Unsupervised Facial Action Unit Intensity Estimation via Differentiable Optimization

Xinhui Song  
NetEase Fuxi AI Lab  
songxinhui@corp.netease.com

Tianyang Shi  
NetEase Fuxi AI Lab  
shitianyang@corp.netease.com

Tianjia Shao  
Zhejiang University  
tianjiashao@gmail.com

Yi Yuan  
NetEase Fuxi AI Lab  
yuanyi@corp.netease.com

Zunlei Feng  
Zhejiang University  
zunleifeng@zju.edu.cn

Changjie Fan  
NetEase Fuxi AI Lab  
fanchangjie@corp.netease.com

Abstract

The automatic intensity estimation of facial action units (AUs) from a single image plays a vital role in facial analysis systems. One big challenge for data-driven AU intensity estimation is the lack of sufficient AU label data. Due to the fact that AU annotation requires strong domain expertise, it is expensive to construct an extensive database to learn deep models. The limited number of labeled AUs as well as identity differences and pose variations further increases the estimation difficulties. Considering all these difficulties, we propose an unsupervised framework GE-Net for facial AU intensity estimation from a single image, without requiring any annotated AU data. Our framework performs differentiable optimization, which iteratively updates the facial parameters (i.e., head pose, AU parameters and identity parameters) to match the input image. GE-Net consists of two modules: a generator and a feature extractor. The generator learns to “render” a face image from a set of facial parameters in a differentiable way, and the feature extractor extracts deep features for measuring the similarity of the rendered image and input real image. After the two modules are trained and fixed, the framework searches optimal facial parameters by minimizing the differences of the extracted features between the rendered image and the input image. Experimental results demonstrate that our method can achieve state-of-the-art results compared with existing methods.

1. Introduction

Facial expression analysis is of great interest to many researchers in computer vision, computer graphics, and psychology. Automated facial expression analysis from a single image enables numerous applications such as human-robot interaction and behavioral and psychological research. The Facial Action Unit (AU) intensity estimation \cite{ekman1978} is one of the main building blocks in single-image facial expression analysis, which aims to estimate anatomically meaningful parameters (i.e., muscle movements) instead of PCA parameters of a 3D morphable model \cite{blanz1999}. The seminal work of Ekman and Friesen \cite{ekman1978} develops a Facial Action Coding System (FACS) for describing facial expressions, which are produced by the movements of facial muscles under the skin. Nearly any anatomically possible facial expression can be coded by a combination of AUs.

Existing AU intensity estimation methods are broadly divided into fully supervised methods \cite{lu2016} \cite{pantic2004} \cite{zhang2016} and weakly supervised methods \cite{zhou2012} \cite{zhou2014} \cite{zhou2015}. Training a fully supervised model requires a large set of labeled samples. However, AU annotation requires strong domain expertise, resulting in very high time-and-labor costs to construct a large database. Due to this demanding labeling process, datasets in the literatures (e.g., CK+ \cite{guo2018}, MMI \cite{holmes2003}, AM-FED \cite{li2018}, DISFA \cite{li2016}, BP4D \cite{zhang2018}) restrict the number of coded AUs, samples, and subjects. The weakly supervised methods focus on various types of human knowledge with limited annotation datasets. Nevertheless, they still require thousands of AU annotations, and the human defined knowledge restricts the space of anatomically plausible AUs. Besides, there are many challenges in automatic AU intensity estimation so that the performance of AU intensity estimation is often unsatisfactory. These challenges include large variations in pose, identity, illumination, and occlusion in the unconstrained environment. Among them, pose and identity have always been essential challenges because they can dramatically increase variances of face images of different people even when they have the same expression.

Considering the above challenges, we propose a framework for unsupervised estimation of facial action unit in-
tensity from a single image, which can jointly estimate the facial parameters (i.e., head pose, AU parameters and identity parameters) without annotated AU data. The core of our framework is differentiable optimization, which iteratively updates the facial parameters to match the input image. It is achieved with the help of a novel network architecture called GE-Net. GE-Net consists of two modules: a generator which learns to “render” a face image from a set of facial parameters in a differentiable way, and a feature extractor which extracts deep features for measuring the similarity of the rendered image and input real image. After the two modules are trained, we integrate them into our framework, and the facial parameters can be fitted by minimizing the loss of the extracted features between the rendered image and the input real one.

Specifically, our generator is trained based on a 3D parametric face model called Justice Face from an online game Justice[1]. The model parameters have explicit physical meanings, including AU parameters and identity parameters. The AU parameters can express all the AUs from FACS, and the identity parameters describe identity-specific face shapes by controlling the attributes of facial components, including position, orientation, and scale, as introduced in the work [37]. We adopt a face semantic segmentation network as our feature extractor, whose intermediate feature maps focus more on the facial shape contents rather than the raw appearance. Hence it can effectively reduce the large domain gap between the rendered image and the real one (see Sec. 4.3 for details).

Our method can be used in many applications, one of which is expression transfer. We can exchange the AU parameters of different individuals, which is shown in Fig. 1. With image A or B as input, our framework can predict their facial parameters. After re-combining the head pose, AU parameters, and identity parameters, we can transfer the pose and expression from B to A while keeping A’s identity.

Our main contributions are summarized as the following: To the best of our knowledge, we firstly propose a deep learning framework for unsupervised estimating the facial AU intensity from a single image, without requiring annotated AU data. What’s more, the novel framework enables differentiable optimization, which consists of a generator performing differentiable rendering and a feature extractor measuring similarity of the rendered image and real image. Large experiments show that the proposed method achieves state-of-the-art results on the AU intensity estimation. Meanwhile, our method can activate more AUs than existing AU annotated databases, such as jaw left and upper lip raiser.

2. Related Work

2.1. Face Reconstruction Methods based on 3DMM

In the past years, many methods [57, 56, 14, 54] describe a 3D facial shape using the parameters of the 3DMM [4], which are constructed with PCA. Early methods [58, 5] use a regressor to directly regress the facial landmark locations from a single image for fitting 3DMM parameters. Recently, the approaches of regressing 3DMM parameters using CNNs and fitting 3DMM to the 2D images have become popular. Jourabloo et al. [20] use cascade CNN regressors to regress the 3DMM parameters (identity, expression and pose parameters). Bindita et al. [8] present a single end-to-end network to jointly predict the bounding box locations and 3DMM parameters for multiple faces. Zhu et al. [56, 59] perform multiple iterations of a single CNN to fit the 3DMM parameters. Different from these 3DMM-based methods, our goal is to predict the facial AU intensity with explicit anatomical meanings, while the 3DMM parameters lack such physical meanings.

2.2. AU intensity Estimation Methods

Similar to AU detection [33], some methods [21, 23, 22] use traditional machine learning frameworks to estimate AU intensity. Recently, some works apply CNNs for AU intensity estimation. Tadas et al. [1] present a facial action

---

1https://n.163.com/
unit intensity estimation system based on geometry features (shape parameters and landmark locations) and appearance (Histograms of Oriented Gradients). The median-based feature normalization is used to account for a person-specific neutral expression. Li et al. [26, 27] design different CNNs to regress AU parameters, where the CNNs focus on different facial regions independently. Batista et al. [8] and Zhou et al. [55] apply DNNs to estimate AU intensity under multiple head poses. However, these supervised methods require a large number of training samples, while AU annotation needs strong domain expertise, and it requires great efforts to construct such extensive database. Besides, limited by the number of annotated samples, above methods usually suffer from an over-fitting problem. Due to the lack of AU annotation datasets, several works [32, 47, 50] use the weakly supervised training paradigm to train deep models with incomplete or inaccurate annotations, where prior knowledge and domain knowledge are involved. However, these methods still require thousands of annotations, and they add prior knowledge, which restricts the space of anatomically plausible AUs. In contrast, our method can estimate more AUs than existing AU annotated databases, without requiring any annotated data.

### 2.3. Neural Render

Recent works on differentiable rendering achieve differentiability in various ways. Kato et al. [24] achieve differentiability by an approximate gradient for the rasterization operation. GQN [11] shows a powerful neural renderer, which learns to represent synthetic scenes using the input images of a scene taken from different viewpoints, but the geometry of objects in the scene is simple. MOFA [40] learns a decoder using a neural renderer model, which is an expert-designed generative model. The expert-designed generative model is difficult to describe the complicated rendering process with the illumination condition and complex shading explicitly. Nguyen-Phuoc et al. [32] propose a CNN architecture to render 3D objects from a 3D voxel grid. Shi et al. [37] learn a generator as a neural render with fixed camera parameters. Inspired by [37], we extend the generator to imitate the face rendering process from meaningful facial parameters including identity, expression and head pose.

### 3. Method

Our goal is to estimate the intensity of facial AUs as well as the face pose and identity from a single image via differentiable optimization, without requiring any labeled data. The optimization framework GE-Net consists of a generator $M$ and a feature extractor $E$. The generator $M$ imitates the rendering process from the facial parameters $y_g$ (i.e., head pose $h$, AU parameters $\alpha_{exp}$ and identity parameters $\alpha_{id}$) to the rendered image $I_{yg}$, while the feature extractor $E$ extracts deep features from images, which are used to measure the similarity between the real input image and the rendered image for fitting the meaningful parameters. The generator is trained with the help of a 3D parametric face model called Justice Face from the game Justice. The model parameters have explicit physical meanings, including AU parameters $\alpha_{exp}$ and identity parameters $\alpha_{id}$. A complete process of our method is summarized as follows:

**Stage 1.** Train the generator $M$ and the feature extractor $E$.

**Stage 2.** Fix the parameters of $M$ and $E$, and iteratively update the facial parameters $(h, \alpha_{exp}, \alpha_{id})$ according to the loss between the input image and the generated one. Empirically, the facial parameters can closely describe the input image after 30 iterations.

### 3.1. 3D Face Model

The parametric face model Justice Face contains the identity parameters and AU parameters. The identity parameters $\alpha_{id}$ control the attributes of facial components.
to create a neutral face for a specific identity, including the position, orientation, and scale as recommended by the method [37]. The AU parameters $\alpha_{\text{exp}}$ control the expression by driving the facial muscle movements relative to the neutral face. Specifically, these AU parameters include eyes closed, upper lid raiser, cheek raiser, inner brows raiser, outer brows raiser, brow lower, jaw open, nose wrinkle, upper lip raiser, down lip downer, lip corner pull, mouth press, outer brows raiser, brow lower, jaw open, nose wrinkle, upper lid raiser, cheek raiser, inner brows raiser, lip pucker, upper lips close, lower lips close, cheek puff, lip corner depressor, jaw left and jaw right. The corresponding maps between AUs and our AU parameters are shown in Table 2 of the appendix. We can adjust the AU and identity parameters to generate an identity-specific expression, which is formulated below:

$$S = S + \alpha_{id} A_{id} + \alpha_{\text{exp}} A_{\text{exp}},$$

where $S$ is a synthesized 3D face and $S$ is the base face. $A_{id}$ is the facial component offsets to the base face in position, orientation, and scale, and the identity parameters $\alpha_{id}$ are the weights of these offsets. $A_{\text{exp}}$ is the offsets of facial muscle movements relative to the neutral face. The AU parameters $\alpha_{\text{exp}}$ are the corresponding weights and represent the facial AU intensity.

### 3.2. Generator

The generator $\mathcal{M} : (y_g) \rightarrow I_g$ learns to map a set of meaningful facial parameters $y_g = (h, \alpha_{\text{exp}}, \alpha_{id})$ to a rendered image $I_g$, based on the Justice Face. $y_g$ is a 269-dimensional vector with head pose $h \in \mathbb{R}^{1 \times 2}$, AU parameters $\alpha_{\text{exp}} \in \mathbb{R}^{1 \times 23}$ and identity parameters $\alpha_{id} \in \mathbb{R}^{1 \times 244}$. The ground truth $I_{yr} \in \mathbb{R}^{H \times W \times 3}$ is the corresponding rendered image from the Justice Face, and the predicted one is marked as the generated image $I_g$. Our generator $\mathcal{M}(y_g)$ consists of eight transposed convolution layers, which is similar to the network of DCGAN [35]. We use instance normalization [41] instead of batch normalization [18] in the generator for improving training stability. The generator is differentiable so that the facial parameters can be updated by gradient descent.

Fig. 3(a) shows the architecture of our generator. To make the generated images indistinguishable from the rendered images, we adopt a content loss $\mathcal{L}_{\text{app}}$ and a perceptual loss $\mathcal{L}_{\text{per}}$ between the rendered image $I_{yr}$ and the predicted one $I_g$. The content loss $\mathcal{L}_{\text{app}}(y_g)$ penalizes large changes in the raw pixel space, and we use $l_1$ loss function rather than $l_2$ for $\mathcal{L}_{\text{app}}$ so as to encourage less blurring. The content loss is defined as:

$$\mathcal{L}_{\text{app}}(y_g) = E_{y_g \sim u(y_g)}[\|I_{yr} - \hat{I}_g\|_1],$$

where $\hat{I}_g = \mathcal{M}(y_g)$ and $u(y_g)$ means the uniform distribution. The input parameters $y_g$ are sampled from a multidimensional joint uniform distribution $u(y_g)$.

The perceptual loss $\mathcal{L}_{\text{per}}(y_g)$ is added to explicitly constrain the generated result to have the same content information as the rendered one, and it is defined as:

$$\mathcal{L}_{\text{per}}(y_g) = E_{y_g \sim u(y_g)}[\|F(\hat{I}_g) - F(I_{yr})\|_2],$$

where $F$ denotes the relu2$_{\cdot}$2, relu3$_{\cdot}$3, relu4$_{\cdot}$3 feature maps in VGG16 [39], which is pre-trained on the image recognition task. In this way, we can train the generator $\mathcal{M}$ by optimizing the following loss function:

$$\mathcal{L}_{\mathcal{M}}(y_g) = \mathcal{L}_{\text{app}}(y_g) + \lambda \mathcal{L}_{\text{per}}(y_g),$$

where $\lambda$ balances the multiple objectives.

During training, all the rendered images are aligned via five facial landmarks using affine transformation, including
left eye, right eye, nose, left mouth corner and right mouth corner, which are obtained by OpenFace [2, 45]. The convolution kernel size in the generator is $4 \times 4$, and the stride of each transposed convolution layer is 2. The Instance-Normalization and ReLU activation are embedded in $\mathcal{M}$ after every transpose convolution layer, except for its output layers. $\mathcal{M}$ is trained with mini-batch stochastic gradient descent (SGD) and the mini-batch size is 64. All the weights are initialized from a zero-centered normal distribution with a standard deviation of 0.02. The learning rate is 0.001, and the learning rate decay is set to 1% per three epochs. After training of 500 epochs, the generator successfully learns how to map each facial parameter to its corresponding component/muscle movement and “render” a visually plausible image.

### 3.3. Feature Extractor

As there is a large domain gap between the generated image and the input real image, to effectively measure their similarity, a face segmentation network is trained as a feature extractor $\mathcal{E}$ in Fig. 3(b). The similarity is computed on the feature maps of the first four layers of $\mathcal{E}$. Specifically, an attention mechanism [9] is deployed in the feature extractor. It allows the feature maps to attend to different parts of the face, which means the feature maps have different influences on the loss. The output probability maps of the facial segmentation network serve as the attention masks, which are represented as the pixel-wise weights $A$ in Eqn. (5). Specifically, the probability map of every facial local region is utilized as an attention mask of each feature map. For example, the feature map of the first layer is sensitive to the eyes, and the eyes’ probability map $A$ is element-wisely multiplied to the feature map of the first layer. Suppose $C$ represents the feature map, and the extracted features can be defined as:

$$\mathcal{E}(I) = \sum_{i=0}^{N} \beta_i A_i(I) C_i(I),$$  \hspace{1cm} \text{(5)}

where $I$ is the input of the feature extractor. $N$ feature maps are extracted, and $\beta$ represents the weight of each feature map $C_i$. We set different attention masks $A_i$ for the feature maps by the probability maps of the facial segmentation result.

The architecture of the feature extractor network is based on Resnet-50 [16]. We set the stride $[2, 1, 1, 1]$ at four sub-layers, remove the fully connected layers and change the average pooling layer to a $1 \times 1$ convolution layer. The output is 1/8 size of the input. We further perform up-sampling with factor 8 to get an upsampled output with the same size as the input. The network is pre-trained on the ImageNet [10] and is fine-tuned with the Helen Dataset [25] with the pixel-wise cross-entropy loss. The learning rate is set to 0.001.

### 3.4. Fitting

After the generator and the feature extractor are learned, we fix their parameters and iteratively optimize the facial parameters (i.e., head pose, AU parameters and identity parameters) to match the input real image. To fit the facial parameters to a real face image with expression, the generator $\mathcal{M}$ and the feature extractor $\mathcal{E}$ serve as a bridge. We optimize the facial parameters by measuring the similarity of the extracted features from $\mathcal{E}$ between the input real image and the generated image from $\mathcal{M}$. The extracted features can teach the generator to produce faces that match firmly with the real image in the feature space.

Fig. 2 shows the pipeline of our proposed method. The meaningful facial parameters can be updated using gradients because the renderer is differentiable. We update the facial parameters by back-propagating the loss on extracted features, which explicitly constrain the “rendered” result to have similar facial shape contents as the real image. The loss is defined as:

$$\mathcal{L}_E(y_g, I_{y_r}) = \|\mathcal{E}(\mathcal{M}(y_g)) - \mathcal{E}(I_{y_r})\|_1,$$ \hspace{1cm} \text{(6)}

where $I_{y_r}$ is the input real image aligned with the base face using five facial landmarks. The parameters are updated with the gradient descent method by:

$$y_g \leftarrow y_g - l_r \frac{\partial \mathcal{L}_E}{\partial y_g},$$ \hspace{1cm} \text{(7)}

where $l_r$ represents the learning rate. The range of $y_g$ is clipped to $[0, 1]$.

However, the identity parameters can describe a large-scale 3D space, in which some points don’t exist in the
Table 1. The Intra-Class Correlation (ICC) and Mean Absolute Error (MAE) on FERA2015 and DISFA. Bold numbers indicate the best performance. Underline numbers indicate the second best. (*) indicates results taken from the reference.

| Database | FERA2015 | DISFA |
|----------|----------|-------|
|          | 01 02 04 05 06 09 12 15 17 20 25 26 Avg | 01 02 04 05 06 09 12 15 17 20 25 26 Avg |

| ICC       | KBSS [47] | KJRE [49] | LBA [15] | 2DC [28] | CFLF [48] | GENet-O | GE-Net |
|----------|-----------|-----------|----------|----------|-----------|---------|--------|
|          | 0.76 0.73 0.84 0.45 0.45 0.65 | 0.71 0.61 0.87 0.39 0.42 0.60 | 0.71 0.64 0.81 0.23 0.50 0.58 | 0.76 0.71 0.85 0.45 0.53 0.66 | 0.70 0.55 0.69 0.05 0.59 0.57 | 0.77 0.70 0.83 0.41 0.60 0.66 | 0.67 0.80 0.71 0.61 0.50 0.66 | 0.75 0.85 0.73 0.63 0.53 0.68 |
| ICC       | KBSS [47] | KJRE [49] | LBA [15] | 2DC [28] | CFLF [48] | GENet-O | GE-Net |
|          | 0.74 0.77 0.69 0.99 0.90 0.82 | 0.82 0.95 0.64 1.08 0.85 0.87 | 0.64 0.80 0.56 1.10 0.62 0.74 | 0.87 0.84 0.92 0.67 0.73 0.81 | 0.72 0.84 0.92 0.67 0.73 0.81 | 0.62 0.83 0.62 1.00 0.63 0.74 | 0.49 0.72 0.45 0.43 0.34 0.49 | 0.45 0.60 0.37 0.40 0.27 0.42 |

The formula is defined as:

\[ U = SVD((\alpha_{id} - m)^T(\alpha_{id} - m)), \] (8)

\[ \hat{\alpha}_{id} = (\alpha_{id} - m)U. \] (9)

\[ \hat{\alpha}_{id} \] means the identity parameters in the low-dimensional subspace. Each of the parameters \( \hat{\alpha}_{id} \) can be normalized to [0,1] according to the 20K face images in the subspace. The \( \hat{\alpha}_{id} \) is back-projected to the high-dimensional space by multiplying the transpose of the projection matrix. The final predicted identity parameters can be reconstructed as:

\[ \alpha_{id} = \hat{\alpha}_{id}U^T + m. \] (10)

Fig. 4 shows some randomly generated faces without and with the facial constraint. Before the input image is fed into the feature extractor, it is aligned to the “base face” via five facial landmarks. The “base face” is created by rendering a frontal emotionless face image, with the identity parameters set to 0.5.

In the fitting process, we set different learning rates for \((h, \alpha_{exp}, \alpha_{id})\). For the identity parameters, the learning rate is 8.0. For the head pose and AU parameters, we find each parameter has a different impact on the generated facial expression, so we set different learning rates. Specifically, starting from the “base face”, to compute the learning rate for one parameter in \((h, \alpha_{exp})\), we set its value to the maximum value while keeping other parameters zero, and obtain a corresponding segmentation output. We count the different pixels compared with the segmentation output of “base face”, and set the inverse of the ratio of different pixels in the whole image as the learning rate. The optimization can converge well after 30 iterations in the experiment.

4. Experiments

4.1. Dataset

**Public Data.** We adopt the Extended Cohn-Kanade database (CK+) [29], the MMI database [34], FERA2015 [42], the DISFA [30] and the FaceWarehouse [6] to evaluate our method. The CK+ database contains 593 sequences from 123 subjects. Approximately 15% of the sequences are coded by a certified FACS coder, which coincides with the peak frames. The MMI database contains more than 2800 samples, which contain static images and image sequences in frontal and in profile view. The FERA 2015 contains about 140,000 images from 41 subjects. Intensities are annotated for 5 AUs (6, 10, 12, 14, 17). The DISFA database consists of 27 sequences from 27 subjects. Around 130,000 frames are annotated with AU intensity for 12 AUs (1, 2, 4, 5, 6, 9, 12, 15, 17, 20, 25, 26).

**Rendered Data.** We render 80,000 face images using Justice Face with random facial parameters. The facial parameters is a 269-dimensional vector with head pose \(h \in \mathbb{R}^{1 \times 23}\) and identity parameters \(\alpha_{id} \in \mathbb{R}^{1 \times 244}\). Specifically, the range of \(h\) is limited to ensure a proper facial expression.
Figure 5. Qualitative AU intensity estimation results (a) and the transferred results (b). The results of DCLM [44], 3DDFA [56] and FEN [7] are from original paper. The results of AU parameters transferred to another 3D model are shown in (b).

is from $-28.6^\circ$ to $28.6^\circ$ in pitch and yaw, and we convert the angles to be within $[0, 1]$. We only consider the head pose in pitch and yaw because the image is aligned in the roll, which means simple 2D in-plane rotation. The initial values of the head pose should represent a frontal view, so the values are set to 0.5. The identity parameters consist of 208-dimensional continuous parameters and 36-dimensional discrete parameters. These discrete parameters are encoded as a one-hot vector and are concatenated with the continuous ones. The range of the facial parameters is from 0 to 1. The face samples are generated randomly from a uniform distribution of all parameters. 90% face samples are used for training the generator $M$, and the others are used for validating the generator.

### 4.2. Comparison with the state-of-the-art

#### Comparison with weakly and semi-supervised learning methods

We compare the proposed method with several state-of-the-art weakly-supervised learning methods (KBSS [47], KJRE [49] and CFLF [48]) and semi-supervised learning methods (LBA [15] and 2DC [28]). LBA [15] encourages cycle-consistent association chains from the embedding of labeled samples to unlabeled samples. The back is based on the assumption that samples with similar labels have similar latent features. 2DC [28] combines variational auto-encoder and Gaussian Process for AU intensity estimation by joint learning of latent representations and classification of multiple ordinal outputs. KBSS [47] exploits four types of domain knowledge on AUs to train a deep model. KJRE [49] jointly learns the representation and estimator with limited annotations by incorporating various types of human knowledge. CFLF [48] uses a learnable task-related context and two types of attention mechanisms to estimate the AU intensity. Note that KBSS, CFLF and 2DC need labeled data to train their models. Unlike them, we propose a framework for facial AU intensity estimation without annotated AU data.

Intra-Class Correlation (ICC(3,1) [38]) and Mean Absolute Error (MAE) are adopted as the metrics for evaluation. The quantitative results are listed in Table 1, and more quantitative results are listed in the supplementary material. We can see that on the FERA2015 dataset, our method achieves superior average performance over other methods on both metrics. On DISFA, our method achieves the best average performance on ICC and the second-best on MAE. Note that ICC and MAE should be jointly considered to evaluate one method. LBA tends to predict the intensity to 0, which is the majority of AU intensity. It can achieve good MAE performance, but its ICC is much worse than ours. Our

| Method     | ICC  | MAE  |
|------------|------|------|
| OpenFace 11| .392 | .421 |
| CNN [13]  | .328 | .423 |
| ResNet18 [16] | .270 | .483 |
| CNN-IT [43] | .377 | .663 |
| GE-Net    | .641 | .329 |
method outperforms KBSS and CFLF on DISFA and FERA 2015. During the training phase, these two approaches need multiple frames as input, hence it requires more than one image per face, which is not always available. Besides, the proposed GE-Net utilizes the attention mechanism for the feature extractor, which achieves better results than GENet-O, which does not use the attention mechanism. It demonstrates the effectiveness of the attention mechanism on the feature extractor. We also noticed that our method has bad performance on AU 12 in DISFA. It is because Justice Face restricts the expression of lip corner puller, whose movement is smaller than the real face so that the model can express elegant expression. These results show the superior performance of the proposed method over existing weakly and semi-supervised learning methods.

Comparison with supervised learning methods. We compare our method with several supervised learning methods of AU intensity estimation, including OpenFace [1], ResNet18 [16], CCNN-IT [43] and CNN [13]. CNN [13] is composed of three convolutional layers and a fully connected layer. CCNN-IT [43] takes advantage of CNNs and data augmentation. ResNet18 is introduced in [16]. OpenFace [1] is based on histograms of oriented gradients and landmark locations. These supervised methods require annotating AU intensity of each frame in sequences while ours does not need the AU annotations. The results are shown in Table 2. Our method achieves better performance on both metrics.

Qualitative AU intensity estimation results. We also show the qualitative results which are rendered using Justice Face with the estimated facial parameters. Fig. 5(a) provides several samples of the rendered images of different methods. Our method can achieve more accurate estimation than the state-of-the-art methods based on 3DMM. As shown in Fig. 5(b), the AU parameters of the inputs are transferred to another 3D face model. Fig. 8 shows our additional qualitative results on the mentioned datasets and internet images. (More results are given in the supplementary material.)

4.3. Ablation study

We perform an ablation study to verify the effectiveness of the feature extractor. A baseline using the loss in the raw pixel space is explored. In addition, instead of using the facial segmentation network, we use the pre-trained ResNet or VGG to extract the features of an image. Fig. 6 shows the comparison results. We can see that the results using the raw pixel loss or the pre-trained VGG16 are rather inferior, with a lot of artifacts. The pre-trained ResNet50 can not distinguish the input images with different expressions. In contrast, our facial segmentation network can be aware of the facial shape changes and produce similar expressions as the input images.

We further validate the effectiveness of the facial segmentation network by comparing it with the facial expression recognition model “ResNet” and the face recognition model “EvoLVe” [51]. A natural idea would be to use a facial expression recognition model to measure the expression similarity between the input photo and the neurally rendered one. However, due to the huge domain gap, existing pre-trained models fail to measure such similarity. The face recognition model also fails to distinguish the different expressions, which is illustrated in Fig. 6. As shown in Fig. 7 ten different expressions are selected to compute the losses on the FaceWarehouse dataset. We compare the

\footnotesize\url{https://github.com/WuJie1010/Facial-Expression-Recognition.Pytorch} \url{https://github.com/ZhaoJ9014/face.evoLVe.PyTorch}
performances of the expression recognition loss, face recognition loss and face segmentation loss using the expression similarity matrix. The expression similarity matrix is normalized and shown in the JET style. Obviously, a useful metric for expression similarity measurement should have low values on the diagonal of the matrix, while only the segmentation loss meets the requirement well.

5. Conclusion

In this paper, we propose an unsupervised framework for estimating the head pose, the AU parameters, and the identity parameters from a real facial image without annotated AU data. We formulate the method as a differentiable optimization problem by minimizing the differences of the extracted features between the real facial image and the generated one. The loss can be back-propagated to the facial parameters because the generator is differentiable. Experimental results demonstrate that the proposed method can fit accurate AU parameters and achieve state-of-the-art performance compared with existing methods. It can also get accurate AU estimation on internet photos. In the future, we plan to collect a large number of face videos with the various expression of different views. The database can be labeled using our GE-Net so that we can exploit faster methods for AU intensity estimation. Meanwhile, a more robust method to handle the large change of the head pose is another direction.
emotion specified expression. In *Computer Vision and Pattern Recognition-Workshops*, pages 94–101. IEEE, 2010.

[30] S Mohammad Mavadati, Mohammad H Mahoor, Kevin Bartlett, Philip Trinh, and Jeffrey F Cohn. Disfa: A spontaneous facial action intensity database. *Transactions on Affective Computing*, 4(2):151–160, 2013.

[31] Daniel McDuff, Rana Kalioyib, Thibaud Senechal, May Amr, Jeffrey Cohn, and Rosalind Picard. Afflectiva-mit facial expression dataset (am-fed): Naturalistic and spontaneous facial expressions collected. In *Computer Vision and Pattern Recognition Workshops*, pages 881–888. IEEE, 2013.

[32] Thu H Nguyen-Phuoc, Chuan Li, Stephen Balaban, and Yongliang Yang. Rendernet: A deep convolutional network for differentiable rendering from 3d shapes. In *Neural Information Processing Systems*, pages 7891–7901, 2018.

[33] Xuesong Niu, Hu Han, Songfan Yang, Yan Huang, and Shiguang Shan. Local relationship learning with person-specific shape regularization for facial action unit detection. In *Computer Vision and Pattern Recognition*, pages 11917–11926, 2019.

[34] Maja Pantic, Michel Valstar, Ron Rademaker, and Ludo Maat. Web-based database for facial expression analysis. In *International Conference on Multimedia and Expo*, pages 5–pp. IEEE, 2005.

[35] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[36] Ognjen Rudovic, Vladimir Pavlovic, and Maja Pantic. Context-sensitive dynamic ordinal regression for intensity estimation of facial action units. *Pattern Analysis and Machine Intelligence*, 37(5):944–958, 2014.

[37] Tianyang Shi, Yi Yuan, Changjie Fan, Zhengxia Zou, Zhenwei Shi, and Yong Liu. Face-to-parameter translation for game character auto-creation. In *International Conference on Computer Vision*, pages 161–170, 2019.

[38] Patrick E Shrout and Joseph L Fleiss. Intraclass correlations: Uses in assessing rater reliability. *Psychological bulletin*, 86(2):420, 1979.

[39] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[40] Ayush Tewari, Michael Zollhofer, Hyeongwoo Kim, Pablo Garrido, Florian Bernard, Patrick Perez, and Christian Theobalt. Mofa: Model-based deep convolutional face autoencoder for unsupervised monocular reconstruction. In *International Conference on Computer Vision*, pages 1274–1283, 2017.

[41] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016.

[42] Michel F Valstar, Timur Almaev, Jeffrey M Girard, Gary McKeown, Marc Meh, Lijun Yin, Maja Pantic, and Jeffrey F Cohn. Fera 2015-second facial expression recognition and analysis challenge. In *International Conference and Workshops on Automatic Face and Gesture Recognition*, volume 6, pages 1–8. IEEE, 2015.

[43] Robert Valecki, Vladimir Pavlovic, Björn Schuller, Maja Pantic, et al. Deep structured learning for facial action unit intensity estimation. In *Computer Vision and Pattern Recognition*, pages 3405–3414, 2017.

[44] Amir Zadeh, Tadas Baltrušaitis, and Louis-Philippe Morency. Deep constrained local models for facial landmark detection. *arXiv preprint arXiv:1611.08657*, 3(5):6, 2016.

[45] Amir Zadeh, Yao Chong Lim, Tadas Baltrušaitis, and Louis-Philippe Morency. Convolutional experts constrained local model for 3d facial landmark detection. In *International Conference on Computer Vision*, pages 2519–2528, 2017.

[46] Xing Zhang, Lijun Yin, Jeffrey F Cohn, Shaun Canavan, Michael Reale, Andy Horowitz, and Peng Liu. A high-resolution spontaneous 3d dynamic facial expression database. In *Automatic Face and Gesture Recognition*, pages 1–6. IEEE, 2013.

[47] Yong Zhang, Weiming Dong, Bao-Gang Hu, and Qiang Ji. Weakly-supervised deep convolutional neural network learning for facial action unit intensity estimation. In *Computer Vision and Pattern Recognition*, pages 2314–2323, 2018.

[48] Yong Zhang, Haiyong Jiang, Baoyuan Wu, Yanbo Fan, and Qiang Ji. Context-aware feature and label fusion for facial action unit intensity estimation with partially labeled data. In *International Conference on Computer Vision*, pages 733–742, 2019.

[49] Yong Zhang, Baoyuan Wu, Weiming Dong, Zhifeng Li, Wei Liu, Bao-Gang Hu, and Qiang Ji. Joint representation and estimator learning for facial action unit intensity estimation. In *Computer Vision and Pattern Recognition*, pages 3457–3466, 2019.

[50] Yong Zhang, Rui Zhao, Weiming Dong, Bao-Gang Hu, and Qiang Ji. Bilateral ordinal relevance multi-instance regression for facial action unit intensity estimation. In *Computer Vision and Pattern Recognition*, pages 7034–7043. IEEE, 2018.

[51] Jian Zhao, Yu Cheng, Yan Xu, Lin Xiong, Jianshu Li, Fang Zhao, Karlekar Jayashree, Sugiri Pranata, Shengmei Shen, Junliang Xing, et al. Towards pose invariant face recognition in the wild. In *Computer Vision and Pattern Recognition*, pages 2207–2216, 2018.

[52] Kaili Zhao, Wen-Sheng Chu, and Aleix M Martinez. Learning facial action units from web images with scalable weakly supervised clustering. In *Computer Vision and Pattern Recognition*, pages 2090–2099. IEEE, 2018.

[53] Rui Zhao, Quan Gan, Shangfei Wang, and Qiang Ji. Facial expression intensity estimation using ordinal information. In *Computer Vision and Pattern Recognition*, pages 3466–3474, 2016.

[54] Yuxiang Zhou, Jiankang Deng, Irene Kotsia, and Stefanos Zafeiriou. Dense 3d face decoding over 2500fps: Joint texture & shape convolutional mesh decoders. In *Computer Vision and Pattern Recognition*, pages 1097–1106, 2019.

[55] Yujian Zhou, Jimin Pi, and Bertram E Shi. Pose-independent facial action unit intensity regression based on multi-task deep transfer learning. In *Automatic Face & Gesture Recognition*, pages 872–877. IEEE, 2017.
[56] Xiangyu Zhu, Zhen Lei, Xiaoming Liu, Hailin Shi, and Stan Z Li. Face alignment across large poses: A 3d solution. In Computer Vision and Pattern Recognition, pages 146–155, 2016.

[57] Xiangyu Zhu, Zhen Lei, Junjie Yan, Dong Yi, and Stan Z Li. High-fidelity pose and expression normalization for face recognition in the wild. In Computer Vision and Pattern Recognition, pages 787–796, 2015.

[58] Xiangyu Zhu, Zhen Lei, Junjie Yan, Dong Yi, and Stan Z Li. High-fidelity pose and expression normalization for face recognition in the wild. In Computer Vision and Pattern Recognition, pages 787–796. IEEE, 2015.

[59] Xiangyu Zhu, Xiaoming Liu, Zhen Lei, and Stan Z Li. Face alignment in full pose range: A 3d total solution. Pattern Analysis and Machine Intelligence, pages 78–92, 2017.
A. Appendix

A.1. More Experiments

**Stability.** We hope that the predicted AU parameters are stable in the continuous frames of one video. The stability is crucial for numerous applications such as expression transfer, facial expression analysis, and human-robot interaction. To evaluate the stability, we suppose that the continuous two frames are almost the same in one video. So, the stability is evaluated by the differences between the continuous frames, which is calculated by the standard deviation of each AU in the fixed-size sliding windows of continuous frames. We compute the standard deviation of ten video sequences from 2 to 10 frames. The ten videos are download from the internet. The results of our method are better than OpenFace on different sliding windows. The result of 5 frames is shown in Table 3. We also show the results of a video in different frames in Fig. 9.

| AU  | 01  | 02  | 03  | 04  | 05  | 06  | 07  | 09  | 10  | 12  | 14  | 15  | 15-2 |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| OpenFace | 0.247 | 0.261 | –   | 0.114 | 0.267 | 0.139 | 0.248 | **0.027** | 0.178 | 0.144 | 0.218 | –   | 0.055 |
| Ours  | **0.032** | **0.013** | 0.034 | **0.045** | **0.096** | **0.059** | **0.055** | 0.034 | **0.038** | **0.047** | **0.044** | 0.064 | **0.034** |

**Robustness to Varying Conditions.** Our network uses facial segmentation features as the input of the loss, yielding results robust to changes in lighting, resolution, and style. To verify it, we add robustness experiments in FaceWarehouse dataset [6]. Different lights, gaussian blur, and style transfer [19] are used to the dataset. Some results are shown in Fig. 10. We qualitatively demonstrate this robustness by varying conditions for a single subject, and the results show consistent output. The first row is the input image. The rendered images and the corresponding histograms of the facial AU parameters are shown in the second and the third row. We also noticed that our method has bad performance on the upper lid raiser (column 4 in the histogram). It is because Justice Face restricts the expression of the upper lid raiser, whose movement is smaller. (please refer to Fig. 11.)

In our paper, the useful action unit and the corresponding name are listed in Table 4 based by row and then column. Each of AU parameters is set to the max value, and the corresponding rendered image to base face is listed in Fig. 11.

**Compared with OpenFace.** We provide a comparative experiment to investigate the effectiveness of our GE-Net. The results are shown in Table 5. The GE-Net achieves better performance in MAE and ICC on average and many AUs, which demonstrates the effectiveness of the attention mechanism.

| AU  | 01  | 02  | 03  | 04  | 05  | 06  | 07  | 09  | 10  | 12  | 14  | 15  | 15-2 |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| OpenFace | 0.017 | –   | 0.162 | 0.077 | 0.226 | –   | –   | –   | 0.221 | –   | –   | –   | –   |
| Ours  | 0.042 | 0.054 | **0.091** | –   | **0.073** | 0.067 | 0.071 | 0.053 | 0.050 | **0.102** | 0.124 | 0.024 | 0.060 |

Figure 9. A video expression transfer example. The upper row shows the several frames of one video, and the bottom row shows the generated images.
Figure 10. Robustness test. The network is robust to change in lighting, resolution, and style. From top to bottom, we show the five input images, the rendered images, and the corresponding AU parameter histograms.

| Name                  | AU   | Name                  | AU   | Name                  | AU |
|-----------------------|------|-----------------------|------|-----------------------|----|
| Left Eye Blink        | 45   | Right Eye Blink       | 45-2 | Upper Lid Raiser      | 5  |
| Cheek Raiser          | 6    | Inner Brow Raiser     | 1    | Left Outer Brow Raiser| 2  |
| Right Outer Brow Raiser| 3    | Brow Lowerer          | 4    | Mouth Stretch         | 27 |
| Nose Wrinkler         | 9    | Upper Lip Raiser      | 10   | Down Lip Down         | 25 |
| Lip Corner Puller     | 12   | left mouth press      | 14   | right mouth press     | 14-2|
| Lip Pucker            | 18   | Lip Stretcher         | 20   | Lip Upper Close       | 24 |
| Lip Lower Close        | 17   | Cheek Puff            | 34   | Lip Corner Depressor  | 15 |
| Jaw Left              | 47   | Jaw Right             | 48   |                       |    |

Table 5. The MAE on CK+ and MMI. Bold numbers indicate the best performance. GENet-O means our method without attention.

| Database | AU | 01  | 02  | 04  | 05  | 06  | 07  | 09  | 10  | 12  | 17  | 23  | 24  | 25  | Avg |
|----------|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| CK+      | OPENFACE | .45 | .41 | .74 | .33 | .65 | 1.34 | .58 | .43 | .46 | .94 | .56 | .60 | .63 | .624|
|          | GE-NET   | .40 | .27 | .66 | .27 | .30 | .30  | .51 | .14 | .68 | .77 | .48 |    |    | .39 | .443|
| MMI      | OPENFACE | 1.09 | .86 | .23 | .68 | .12 | .21 | .16 | .27 | .20 | .60 | .27 | .46 | .32 | .421|
|          | GE-Net   | .12 | .18 | .23 | .09 | .11 | .11  | .33 | .56 | .24 | .324| .32 |    |    | .10 | .231|
| CK+      | OPENFACE | .36 | .48 | .24 | .51 | .29 | .35  | .51 | .34 | .58 | .53 | .48 | .31 | .61 | .430|
|          | GE-NET   | .58 | .53 | .46 | .78 | .70 | .71  | .80 | .36 | .75 | .61 |    |    |    | .63 | .611|
| ICC      | OPENFACE | .50 | .31 | .24 | .44 | .43 | .47  | .31 | .45 | .38 | .27 | .48 | .46 | .32 | .389|
|          | GE-Net   | .53 | .48 | .38 | .63 | .47 | .53  | .57 | .49 | .48 | .41 | .60 | .51 | .495|     |
B. AU parameters expression on the 3d facial model

There are a little bit differences between the AU parameters from the 3d facial model and the AUs from the FACS system (e.g. we separate the dimple to left dimple and right dimple), as shown in Fig. [11].

Figure 11. The generated images with the corresponding AU parameters.
C. Training samples of our generator

In the training process, we train our generator with randomly generated game faces, as shown in Fig. 12, which are decided by the head pose, AU parameters and identity parameters.

D. More examples of our method

More examples of our method are shown in Fig. 13 and 14. Due to richer expressions on FaceWarehouse dataset, we show more examples.
Figure 13. More results of our method on MMI.
Figure 14. More results of our method on FaceWarehouse.