e., the data in 300W-LP can only be grouped into 3, 5 and 7 PCs.

The following observations are based on the experimental results given in Table 4.

- The additional pitch-augmented data in the 300W-LPA helps to improve the accuracy, shown by all three (3, 5, 7) PCs, compared with the training on 300W-LP.
- More PCs yields better landmark accuracy, which verifies the proposition that a reduced pose variation within a PC improves the regression for the landmark location.

3) FACIAL LANDMARK LOCALIZATION

Table 5 shows the normalized landmark localization error of the proposed approach, along with the errors of other state-of-the-art approaches. The errors of the selected approaches are either directly copied from their publications or obtained by running the codes provided by the authors. Those without

numbers and shown in "−" are either not available in the publications or unable to handle by their methods/codes. For example, the code for the RLBF [2] can only deal with poses ≤45°, and fails when handling larger poses. In each column, the best three are shown in **boldface**.

Note the four closely related approaches in the last four rows, namely our previous approaches MDN [15] and ERN [16], the proposed EEMD with two pose classes (3 PCs) and the EEMD with 14 pose classes (14 PCs). The EEMD outperforms the ERN [16] for two reasons:

- A better contour detector built on the more advanced RCF architecture;
- Both the between-landmark and between-shape losses are considered in the EEMD framework.

These upgraded versions also highlight the advantages of contour embedding architecture, i.e., how advantageous the CDN can contribute to the landmark accuracy. The EEMD (14 PCs), obtained by using the training set with 14 pose groups (described in Sec. IV), along with the ablation study reported in Sec. V-B2, demonstrates the advantages of using more pose classes.

For comparison purpose, we experimented with the facial contour as the only input to the MDN, i.e., without the RGB image in Fig. 1, in contrast to the EEMD and the previous MDN [15], which considers the RGB image only without the facial contour as input. Tested on the 300W with 3 PCs, we have obtained 7.78%, 14.21% and 10.11% NME on the common set, the challenging set and the full set, respectively. As the accuracy is far from comparable to those attained by the EEMD and MDN, the experiment was not extended to the AFLW dataset. This comparison shows that the facial contour can be considered as an effective clue to improve the landmark localization; however, it is not appropriate to use it alone for locating the facial landmarks.

The proposed EEMD (14 PCs) shows a highly competitive performance to other state-of-the-art methods. When tested on the 300-W common set, the EEMD outperforms all selected methods. When tested on the 300-W challenge set, it outperforms most except the TCDCN [6] with a small margin. As the data in the 300-W common set are more than those in the challenge set, the performance on the full set stays at the best. The comparison on the AFLW dataset also shows the effectiveness of the EEMD. For the yaw range [0°∼30°], the EEMD (14 PCs) outperforms all; for [30°∼60°], the EEMD (14 PCs) and HyperFace [9] both perform equally well; but for [60°∼90°], it is slightly outperformed by the HyperFace. This reveals that the EEMD can be further improved by incorporating the multi-task learning network with fused features as those exploited by the HyperFace. However, there is a price to pay for the complex multi-task learning network, it takes 200 ms for the HyperFace to process a face. As our target solution must be able to meet the real-time requirement, the HyperFace is therefore not considered as a potential candidate. Figures 9 and 10 show the

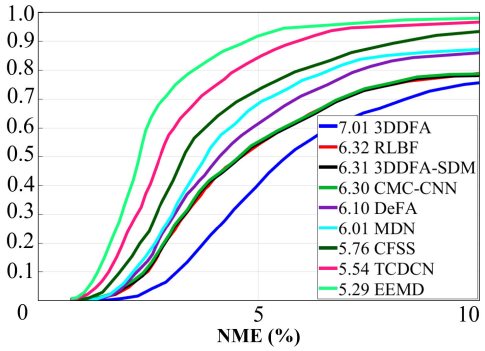

FIGURE 9. Comparison with contemporary approaches in NME vs. the fraction of the test faces on 300-W.

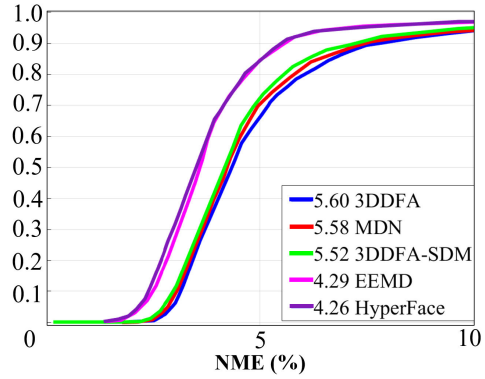

FIGURE 10. Comparison with contemporary approaches in NME vs. the fraction of the test faces on AFLW.

comparisons with other contemporary approaches in terms of the NME versus the fraction of test faces on 300-W and AFLW databases, respectively. Both figures demonstrate the same performances as shown in Table 5 but from a different perspective.

Note the performance improvement from the previous MDN [15] to the EEMD (3 PCs), and then to the EEMD (14 PCs). The improvement made by the facial contour detection and CDN edge embedding appears more significant than that made by incorporating more pose groups in the framework. The experiments have verified that the facial landmark localization can be better solved by considering the contour features in the learning phase. More pose groups make the shape variation in each group more constrained, and therefore make the localization easier for the regression network to handle. An attractive merit of the EEMD framework is the 18 ms/face processing speed, contributed by the simple network structure adopted. Compared with 76 ms using the 3DDFA [7], 150 ms using the CMC-CNN [5] and 200 ms using the Hyperface [9], the EEMD can be considered as one of the fastest and most effective approaches for face alignment across large poses.

VI. CONCLUSION

Most landmarks are located on specific contours/edges, and the proposed EEMD framework can be one of the pioneer

works that consider this observation. The EEMD framework consists of four components: 1) the YOLO3 face detector, 2) the MDN pose classifier, 3) the CDN contour detector and 4) the MDN landmark detectors. The CDN contour detector is modified from the RCF edge detector with modifications on the design of the convolution blocks and layers. With the hierarchical features extracted from selected convolution blocks, the CDN can generate edges in close similarity to the targeted landmarked edges. The MDN architecture is better understood from this study. It reveals that multiple dropouts can better stabilize the training, and dropouts can be better implemented at convolution layers, instead of at the common fully-connected layers, especially for the regression network. Extensive experiments show that the EEMD can be an effective and competitive real-time solution for face alignment. We would consider the fused features from multi-task learning using simplified architecture to warrant the runtime speed in the continuing phase of this research.

REFERENCES

[1] X. Xiong and F. De la Torre, "Supervised descent method and its applications to face alignment," in *Proc. CVPR*, Jun. 2013, pp. 532–539.

[2] S. Ren, X. Cao, Y. Wei, and J. Sun, "Face alignment at 3000 fps via regressing local binary features," in *Proc. CVPR*, Jun. 2014, pp. 1685–1692.

[3] S. Zhu, C. Li, C. Change Loy, and X. Tang, "Face alignment by coarse-to-fine shape searching," in *Proc. CVPR*, Jun. 2015, pp. 4998–5006.

[4] G.-S. J. Hsu, K.-H. Chang, and S.-C. Huang, "Regressive tree structured model for facial landmark localization," in *Proc. ICCV*, Dec. 2015, pp. 3855–3861.

[5] Q. Hou, J. Wang, L. Cheng, and Y. Gong, "Facial landmark detection via cascade multi-channel convolutional neural network," in *Proc. ICIP*, Sep. 2015, pp. 1800–1804.

[6] Z. Zhang, P. Luo, C. C. Loy, and X. Tang, "Learning deep representation for face alignment with auxiliary attributes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 5, pp. 918–930, May 2016.

[7] X. Zhu, Z. Lei, X. Liu, H. Shi, and S. Z. Li, "Face alignment across large poses: A 3D solution," in *Proc. CVPR*, Jun. 2016, pp. 146–155.

[8] Y. Liu, A. Jourabloo, W. Ren, and X. Liu, "Dense face alignment," in *Proc. ICCV Workshop*, Oct. 2017, pp. 1619–1628.

[9] 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 Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 1, pp. 121–135, Jan. 2017.

[10] A. Bulat and G. Tzimiropoulos, "How far are we from solving the 2D & 3D face alignment problem? (and a dataset of 230,000 3D facial landmarks)," in *Proc. ICCV*, Oct. 2017, pp. 1021–1030.

[11] H. Zhang, Q. Li, and Z. Sun, "Joint voxel and coordinate regression for accurate 3D facial landmark localization," in *Proc. ICPR*, Aug. 2018, pp. 2202–2208.

[12] J. Deng, Y. Zhou, S. Cheng, and S. Zaferiou, "Cascade multi-view hourglass model for robust 3D face alignment," in *Proc. 13th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, May 2018, pp. 399–403.

[13] G. Zhang, H. Han, S. Shan, X. Song, and X. Chen, "Face alignment across large pose via MT-CNN based 3D shape reconstruction," in *Proc. FG*, May 2018, pp. 210–217.

[14] J. Si, F. Jiang, and R. Shen, "Large-pose face alignment via shape-aware heatmap," in *Proc. ICASSP*, May 2019, pp. 3037–3041.

[15] G.-S. Hsu and C.-H. Hsieh, "Cross-pose landmark localization using multi-dropout framework," in *Proc. IJCB*, Oct. 2017, pp. 390–396.

[16] G.-S. Hsu and K.-C. Ho, "Edge-coupled and multi-dropout face alignment," in *Proc. ICIP*, Oct. 2018, pp. 2645–2649.

[17] S. Xie and Z. Tu, "Holistically-nested edge detection," in *Proc. CVPR*, Dec. 2015, pp. 1395–1403.

[18] Y. Liu, M.-M. Cheng, X. Hu, K. Wang, and X. Bai, "Richer convolutional features for edge detection," in *Proc. CVPR*, Jul. 2017, pp. 5872–5881.

[19] C. Sagonas, E. Antonakos, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "300 faces in-the-wild challenge: Database and results," *Image Vis. Comput.*, vol. 47, pp. 3–18, Mar. 2016.

[20] Y. Sun, X. Wang, and X. Tang, "Deep convolutional network cascade for facial point detection," in *Proc. CVPR*, Jun. 2013, pp. 3476–3483.

[21] W. Wu, C. Qian, S. Yang, Q. Wang, Y. Cai, and Q. Zhou, "Look at boundary: A boundary-aware face alignment algorithm," in *Proc. CVPR*, Jun. 2018, pp. 2129–2138.

[22] X. Zhu, Z. Lei, J. Yan, D. Yi, and S. Z. Li, "High-fidelity pose and expression normalization for face recognition in the wild," in *Proc. CVPR*, Jun. 2015, pp. 787–796.

[23] M. Köstinger, P. Wohlhart, P. M. Roth, and H. Bischof, "Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization," in *Proc. ICCVW*, Nov. 2011, pp. 2144–2151.

[24] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *Proc. BMVC*, Sep. 2015, p. 6.

[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2012, pp. 1097–1105.

[26] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[27] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," Apr. 2018, *arXiv:1804.02767*. [Online]. Available: https://arxiv.org/abs/1804.02767

[28] S. Yang, P. Luo, C. C. Loy, and X. Tang, "WIDER FACE: A face detection benchmark," in *Proc. CVPR*, Jun. 2016, pp. 5525–5533.

[29] X. Zhu and D. Ramanan, "Face detection, pose estimation, and landmark localization in the wild," in *Proc. CVPR*, Jun. 2012, pp. 2879–2886.

[30] M. Mathias, R. Benenson, M. Pedersoli, and L. Van Gool, "Face detection without bells and whistles," in *Proc. ECCV*, 2014, pp. 720–735.

[31] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst. (ANIPS)*, 2015, pp. 91–99.

[32] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "300 faces in-the-wild challenge: The first facial landmark localization challenge," in *Proc. CVPRW*, Jun. 2013, pp. 397–403.

[33] S. Romdhani and T. Vetter, "Estimating 3D shape and texture using pixel intensity, edges, specular highlights, texture constraints and a prior," in *Proc. CVPR*, vol. 2, Jun. 2005, pp. 986–993.

[34] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter, "A 3D face model for pose and illumination invariant face recognition," in *Proc. AVSS*, Sep. 2009, pp. 296–301.

[35] C. Cao, Y. Weng, S. Zhou, Y. Tong, and K. Zhou, "FaceWarehouse: A 3D facial expression database for visual computing," *IEEE Trans. Vis. Comput. Graphics*, vol. 20, no. 3, pp. 413–425, Mar. 2014.

[36] Z. Zhang, P. Luo, C. C. Loy, and X. Tang, "Facial landmark detection by deep multi-task learning," in *Proc. ECCV*. Zürich, Switzerland: Springer, 2014, pp. 94–108.

GEE-SERN (JISON) HSU (M'07–SM'14) received the dual M.S. degree in electrical and mechanical engineering and the Ph.D. degree in mechanical engineering from the University of Michigan, Ann Arbor, in 1993 and 1995, respectively. From 1995 to 1996, he was a Postdoctoral Fellow with the University of Michigan. From 1997 to 2000, he was a Senior Research Staff with the National University of Singapore. In 2001, he joined Penpower Technology, where he led research on face recognition and intelligent video surveillance. His team at Penpower Technology was a recipient of the Best Innovation and Best Product Award at the SecuTech Expo for three consecutive years from 2005 to 2007. In 2007, he joined the Department of Mechanical Engineering, National Taiwan University of Science and Technology (NTUST), where he is currently an Associate Professor. His research interests are computer vision and pattern recognition. He received the best paper awards in ICMT 2011, CVGIP 2013, CVPRW2014, ARIS 2017, and CVGIP 2018. He is a Senior Member of IAPR.

WEN-FONG HUANG received the B.S. degree in mechanical engineering from the National Kaohsiung University of Science and Technology, Kaohsiung, Taiwan, in 2016, and the M.S. degree in mechanical engineering from the National Taiwan University of Science and Technology, Taipei, Taiwan, in 2018. He is currently an Algorithm R&D Engineer at Lite-On Technology Corporation. His research interest focuses on deep learning and computer vision.

MOI HOON YAP received the Ph.D. degree in computer science from Loughborough University, Loughborough, U.K., in 2009. She is currently a Reader (Associate Professor) of computer vision with the Manchester Metropolitan University, Manchester, U.K., and a Royal Society Industry Fellow with Image Metrics Ltd., Manchester, U.K. After her Ph.D., she was a Postdoctoral Research Assistant with the Centre for Visual Computing, the University of Bradford, from April 2009 to October 2011. Her research expertise is computervision, machine learning, and deep learning. She was an Associate Editor of the *Journal of Open Research Software* and a reviewer of the IEEE TRANSACTIONS ON IMAGE PROCESSING, the IEEE TRANSACTIONS ON MULTIMEDIA, the IEEE TRANSACTIONS ON CYBERNETICS, and the IEEE JOURNAL OF BIOMEDICAL HEALTH AND INFORMATICS.