Facial Expression Recognition With Two-Branch Disentangled Generative Adversarial Network

Siyue Xie, Haifeng Hu, *Member, IEEE*, and Yizhen Chen

***Abstract*— Facial Expression Recognition (FER) is a challenging task in computer vision as features extracted from expressional images are usually entangled with other facial attributes, e.g., poses or appearance variations, which are adverse to FER. To achieve a better FER performance, we propose a model named Two-branch Disentangled Generative Adversarial Network (TDGAN) for discriminative expression representation learning. Different from previous methods, TDGAN learns to disentangle expressional information from other unrelated facial attributes. To this end, we build the framework with two independent branches, which are specific for facial and expressional information processing respectively. Correspondingly, two discriminators are introduced to conduct identity and expression classification. By adversarial learning, TDGAN is able to transfer an expression to a given face. It simultaneously learns a discriminative representation that is disentangled from other facial attributes for each expression image, which is more effective for FER task. In addition, a self-supervised mechanism is proposed to improve representation learning, which enhances the power of disentangling. Quantitative and qualitative results in both in-the-lab and in-the-wild datasets demonstrate that TDGAN is competitive to the state-of-the-art methods.**

***Index Terms*— Adversarial learning, discriminative expression representation learning, expression transferring, Facial Expression Recognition, Two-branch Disentangled Generative Adversarial Network.**

I. Introduction

FACIAL Expression Recognition (FER) is one of the most popular research topics in the field of computer vision, which aims to recognize six basic facial expressions (i.e., angry, disgust, fear, happy, sad and surprise) from images or videos. Previous works on FER usually go by two steps: researchers first extract expressional features to represent the given image/video and then train/use a classifier to recognize different expressions based on the extracted features.

Manuscript received June 29, 2020; accepted September 10, 2020. Date of publication September 15, 2020; date of current version June 4, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 61673402, Grant 61273270, and Grant 60802069; in part by the Natural Science Foundation of Guangdong Province under Grant 2017A030311029; in part by the Science and Technology Program of Guangzhou of China under Grant 201704020180; and in part by the Fundamental Research Funds for the Central Universities of China. This article was recommended by Associate Editor G. Hua. *(Corresponding author: Haifeng Hu.)*

Siyue Xie is with the Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong.

Haifeng Hu and Yizhen Chen are with the School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou 510275, China (e-mail: huhaif@mail.sysu.edu.cn).

Color versions of one or more of the figures in this article are available online at https://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2020.3024201

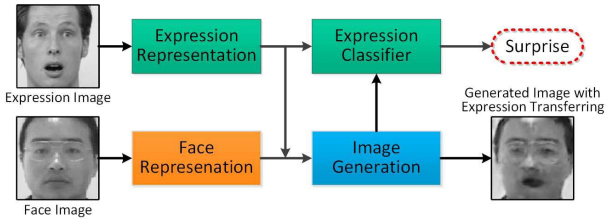


Fig. 1. An overview of the proposed TDGAN model.

Conventional methods utilize some hand-crafted features to represent an expressional images [1], [2]. In recent years, deep-learning algorithms are also introduced to solve the FER task [3], [4]. Since deep-learning models learn features automatically and always be trained in an end-to-end manner, the performance of them is usually better and therefore be regarded as a promising solution. However, FER is still far from being solved. On one hand, expressional actions usually occur in specific local areas (e.g., the neighbourhood of eyes or mouth), while features extracted from expression-unrelated regions can be redundant to FER and may deteriorate the overall performance. On the other hand, some variations, such as poses or subjects' appearance differences, can impair the quality of the extracted features, which make it less discriminative when used to classify different expressions.

Having considered the aforementioned obstacles, in this paper, we propose a novel model, named Two-branch Disentangled Generative Adversarial Network (TDGAN) to learn discriminative expression representations for FER. Instead of suppressing variations information when extracting expression features, which is common to some previous FER methods [4], [5], TDGAN learns to disentangle expression from other unrelated facial attributes through expression transferring. Different from other FER methods, the model is based on the framework of Generative Adversarial Network (GAN) and built with two independent branches: the expression branch is specific for expressional information processing while the face branch is responsible for the encoding of information of other facial attributes. An overview of TDGAN is presented in Fig.1. Concretely, the model takes an image pair as input, which includes a face image and an expression image. In the generator of TDGAN, representations of the two inputs are yielded by two independent encoders respectively and then fused by a decoder, which synthesizes an image by transferring the input expression into the face image. In order to evaluate

the generated image, we configure two discriminators for TDGAN. One is used for face recognition and the other for expression recognition. Following the principle of adversarial learning, the generator is encouraged to learn a disentangled expressional representation for the input expression image, which is discriminative for FER task. Visualization results also demonstrate the effectiveness of the model (see Fig.7 in Section IV-C).

The contributions of our paper can be summarized as follows:

- We propose a model named TDGAN that is able to learn disentangled and discriminative expressional representation for FER task. By adopting adversarial learning, TDGAN is encouraged to disentangle expressional information from other redundant facial attributes or variations, which results in a better performance on FER task.
- The proposed TDGAN is able to conduct expression transferring. Since representations of the input image pair can be effectively fused, TDGAN is able to synthesize an image with expected expression by modifying some specific regions of the input face image.
- TDGAN can be trained in some small expression datasets. By jointly training with an auxiliary face dataset, TDGAN is encouraged to learn the data distribution of both expression and face datasets. This makes it possible to work well even though there are only limited training samples of labelled expression images.
- TDGAN outperforms many existing FER methods on both in-the-lab and in-the-wild datasets. Visualization and recognition results demonstrate the effectiveness of our model.[1]

II. Related Works

In this section, we briefly review two main topics that are relevant to our work: Facial Expression Recognition (FER) and Generative Adversarial Network (GAN).

A. Facial Expression Recognition

Methods on FER can generally be categorized into two groups according to the type of the extracted features, i.e., hand-crafted features and deep-learning-based features. Some hand-crafted features, e.g., geometrical features [1], are commonly used in traditional FER methods. Since these patterns are mostly defined based on prior knowledge, methods based on hand-crafted features usually perform well in some in-the-lab datasets that without many variations [6]. In recent years, deep-learning-based approaches develop rapidly and show great capacity to process different expressions. Jung *et al.* [3] utilize two deep networks to process different patterns and improve the overall performance by using a joint fine-tune framework. Xie and Hu [7] extract holistic and local features for FER through multiple independent CNN networks. Yang *et al.* [4] propose a De-expression Residue Learning (DRL) method, which recognizes facial expression by learning the residual expressive component. Some related works resort to disentangling representation learning for facial or expressional information processing. Jiang *et al.* [8] propose a 3D face shaping model with two branches to encode expression and identity information respectively. Zhang *et al.* [9] propose a supervised model for face synthesis based on Siamese Network, which disentangles identity from variations of poses and illuminations. Hinz and Wermter [10] use an encoder to generate image representations, which is divided into a controllable part and redundant part. Different from those aforementioned approaches, our model learns a disentangled expressional representation by transferring an expression to another face, which effectively improves the classification performance on different expressions.

[1]Code will soon be released in https://github.com/XsLangley/TDGAN

B. Generative Adversarial Network

Generative Adversarial Network (GAN) was first proposed by Goodfellow *et al.* [11], which consists of a generator G and a discriminator D. The generator is expected to synthesize fake/negative samples using a random noise z while the discriminator is responsible for distinguishing real/positive samples from the fakes. By playing a minmax two-player game, the model can be trained in a competing manner as follows:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{data}}[\log(D(x))] + \mathbb{E}_{z\sim p_z}[\log(1-D(G(z)))], \quad (1)$$

where p_{data} is the data distribution of real samples and the noise z follows the distribution of p_z.

GAN has already been widely applied in many task of computer vision [12], [13]. Choi *et al.* [14] propose the StarGAN to perform image-to-image translations, which is able to render a specific expression given a face image. In [15], Pumarola *et al.* introduce a novel GAN conditioning scheme based on Action Units (AU) annotations. This approach synthesize facial animation reflecting different expressions. To solve the task of pose-invariant face recognition, Tran *et al.* [16] propose the DR-GAN, which learns a discriminative face representation that can be used for frontal face generation. Ding *et al.* [17] propose ExprGAN to control the expression intensity in a fully-supervised manner. Different from the aforementioned methods, we adopt a dual-path structure in the generator. Following the concept of adversarial learning, our model is encouraged to disentangle expressional information from other facial attributes, which yields a more discriminative expression representation for FER task.

III. Proposed Method

This section describes the details of our novel end-to-end approach, which generates disentangled expression representations through expression transferring. Compared with existing methods, the synthesized images of TDGAN directly reflect the effect of disentangling. This provides a more intuitive way to evaluate the model. In the following, we first illustrate the architecture of TDGAN and then strategies for model

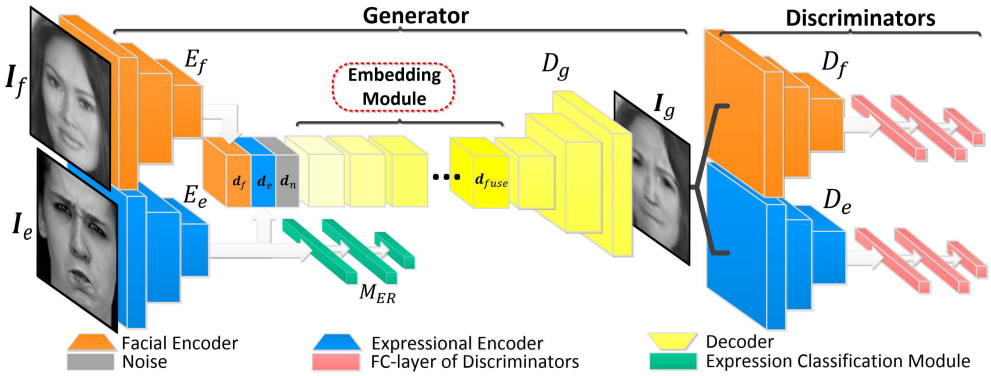


Fig. 2. The framework of TDGAN. Expression and face representations are generated through the two encoders and then used to generate an image, while the two discriminators help evaluating the synthesized image. By adversarial training, the learned expression representation (in generator) can be disentangled from other facial attributes and therefore used for FER task.

improvement. We finally specify how to recognize different expressions using the learned model.

A. Network Architecture of TDGAN

The framework of TDGAN is shown in Fig.2. Following the basic principle of GAN, the proposed model can be divided into two parts: the generator and discriminators.

1) Generator: The goal of generator is to extract specific image features and transfer a specific expression to another face, as shown in Fig.2. The network structure of the generator is based on the encoder-decoder architecture, where detailed configuration can be found in the Section IV-A2. To better control the process of expression transferring, TDGAN is expected to disentangle expressional information from other facial attributes. To this end, we configure the generator with two independent encoder branches, one specific for expressional features extraction and the other for facial features extraction.

The input of the model is an image pair, which includes a face image $\boldsymbol{I}_f$ (only sampled from a face dataset) and an expression image $\boldsymbol{I}_e$ (only sampled from an expression dataset). We can obtain face and expression representations can be obtained by feeding the inputs into the corresponding encoders. To embed the input expression into the face image, we fuse the face representation with the expression representations before up-sampling. Concretely, we introduce an embedding module into our model, which is made up of several residual blocks [18]. The introduction of the embedding module gives the generator more capacities to process data from different representation space. This enables the model to modify the expression but keep other facial attributes, e.g. hairs, glasses or poses, nearly untouched. The output of the embedding module encodes the high-level semantics of the two input images and then be fed into the decoder for image synthesis. To recognize different expressions, we configure a classification module (consists of a three-layer fully-connected network and a softmax classifier) after the expressional encoder, which takes the generated expression representation as input for the final FER task.

2) Discriminators: Different from most GAN-based methods, the proposed TDGAN contains two discriminators: an expression discriminator D_e and a face discriminator D_f. These two discriminators are trained to evaluate whether the generated image contains the content we expect. Specifically, the expression discriminator is trained to conduct expression classification, while the face discriminator is expected to classify different identities. These two discriminators have similar network structure, which includes an encoding module (the network configuration is the same as the two encoders in the generator but with independent parameters) for feature extraction and a three-layer fully-connected network followed by a softmax classifier for classification task. One main difference between D_e and D_f is that we add an additional class to the target of face discriminator, which is used to indicate whether the input image is from real samples or from generated images. The role of the additional class can be two-fold. On one hand, such setting follows the basic concept of GAN framework, which drives the model to distinguish fake/negative samples from real/positive samples. On the other hand, the introduction of the additional class keeps the model in a correct track during training. Detailed explanations will be specified in the following Section III-B1.

B. Model Learning

As a GAN-based model, TDGAN is trained following the principle of adversarial learning. Furthermore, we propose a dual image consistency framework and introduce the perceptual constraint into our model for performance improvement. The former introduces a self-supervision mechanism into the model to improve the model learning, while the latter induces the generated image to follow the similar high-level semantics as the input face image. The whole learning strategy encourages the model to yield a disentangled expressional representation, which can be more effective in the FER task.

1) Expression Transferring With Adversarial Learning: Given an input image pair $\{(\boldsymbol{I}_f, \boldsymbol{y}_f), (\boldsymbol{I}_e, \boldsymbol{y}_e)\}$, where $\boldsymbol{y}_f$ and $\boldsymbol{y}_e$ are one-hot labels of the identity and expression respectively, we feed them into the two encoders to obtain the corresponding image representations. Concretely, we denote E_f as the face encoder and E_e as the expression encoder. Thus, the encoded representations of the two inputs can be

formulated as:

$$\boldsymbol{d}_f = E_f(\boldsymbol{I}_f), \boldsymbol{d}_e = E_e(\boldsymbol{I}_e), \tag{2}$$

where $\boldsymbol{d}_f$ is the face representation and $\boldsymbol{d}_e$ is the expression representation. We then fuse these two representations through the embedding module, which can be described as:

$$\boldsymbol{d}_{fuse} = Emb(con(\boldsymbol{d}_f, \boldsymbol{d}_e, \boldsymbol{d}_n)), \tag{3}$$

where $\boldsymbol{d}_{fuse}$ denotes the fused representation, $Emb(x)$ is the embedding module and $con(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ indicates the operation of channel-wise concatenation. We additionally introduce a noise vector $\boldsymbol{d}_n$ into the fusing process, which results in some minor variations, e.g., a different contrast ratio, in generated images and makes the model more robust during training. In our expectation, $\boldsymbol{d}_{fuse}$ lies in a hidden space that encodes both the high-level semantics of the input face and expression images. Therefore, we generate the image through the decoder based on $\boldsymbol{d}_{fuse}$, which can be expressed as follows:

$$\boldsymbol{I}_g = D_g(\boldsymbol{d}_{fuse}) = G(\boldsymbol{I}_f, \boldsymbol{I}_e, \boldsymbol{d}_n), \tag{4}$$

where D_g and $G(\boldsymbol{x})$ denote the decoder and the overall generator respectively.

Ideally, $\boldsymbol{I}_g$ should follow the same facial appearance as $\boldsymbol{I}_f$ and simultaneously with an expression consistent with $\boldsymbol{I}_e$. To evaluate whether the generated image fulfills our expectation, we design two discriminators to guide the generator. The expression discriminator $D_e(\boldsymbol{x}) \in \mathbb{R}^{K_e}$ conducts the task of expression recognition while the face discriminator, i.e., $D_f(\boldsymbol{x}) \in \mathbb{R}^{K_f+1}$, is responsible for identity classification. Here, K_e is the class number of expressions and K_f is the class number of subjects we used in training. The additional class in the face discriminator is designed to distinguish fake/negative images from real/positive samples. In other words, all generated images will be labelled with the $(K_f + 1)$-th class as ground truth when we train the face discriminator. In addition, the introduction of the additional class stabilizes the training of our model. This is because generated images can be visually meaningless in the beginning of training. If without the additional negative class, all generated images will be regarded as 'successful cases' and used to train face discriminator, which may mislead the generator to a wrong learning direction. By introducing the additional negative class, the generator gets feedback from the discriminator, which forces the generator to synthesize images that follow the data distribution of face dataset and helps rendering a consistent expression. Therefore, the objective function can be formulated as:

$$\begin{aligned} L_f &= \mathbb{E}_{(\boldsymbol{I}_f, \boldsymbol{y}_f)\sim p_f}[\log D_f(\boldsymbol{I}_f)] + \mathbb{E}_{(\boldsymbol{I}_g, \boldsymbol{y}_g)\sim p_g}[\log D_f(\boldsymbol{I}_g)] \\ L_e &= \mathbb{E}_{(\boldsymbol{I}_e, \boldsymbol{y}_e)\sim p_e}[\log D_e(\boldsymbol{I}_e)] \end{aligned} \tag{5}$$

where L_f and L_e are the corresponding loss function of the face discriminator and the expression discriminator. p_f, p_e and p_g refer to the data distribution of face images, expression images and generated images respectively. y_g is the label of generated images. Note that in the training of discriminators, generated images are only fed into the face discriminator. This is because TDGAN is only expected to modify the expression of the input face image. In other words, contents of the generated image should follow the input face image rather than the expression image. Therefore, the additional 'fake' class is only assigned to the face discriminator instead of the expression discriminator. By sending the generated image to the face discriminator and following adversarial training, the decoder in TDGAN is induced to learn the distribution of the face dataset. Furthermore, the expression discriminator is not expected to verify whether an image is real or fake as we have already set an additional (negative) class in face discriminator. Thus, the generated image will not be sent to the expression discriminator when training. Instead, the expression discriminator is expected to concentrate on learning meaningful knowledge of different expressions. It will be more effective to learn expressional knowledge from real samples than generated images. In addition, generated images can be visually meaningless or with unexpected expressions in the beginning of training. Training the expression discriminator with incorrect expressional information can 'contaminate' the learning process, which may result in the collapse of the training. By maximizing Equ.5, we can optimize these two discriminator separately.

As for the generator, its goal is to generate images that can fool the two discriminators. Thus, the classification loss of the generator can be defined as:

$$\begin{aligned} L_C = -\{&\lambda_{G_f}\mathbb{E}_{(\boldsymbol{I}_g, \boldsymbol{y}_f)\sim p_g}[\log(D_f(\boldsymbol{I}_g))] \\ &+\lambda_{G_e}\mathbb{E}_{(\boldsymbol{I}_g, \boldsymbol{y}_e)\sim p_g}[\log(D_e(\boldsymbol{I}_g))]\}, \end{aligned} \tag{6}$$

where λ_{G_f} and λ_{G_e} are two adjustable coefficients to balance the face and expression branch. Since the knowledge of the desired expression of $\boldsymbol{I}_g$ can only be learnt from $\boldsymbol{I}_e$, the expression encoder is encouraged to extract the most intrinsic features of expressions. In the meanwhile, the generated image should preserve the same identity or facial appearance as $\boldsymbol{I}_f$, which induces the face encoder to learn some core facial information. By minimizing Equ.6, the generator can effectively disentangle the expressional information from other irrelevant facial attributes. The image representation yielded by the expressional encoder can therefore be more discriminative for FER task.

2) Dual Image Consistency: With the previously defined loss, TDGAN is able to complete basic expression transferring and learn disentangled representations. However, it will be better if the model can be trained with ground-truth supervision. Considering that the generated image should preserve information of both inputs, we can make use of it to reconstruct the inputs, which indirectly introduces a self-supervision into the model. To this end, we propose a novel dual-image-consistency (DIC) constraint to improve model learning. The framework of DIC is shown in Fig.3. Concretely, we force the generator to reconstruct the input image pair by a two-stage generation. In the first stage, we use the given image pair $\{\boldsymbol{I}_f, \boldsymbol{I}_e\}$ to synthesize image $\boldsymbol{I}_g$. In the second stage, we reconstruct the two input images separately. $\boldsymbol{I}_f$ and $\boldsymbol{I}_g$ are fed into the expression and face branch respectively and generate the reconstructed face image $\hat{\boldsymbol{I}}_f = G(\boldsymbol{I}_g, \boldsymbol{I}_f)$. Analogously, $\boldsymbol{I}_e$ is fed into the face branch while $\boldsymbol{I}_g$ is fed into the expression branch to generate $\hat{\boldsymbol{I}}_e = G(\boldsymbol{I}_e, \boldsymbol{I}_g)$. In this way,

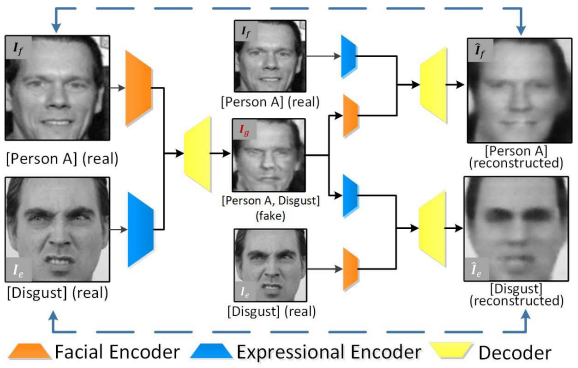


Fig. 3. Illustration of the dual-image-consistency of TDGAN.

we can build the DIC loss by penalizing the difference between the original input images ($\boldsymbol{I}_f$ and $\boldsymbol{I}_e$) and their corresponding reconstructions:

$$L_D = \lambda_D \mathbb{E}_{\boldsymbol{I}_f \sim p_f, \boldsymbol{I}_e \sim p_e}[\|\hat{\boldsymbol{I}}_f - \boldsymbol{I}_f\|_1 + \|\hat{\boldsymbol{I}}_e - \boldsymbol{I}_e\|_1], \quad (7)$$

where λ_D is an adjustable coefficient. Since the reconstruction process (the second stage) is based on the generated image $\boldsymbol{I}_g$, facial information used to generate $\hat{\boldsymbol{I}}_f$ is only extracted from $\boldsymbol{I}_g$. Expressional information for $\hat{\boldsymbol{I}}_e$ only comes from $\boldsymbol{I}_g$ as well. To better reconstruct the two input images, the two encoders are forced to extract the required information accordingly, which drives the model to disentangle the information of expression from other facial attributes.

Compared with previous works [13]–[15], the proposed DIC constraint effectively unifies the two branch and improves the learning process of our model. On one hand, in DIC, we build two separated paths for reconstruction and one of them is specific for expressions recovering. This induces the model to be more sensitive to expressions information, which explicitly enhances the ability of the model on disentangling face and expression representations. On the other hand, the generator is shared in both stages of the reconstruction process when building the DIC constraint. Such implementation encourages each encoder to learn knowledge from both face and expression datasets, which makes fully use of data from different sources. Expression representations learned by our model can therefore be adaptive to some variations that appeared in the face dataset in some extent. This results in a better performance on FER when compared with the models that are trained on only single expression dataset.

3) Semantic Content Consistency: As for expression transferring, we should keep the semantic content of the input face image unchanged but embed the given expression into the face. To this end, we train TDGAN with an additionally perceptual loss [19], which has been widely applied to measure the difference of semantic content between two images in a high level [20], [21]. In our model, the perceptual loss is defined as:

$$L_P = \lambda_{P_f} \|\boldsymbol{d}_f - \boldsymbol{d}_{(g,f)}\|_1, \quad (8)$$

where λ_{P_f} is an adjustable parameter, $\boldsymbol{d}_f$ has been defined in Equ.2 and $\boldsymbol{d}_{(g,f)} = E_f(\boldsymbol{I}_g)$.

4) Loss Function: Taking all the loss together, we have the following loss function for our generator:

$$L_G = L_C + L_D + L_P \quad (9)$$

By optimizing Equ.9 and Equ.5 alternately, we can continuously update TDGAN.

C. Expression Recognition Using Disentangled Representation

In this work, we recognize different expressions based on the disentangled expression representations learned by the expression encoder. Specifically, we simultaneously train an expression classification module M_{ER} (as shown in Fig. 2) as we optimize TDGAN. M_{ER} is trained following the cross-entropy loss, which can be formulated as:

$$L_{ER} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{K_e} y_i^j log(\widetilde{y_i}^j), \quad (10)$$

where y_i^j is the j-th value of the ground truth expression label of the i-th sample, and $\widetilde{y_i}^j$ is the j-th output value of the softmax classifier. N is the number of training samples. When training, gradients of L_{ER} are only used to optimize M_{ER}, which will not be back-propagated to the expressional encoder.

IV. Experiments

In this section, the proposed TDGAN is evaluated on both in-the-lab datasets (CK+, TFEID and RaFD) and in-the-wild datasets (BAUM-2i and RAF-DB). In addition, we visualize the qualitative results to show the performance on expression transferring.

A. Implemental Details

1) Expression Datasets:

a) CK+: The Extended Cohn-Kanade (CK+) [22] contains 593 image sequences collected from 123 subjects. Expressions in each sequence vary from neutral face to the peak expression. In this dataset, only 309 sequences are labelled with one of the six prototypical expressions (anger, disgust, fear, happiness, sadness, surprise) and thus selected out for experiments. We pick out the last three frames of each sequence to construct the training and testing sets. Additionally, the first frame of each selected sequence is collected as a neural face. Therefore, there are totally 1236 images involved in our experiments.

b) TFEID: Taiwanese Facial Expression Image Database (TFEID) [23] is captured from 40 models, each with eight facial expressions(six typical expression + neural + contempt). In our experiments, we only pick out the images that are labeled with six basic expression and the neutral. Therefore, 580 images of TFEID are involved in our experiments.

c) RaFD: The Radboud Faces Database (RaFD) [24] is a set of pictures collected from 67 model with different ethnics. Each picture of the dataset is annotated with an expression as well as the corresponding identity. To compare with other state-of-the-art methods, we only collect the images labeled with one of the seven expressions (six prototypical expressions+neutral). Therefore, there are totally 1407 images used in our experiments.

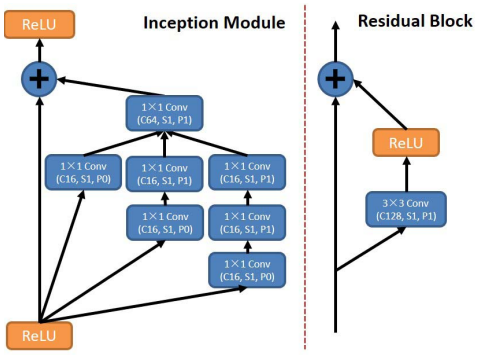


Fig. 4. Configurations of the inception module and the residual block.

TABLE I

THE ARCHITECTURE OF THE FACIAL/EXPRESSIONAL ENCODER

(C: output channels, K: kernel size, S: stride size, P: padding size, IN: Instance Normalization, *: only for face discriminators, $^\#$: only for expression discriminator and the expression classification module)

Part	Input Shape	Output Shape	Detailed Configuration
Down-Sample	$(h, w, 3)$	$(h, w, 32)$	Conv-(C32, K5, S1, P2), IN, ReLU
	$(h, w, 32)$	$(\frac{h}{2}, \frac{w}{2}, 32)$	Maxpool-(K3, S2)
	$(\frac{h}{2}, \frac{w}{2}, 32)$	$(\frac{h}{2}, \frac{w}{2}, 64)$	Conv-(C64, K5, S1, P2), IN, ReLU
	$(\frac{h}{2}, \frac{w}{2}, 64)$	$(\frac{h}{4}, \frac{w}{4}, 64)$	Maxpool-(K3, S2)
Inception × 4	$(\frac{h}{4}, \frac{w}{4}, 64)$	$(\frac{h}{4}, \frac{w}{4}, 64)$	Shown in Fig. 4
Down-Sample	$(\frac{h}{4}, \frac{w}{4}, 64)$	$(\frac{h}{4}, \frac{w}{4}, 128)$	Conv-(C128, K5, S1, P2), IN, ReLU
	$(\frac{h}{4}, \frac{w}{4}, 128)$	$(\frac{h}{8}, \frac{w}{8}, 128)$	Maxpool-(K3, S2)
	$(\frac{h}{8}, \frac{w}{8}, 128)$	$(\frac{h}{8}, \frac{w}{8}, 128)$	Conv-(C128, K5, S1, P2), IN, ReLU
	$(\frac{h}{8}, \frac{w}{8}, 128)$	$(\frac{h}{16}, \frac{w}{16}, 128)$	Maxpool-(K3, S2)
fc-layer*	$(\frac{h}{16} \times \frac{w}{16} \times 128)$	(1024)	fully-connected, ReLU
	(1024)	(1024)	fully-connected, ReLU
	(1024)	$(K^f + 1)$	fully-connected, Softmax
fc-layer$^\#$	$(\frac{h}{16} \times \frac{w}{16} \times 128)$	(1024)	fully-connected, ReLU
	(1024)	(1024)	fully-connected, ReLU
	(1024)	(K^e)	fully-connected, Softmax

TABLE II

THE ARCHITECTURE OF THE EMBEDDING MODULE AND THE DECODER

(C: output channels, K: kernel size, S: stride size, P: padding size, IN: Instance Normalization)

Module	Input Shape	Output Shape	Detailed Configuration
Embedding	$\boldsymbol{d}_e(\frac{h}{16}, \frac{w}{16}, 128)$ $\boldsymbol{d}_f(\frac{h}{16}, \frac{w}{16}, 128)$ $\boldsymbol{d}_n(\frac{h}{16}, \frac{w}{16}, 256)$	$\boldsymbol{d}_{fuse}$ $(\frac{h}{16}, \frac{w}{16}, 128)$	Concatenation, Conv-(C128, K3, S1, P1), ReLU, residual × 6 (shown in Fig. 4)
Decoder	$(\frac{h}{16}, \frac{w}{16}, 128)$	$(\frac{h}{8}, \frac{w}{8}, 512)$	Deconv-(C512, K4, S2, P1), IN, ReLU
	$(\frac{h}{8}, \frac{w}{8}, 512)$	$(\frac{h}{4}, \frac{w}{4}, 256)$	Deconv-(C256, K4, S2, P1), IN, ReLU
	$(\frac{h}{4}, \frac{w}{4}, 256)$	$(\frac{h}{2}, \frac{w}{2}, 128)$	Deconv-(C128, K4, S2, P1), IN, ReLU
	$(\frac{h}{2}, \frac{w}{2}, 128)$	$(h, w, 64)$	Deconv-(C64, K4, S2, P1), IN, ReLU
	$(h, w, 64)$	$(h, w, 1)$	Deconv-(C1, K3, S1, P1), Sigmoid

d) BAUM-2i: BAUM-2i [25] is a static expression dataset, which is sorted out from BAUM-2, a dataset of audio-visual affective facial clips collected from movies and TV series. BAUM-2i contains samples under diverse conditions reflecting the eight expressions (six typical expressions plus the neutral and contemp). In our experiments, 998 images labeled with one of the seven expressions (contempt excluded) are used to evaluate our model.

e) RAF-DB: Real-world Affective Faces Database (RAF-DB) [26] is a large-scale facial expression database that all images are collected from the Internet and annotated by human. In our experiments, we only use the single-label subset (including six typical expressions and the neutral) to evaluate the TDGAN. The predefined training set includes 12271 samples and the size of the test set is 3068.

2) Experimental Settings: The great performance of inception layers in FER task has been proven in [27]. Therefore, we build a CNN-based architecture with inception blocks for the facial encoder and expressional encoder. For comparison, the expression discriminator of TDGAN is treated as the baseline of our model, which is denoted as CNN-Base in the following. Detailed network configurations of each module can be found in Table I and Table II.

In order to complete the task of expression transferring, the input images of the face branch should be labeled with identity. To this end, we choose CASIA-WebFace as the face dataset [28]. Since our main task is expression recognition instead of face recognition, we only utilize the images of the first 20 subjects in the CASIA-WebFace to train our model, where 2894 face images are involved in total. In the training stage, a face image and an expression image are randomly sampled from CASIA-WebFace and the expression dataset respectively, which forms the input image pair.

In the preprocessing stage, faces in the input images (including both face and expression images) are first detected by the MTCNN [29] and then resized to $128 \times 128 \times 1$. Data augmentation (random cropping and horizontal flipping) is also adopted in the training stage. To fairly compared with other methods, in CK+, TFEID, RaFD and BAUM-2i, we conduct subject-independent 10-fold cross-validation to evaluate our model, which is similar to the setting in [3], [4]. Concretely, for each fold of experiments on CK+, 1113 images are used for training and the remaining 123 images are for testing. In TFEID, the number of samples in training and testing set in each fold are 522 and 58 respectively. In each fold of RaFD, the training set contains 1260 images and the testing set contains 147 images. As for BAUM-2i, about 900 images are used for training and the remaining forms the testing set in each fold. In RAF-DB, TDGAN is trained and evaluated in the predefined training and testing sets. We run the model three times and report the averaged recognition accuracy. As for the adjustable parameters, we set $\lambda_{G_f} = 0.2$, $\lambda_{G_e} = 0.8$, $\lambda_{DIC} = 5$ and $\lambda_{per_f} = 1$.

B. Quantitative Results

1) Performance on In-the-Lab Datasets: We have evaluated TDGAN on the CK+, TFEID and RaFD datasets, where the averaged recognition accuracies reach 97.53%, 97.20% and 99.32% respectively. The confusion matrices are shown in Fig.5. From the confusion matrices, we can observe that TDGAN performs well when inputs are all captured under constrained environment. In addition, TDGAN is proficient in recognizing the expression of happiness, where the corresponding samples in these three datasets are identified well.

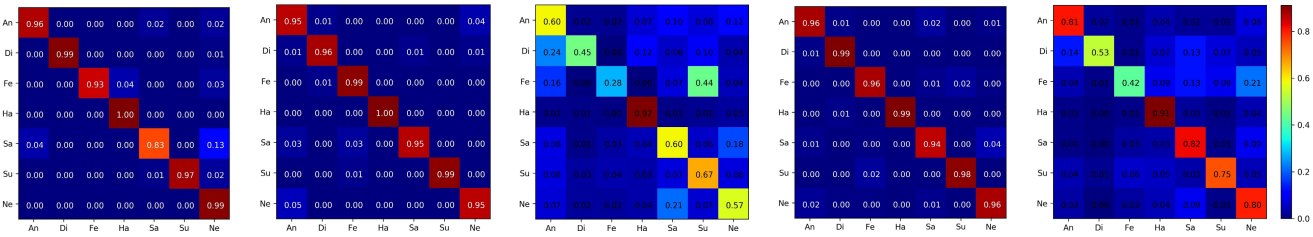
We also compare TDGAN with some state-of-the-art methods. Table III shows the comparison results. It can be observed that our TDGAN outperforms all other methods on these in-the-lab datasets. Note that for CK+, DTAGN [3] and DeRF [4] both extract temporal expressional information from

TABLE III

PERFORMANCE COMPARISON ON FIVE EXPRESSION DATASETS (%)

(The best performance is marked in bold.)

Dataset	CK+		TFEID		RaFD		BAUM-2i		RAF-DB	
Method	PHOG-LBP [30]	94.63	REC [31]	85.45	SURF [32]	90.64	WLD [33]	54.97	B-POOF [34]	73.19
	IACNN [35]	95.37	LMBP [36]	90.49	VisAtt [37]	93.10	LMP [33]	57.43	DLP-CNN [26]	74.20
	B-POOF [34]	95.70	MPC [38]	92.54	SVM [39]	94.51	LAP [33]	58.32	ORV [40]	80.00
	pACNN [41]	97.03	LGBPHS [33]	93.66	ELM [42]	96.94	LBP [33]	58.32	ASL [43]	84.63
	DTACN [3]	97.25	LTeP [33]	95.15	VGG [44]	98.33	LPQ [33]	58.99	gACNN [41]	85.07
	DeRF [4]	97.30	LAP [33]	95.15	ANN-Gabor [45]	99.15	SLPM [46]	63.84	3DFM [47]	**86.10**
	CNN-Base	96.45±2.18	CNN-Base	95.38±3.27	CNN-Base	98.64±1.47	CNN-Base	64.90±4.07	CNN-Base	77.25±1.19
	TDGAN	**97.53**±2.03	TDGAN	**97.20**±2.69	TDGAN	**99.32**±0.91	TDGAN	**65.76**±3.02	TDGAN	81.91±1.18



(a) CK+ (b) TFEID (c) RaFD (d) BAUM-2i (e) RAF-DB

Fig. 5. The confusion matrices on each dataset.

expression sequence to enhance their FER ability. In contrast, TDGAN is only trained and tested using static images. Even so, TDGAN still outperforms all other competitive methods, which demonstrates the efficiency of the proposed model. Additionally, compared with the CNN-Base model, TDGAN achieves higher averaged accuracy but with a smaller value of standard deviation. This fact shows that TDGAN acts with better effectiveness and robustness. The expression representations learned by the generator can be more discriminative than that of the CNN-Base, which verifies the efficiency of TDGAN in FER task.

2) Performance on In-the-Wild Datasets: The detailed performance of TDGAN on BAUM-2i and RAF-DB can be found in Fig.5, where an averaged recognition accuracy of 65.76% and 81.91% can be achieved. TDGAN performs the best when recognizing the expression of happiness, with an recognition rate over 90% in both datasets. However, the model seems insensitive to fear. Reasons can be twofold. On one hand, the sample number of fear is the least among all expressions, which accounts for only 6.81% in BAUM-2i and 2.29% in RAF-DB, while happiness accounts for more than 20% in BAUM-2i and even nearly 40% in RAF-DB. When facing such a huge gap of sample size among different expressions, knowledge is hard to be equally learned from different expressions. In other words, TDGAN may be biased by the imbalanced sample distribution, which results in learning less knowledge about fear. On the other hand, most samples in CASIA-WebFace are smiling faces and few fear faces are included. Since TDGAN can learn from both the face and expression datasets, the model may learn more knowledge about happiness and therefore relatively more discriminative to happiness than fear. Using a face dataset that with more evenly distributed expressions may partly mitigate the biases on different expressions.

We compare the performance of TDGAN with some state-of-the-art methods in Table III. Although BAUM-2i and RAF-DB are more challenging as variations such as poses and backgrounds are diverse, the proposed TDGAN still outperforms other existing models or is comparable to the state-of-the-art methods. There are three existing methods achieve higher recognition accuracy in RAF-DB. It is reasonable as the network architecture of ASL [43], gACNN [41] and 3DFM [47] is much deeper than that of TDGAN, which gives them a more powerful capacity to handle more complicated scenes. 3DFM [47] even ensembles three different models, which raises the performance but results in a higher computation cost. Even though, the performance of TDGAN on RAF-DB is far higher than that of the CNN-Base model, which can be owed to the effect of dual-branch architecture. With adversarial learning, the expression branch is forced to extract intrinsic expressional features, while other redundant information about face are all encoded into the other branch. This enables TDGAN to yield a disentangled and discriminative expressional representation for an input expression image. The performance gap between TDGAN and CNN-Base demonstrates the effectiveness of this network architecture.

3) Cross-Database Evaluation: We additionally conduct a cross-database experiment on CK+ and TFEID to evaluate TDGAN. Specifically, we train the model using CK+ and then test it on the TFEID, vice versa. We run the model three times and report the averaged recognition accuracy. The result is shown in Table IV. We can find that the performance of CNN-Base is far lower than that of the proposed TDGAN. This is reasonable as data distribution of these two datasets can be quite different in many aspects, e.g., subjects in CK+ are mostly Euro-American and Afro-American, while images of TFEID are all captured from Taiwanese. These

TABLE IV
RESULTS ON CROSS-DATABASE EVALUATION (%)

Train	Test	Method	Accuracy
CK+	TFEID	LTeP [33]	35.96
		LPQ [33]	38.16
		CNN-Baseline	61.83
		TDGAN (proposed)	**70.72**
TFEID	CK+	LGIP [33]	42.61
		LAP [33]	45.85
		CNN-Baseline	43.52
		TDGAN (proposed)	**64.54**

TABLE V
ABLATION RESULTS (%)

Dataset	CK+	TFEID	RaFD	BAUM-2i	RAF-DB
TDGAN-D	96.38	96.00	98.37	64.90	80.79
TDGAN	97.53	97.20	99.32	65.76	81.91

challenges may result in over-fitting of CNN-Base. In contrast, the proposed TDGAN is forced to disentangle expressional features from redundant facial variations, which finally yields a more discriminative expression representation with a better generation ability. The disentangled expression representation lowers the risk (or influence) of being over-fitting, which leads to a better performance on cross-database experiments.

4) Ablation Study: To evaluate the effect of the proposed DIC constraint, we additionally adopt an ablation experiment. Concretely, we train a model without the DIC constraint (let $\lambda_D = 0$), which is denoted as TDGAN-D. Experiments follow the same settings we aforementioned in Section IV-A2 and results are listed in Table V. It can be observed that the performance of TDGAN is always better when trained with the DIC constraint. This is tenable as the DIC constraint introduces a self-supervised manner into the model. The supervision provides a more explicit target for training, which encourages the model to further separate expressional features from other redundant variations. The results of the ablation experiment verify the effectiveness of the DIC constraint.

C. Visualization Analysis

1) Performance on Expression Transferring: Since TDGAN is trained through expression transferring, in Fig.7, we display some images that generated by our model using different expression datasets. As we can see in Fig.7, TDGAN is able to embed an expression to a specific face. Compared with the input face image, expressional regions in the generated image are modified by TDGAN with respect to different input expression images (e.g., the neighborhood of mouth for happiness and forehead for anger). This fact reflects that TDGAN has learnt to understand the information of different expressions. Note that the face dataset CASIA-WebFace and expression datasets BAUM-2i and RAF-DB are collected from unconstrained scenes, which contain various variations. For instance, the person in the input expression image of the fifth panel in Fig.7(d) shows a face with a downward angle. Similar situation can be found in many cases in Fig.7(e), where faces are with different poses and backgrounds. Even though,

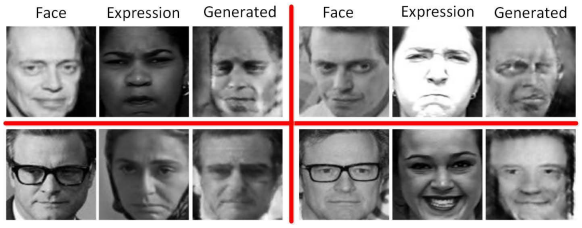


Fig. 6. Images generated without using perceptual loss. In each panel (separated by red solid lines), from left to right are the input face, input expression and the generated image respectively. Although the identity and expression are consistent with inputs, some other visual contents, such as poses (e.g., the generated images in the two top panels) or glasses (e.g., the generated images in the two bottom panels), get changed.

the expression of generated images are still consistent with the input expression image, which verifies that TDGAN is robust to pose variations.

We also evaluate the effect of the perceptual loss. To this end, we train the model without introducing perceptual loss and some generated images are displayed in Fig.6. We can observe that the identity of the input face image is preserved during expression transferring. However, some other semantic contents, e.g., accessaries or face poses, may be different or get lost. Compared with the images in Fig.7, we can find that the perceptual loss is effective to preserve semantic contents of the input face image.

2) Interpolation Experiments: One desirable character of TDGAN is to learn disentangled expression and face representations. This motivates us to synthesize some transitional images between two generated images by interpolation. To this end, we conduct two other experiments, i.e., expression interpolation and face interpolation, based on the learned expression and face representations. For expression interpolation, linear interpolation is conducted between two expression representations with coefficient α to generate a transitional expression representation, which can be formulated as:

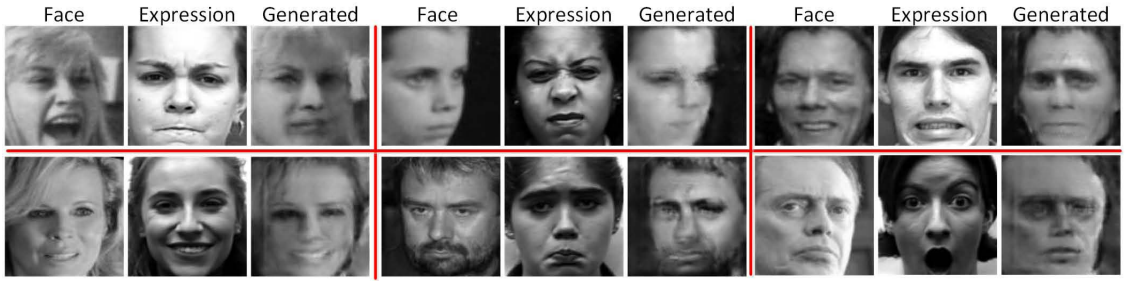
$$\boldsymbol{d}_{e_{int}} = (1-\alpha)\boldsymbol{d}_{e_1} + \alpha \boldsymbol{d}_{e_2}, \tag{11}$$

where $\boldsymbol{d}_{e_1}$ and $\boldsymbol{d}_{e_2}$ are representations of the source and target expression respectively. We generate transitional face representations $\boldsymbol{d}_{f_{int}}$ with another coefficient β following the same interpolation strategy:

$$\boldsymbol{d}_{f_{int}} = (1-\beta)\boldsymbol{d}_{f_1} + \beta \boldsymbol{d}_{f_2}, \tag{12}$$

where $\boldsymbol{d}_{f_1}$ and $\boldsymbol{d}_{f_2}$ are representations of the source face and target face respectively.

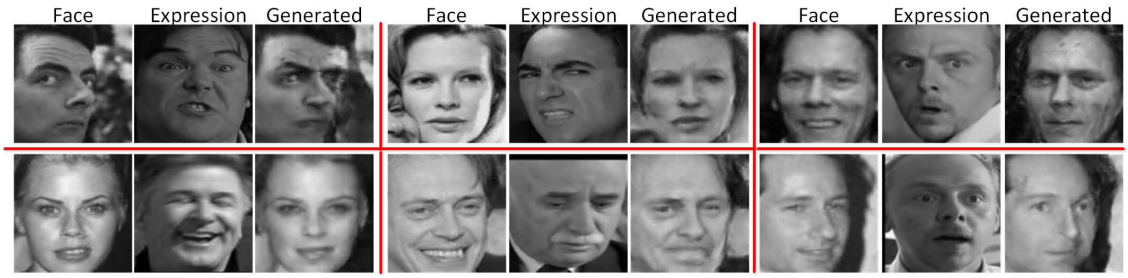
In expression interpolation, we linearly change the value of α to generate different transitional expression representations $\boldsymbol{d}_{e_{int}}$. We then replace $\boldsymbol{d}_e$ with different $\boldsymbol{d}_{e_{int}}$ but fix $\boldsymbol{d}_f$ and $\boldsymbol{d}_n$ in Equ. 3 and generate a series of transitional images with respect to different values of α. On the contrary, for face interpolation, β varies while keeping the expression representation unchanged. Some examples of the expression and face interpolation are shown in Fig. 8 and Fig. 9 respectively. From the results of expression interpolation (refer to Fig. 8), we can find that only expression is gradually changed if α varies, while other facial attributes (e.g., general face appearance and pose) are retained. Since the face representation is fixed



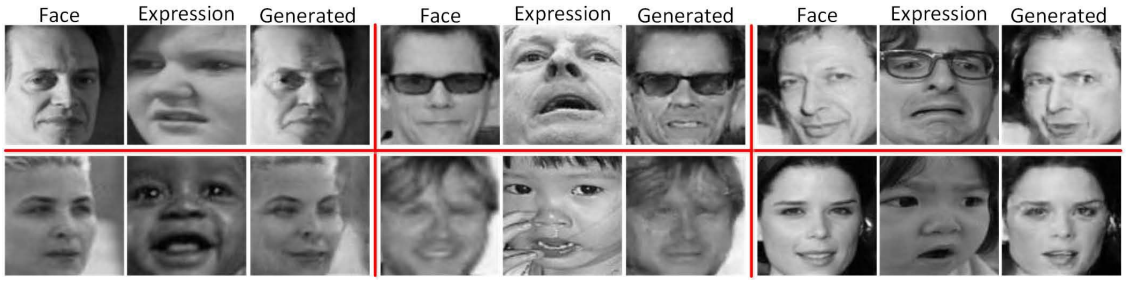
(a) Expression transferring using CK+.

(b) Expression transferring using TFEID.

(c) Expression transferring using RaFD.



(d) Expression transferring using BAUM2-i.



(e) Expression transferring using RAF-DB.

Fig. 7. Visualization of expression transferring using different expression datasets. Each sub-figure is divided into six panels (by red solid line), each of which shows a kind of expression transferring, i.e., anger, disgusting, fear, happiness, sadness and surprise (from left to right, top to down). There are three images in each panel: the left and the middle are the input face image (sampled from CASIA-WebFace) and expression image (sampled from the corresponding expression dataset) respectively, and the right one is the transferring result (generated image based on the two inputs).

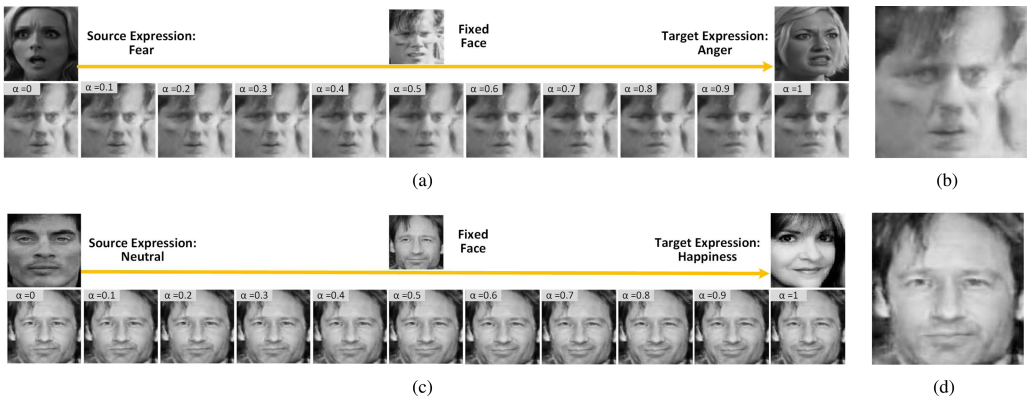


Fig. 8. Examples of expression interpolation. (a) and (c) are the interpolated expressions with respect to different values of α. (b) and (d) are the animations of the corresponding interpolation process. The animation figures are best viewed via Acrobat Reader on a desktop. Click the image to start the animation.

Fig. 9. Examples of face interpolation. (a) and (c) are the interpolated face with respect to different values of β. (b) and (d) are the animations of the corresponding interpolation process. Note that the input expressions in (a) and (c) are different. The animation figures are best viewed via Acrobat Reader on a desktop. Click the image to start the animation.

and the noise is uncorrelated to the input expression image, we can infer that changing the expression representation can only modify the generated expression instead of other parts of the face. This observation reflects that other facial attributes are isolated from expression. On the other hand, in face interpolation (refer to Fig. 9), different face representations result in different face appearance, while expressions are kept the same. This fact verifies that expression has already been isolated from other facial attributes. Obviously, the information of expression and other facial attributes has already been disentangled from each other by TDGAN. In other words, the learned expression representation reflects the intrinsic features of different expressions.

D. Model Analysis

From the above quantitative and qualitative results, we can clearly observe the effectiveness of TDGAN. However, some limitations of our model can also be observed during experiments. Here we make a deeper discussion on TDGAN.

1) Discussions on the Recognition Task: For our main task, i.e., expression recognition, we believe there is still room for TDGAN to make improvement. An intuitive way to improve the performance is to train with more face samples/subjects. However, since our model consists of two independent branches, we would like them to converge synchronously in the training stage so that the model can better

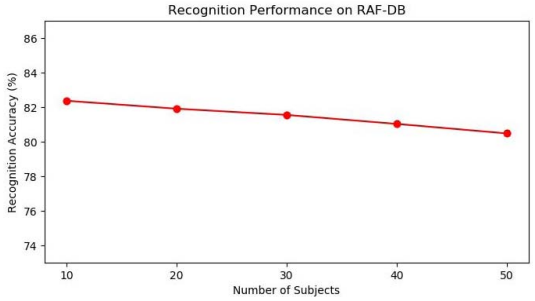


Fig. 10. The recognition accuracy on RAF-DB with different settings.

unify the learned knowledge from the the two branches. Compared with the expression branch, the face branch is more difficult to be trained from scratch as it is forced to cover all facial attributes other than expression. Therefore, the face branch converges much slower than the expression branch if we increase the number of subjects used in the experiments. Such asynchrony may lead the expression branch to overfitting before the face branch converge, which degrades the quality of the learned expression representations and result in an inferior performance on FER task. This prevents us using more subjects in the training.

In fact, we have trained our model on RAF-DB with different number of subjects and the result is shown in Fig. 10. We can observe that the recognition accuracy drops slightly

Fig. 11. Some failed cases of generated images. Each panel (separated by red solid lines) corresponds to an example. In each panel, the first and second column are the input face and expression images, and the third column is the generated image.

if more subjects are used, although the performance on expression transferring gets better. Since the main task of our work is expression classification, we make a trade-off between the performance of recognition and expression transferring. (This is the reason why we only use the first 20 subjects of CASIA-WebFace instead of the whole dataset to train our model.)

One possible solution for such a dilemma is to replace the face encoder with a pretrained face features extractor with deeper / elaborated architecture and fine-tune the extractor during training. This is because a deeper network has a larger capacity to capture more complicated facial patterns, which is suitable to train with more subjects [48], [49]. On the other hand, fine-tuning a pretrained model (instead of training from scratch) makes it easier to train the model, which helps synchronize the convergency of the two branches. This can be left as our future work.

2) Discussions on Transferring Performance: For our auxiliary task, i.e., expression transferring, there is still room for improvement. One observation is that expression transferring is not always successfully made. Typical failed cases are shown in Fig.11.

Some factors may account for the failures. For example, TDGAN may fail to generate an ideal image when the input face is with an extreme pose (the first row of Fig. 11). Large areas of occlusions in the input face can also result in failure generation (the third row of Fig. 11). Since there are few samples that with extreme poses or large occlusions in the training set, TDGAN can hardly learn much knowledge on how to deal with such situations. Furthermore, when the occlusion cover some key regions that related to an expression (e.g., the neighborhood of brow for anger), TDGAN is not only required to render the correct expression texture, but also complete the face content that being occluded. This make it more difficult to successfully transfer the expression.

Another factor is the image quality of the inputs. For example, images with low contrast ratio (the left panel of the second row in Fig.11) or blur (the right panel of the second row in Fig. 11) can lead to the failure. Detailed information of face or expressions is hard to extract from such images, which makes it hard to generate an ideal image. Also, images contained multiple faces (such as the input face image in the left panel of the second row in Fig. 11) may confuse the model.

One other case different from all aforementioned examples is the transferring of surprise, as shown in the fourth row in Fig. 11. We can observe that TDGAN tries to modify some facial regions correlated to the expression of surprise, which means the model 'understands' the input expression and know where should be modified. However, it fails to render a realistic mouth in both cases. This may due to the lack of surprise-like samples in our face training set. For some facial textures that form a prototypical expression, the knowledge of them can be learnt from many commonly seen faces. For example, the texture of frown, which is a basic character of anger and disgust, is common in many input face images. However, characters like a big opening mouth are specific for a surprise face, which is rare in the face dataset. In other words, the decoder in the generator learns little knowledge about how a surprise expression of a specific face should like, which results in generating inferior images in some cases. A direct way to solve this problem is to use more images with surprise faces in the training. Another solution is to make use of low-level (texture information) expression features of the input expression image and incorporate them into the generation process (e.g., introduce some skip-connection from lower layers of the expression encoder into the decoder). However, the data distribution between face and expression datasets can be with much difference or even with different modality. This requires us to introduce other techniques, such as cross-modality fusion, to deal with the problem. Since the main task of this work is expression recognition, we leave it as our future work.

V. Conclusion

In this paper, we present a novel model named Two-branch Disentangled Generative Adversarial Network (TDGAN) for FER task. Different from other GAN-based model, we separate the generator into the face and expression branch. Following adversarial learning, TDGAN learns to disentangle the intrinsic expression information from other facial attributes, which encourages it to yield a more discriminative expression representation for FER. In addition, TDGAN is able to generate images with basic expressions, which makes it possible to conduct expression transferring. Experiments are conducted on both in-the-lab and in-the-wild datasets, where TDGAN achieves the state-of-the-art or comparable performances to most existing methods. As our model is a general framework, it can be extended to other task like face recognition or image translation, which are left as our future work.

Acknowledgment

The authors would like to acknowledge Dr. Shiyuan Li for the help on visualization experiments.

REFERENCES

[1] A. Majumder, L. Behera, and V. K. Subramanian, "Emotion recognition from geometric facial features using self-organizing map," *Pattern Recognit.*, vol. 47, no. 3, pp. 1282–1293, Mar. 2014.

[2] Y. Tong, R. Chen, J. Yang, and M. Wu, "Robust facial expression recognition based on local tri-directional coding pattern," in *Proc. 12th Int. Conf. Complex, Intell., Softw. Intensive Syst. Complex, Intell., Softw. Intensive Syst. (CISIS)*, 2018, pp. 606–614.

[3] H. Jung, S. Lee, J. Yim, S. Park, and J. Kim, "Joint fine-tuning in deep neural networks for facial expression recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 2983–2991.

[4] H. Yang, U. Ciftci, and L. Yin, "Facial expression recognition by de-expression residue learning," in *Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 2168–2177.

[5] S. Xie, H. Hu, and Y. Wu, "Deep multi-path convolutional neural network joint with salient region attention for facial expression recognition," *Pattern Recognit.*, vol. 92, pp. 177–191, Aug. 2019.

[6] A. Majumder, L. Behera, and V. K. Subramanian, "Automatic facial expression recognition system using deep network-based data fusion," *IEEE Trans. Cybern.*, vol. 48, no. 1, pp. 103–114, Jan. 2018.

[7] S. Xie and H. Hu, "Facial expression recognition using hierarchical features with deep comprehensive multipatches aggregation convolutional neural networks," *IEEE Trans. Multimedia*, vol. 21, no. 1, pp. 211–220, Jan. 2019.

[8] Z.-H. Jiang, Q. Wu, K. Chen, and J. Zhang, "Disentangled representation learning for 3D face shape," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 11957–11966.

[9] T. Zhang, H. Wang, and Q. Dong, "Deep disentangling siamese network for frontal face synthesis under neutral illumination," *IEEE Signal Process. Lett.*, vol. 25, no. 9, pp. 1344–1348, Sep. 2018.

[10] T. Hinz and S. Wermter, "Image generation and translation with disentangled representations," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2018, pp. 1–8.

[11] I. J. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Conf. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[12] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," 2015, *arXiv:1511.06434*. [Online]. Available: http://arxiv.org/abs/1511.06434

[13] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2242–2251.

[14] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 8789–8797.

[15] A. Pumarola, A. Agudo, A. M. Martinez, A. Sanfeliu, and F. Moreno-Noguer, "Ganimation: Anatomically-aware facial animation from a single image," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 835–851.

[16] L. Tran, X. Yin, and X. Liu, "Disentangled representation learning GAN for pose-invariant face recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1283–1292.

[17] H. Ding, K. Sricharan, and R. Chellappa, "Exprgan: Facial expression editing with controllable expression intensity," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 6781–6788.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[19] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 694–711.

[20] W. Shen and R. Liu, "Learning residual images for face attribute manipulation," in *Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1225–1233.

[21] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2414–2423.

[22] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, "The extended Cohn-Kanade dataset (CK+): A complete dataset for action unit and emotion-specified expression," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR) Workshops*, Jun. 2010, pp. 94–101.

[23] L. Chen and Y. Yen, "Taiwanese facial expression image database," Brain Mapping Lab., Inst. Brain Sci., Nat. Yang-Ming Univ., Taipei, Taiwan, Tech. Rep., 2007. [Online]. Available: http://bml.ym.edu.tw/tfeid/modules/wfdownloads/

[24] O. Langner, R. Dotsch, G. Bijlstra, D. H. J. Wigboldus, S. T. Hawk, and A. van Knippenberg, "Presentation and validation of the radboud faces database," *Cognition Emotion*, vol. 24, no. 8, pp. 1377–1388, Dec. 2010.

[25] C. Eroglu Erdem, C. Turan, and Z. Aydin, "BAUM-2: A multilingual audio-visual affective face database," *Multimedia Tools Appl.*, vol. 74, no. 18, pp. 7429–7459, Sep. 2015.

[26] S. Li and W. Deng, "Reliable crowdsourcing and deep locality-preserving learning for unconstrained facial expression recognition," *IEEE Trans. Image Process.*, vol. 28, no. 1, pp. 356–370, Jan. 2019.

[27] A. Mollahosseini, D. Chan, and M. H. Mahoor, "Going deeper in facial expression recognition using deep neural networks," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2016, pp. 1–10.

[28] D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Learning face representation from scratch," *CoRR*, vol. abs/1411.7923, 2014.

[29] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Process. Lett.*, vol. 23, no. 10, pp. 1499–1503, Oct. 2016.

[30] S. L. Happy and A. Routray, "Robust facial expression classification using shape and appearance features," in *Proc. 8th Int. Conf. Adv. Pattern Recognit. (ICAPR)*, Jan. 2015, pp. 1–5.

[31] W. F. Gu, Y. V. Venkatesh, and C. Xiang, "A novel application of self-organizing network for facial expression recognition from radial encoded contours," *Soft Comput.*, vol. 14, no. 2, pp. 113–122, Jan. 2010.

[32] Q. Rao, X. Qu, Q. Mao, and Y. Zhan, "Multi-pose facial expression recognition based on SURF boosting," in *Proc. Int. Conf. Affect. Comput. Intell. Interact. (ACII)*, Sep. 2015, pp. 630–635.

[33] C. Turan and K.-M. Lam, "Histogram-based local descriptors for facial expression recognition (FER): A comprehensive study," *J. Vis. Commun. Image Represent.*, vol. 55, pp. 331–341, Aug. 2018.

[34] Z. Liu, S. Li, and W. Deng, "Boosting-POOF: Boosting part based one vs one feature for facial expression recognition in the wild," in *Proc. 12th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, Jun. 2017, pp. 967–972.

[35] Z. Meng, P. Liu, J. Cai, S. Han, and Y. Tong, "Identity-aware convolutional neural network for facial expression recognition," in *Proc. 12th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, May 2017, pp. 558–565.

[36] M. M. Goyani and N. Patel, "Recognition of facial expressions using local mean binary pattern," *ELCVIA: Electron. Lett. Comput. Vis. image Anal.*, vol. 16, no. 1, pp. 54–67, 2017.

[37] G.-C. Luh, H.-B. Wu, Y.-T. Yong, Y.-J. Lai, and Y.-H. Chen, "Facial expression based emotion recognition employing YOLOv3 deep neural networks," in *Proc. Int. Conf. Mach. Learn. Cybern. (ICMLC)*, Jul. 2019, pp. 1–7.

[38] N. Farajzadeh, G. Pan, and Z. Wu, "Facial expression recognition based on meta probability codes," *Pattern Anal. Appl.*, vol. 17, no. 4, pp. 763–781, Nov. 2014.

[39] G. L. Libralon and R. A. F. Romero, "Investigating facial features for identification of emotions," in *Proc. Int. Conf. Neural Inf. Process.* Springer, 2013, pp. 409–416.

[40] V. Vielzeuf, C. Kervadec, S. Pateux, A. Lechervy, and F. Jurie, "An Occam's razor view on learning audiovisual emotion recognition with small training sets," in *Proc. Int. Conf. Multimodal Interact. (ICMI)*, 2018, pp. 589–593.

[41] Y. Li, J. Zeng, S. Shan, and X. Chen, "Occlusion aware facial expression recognition using CNN with attention mechanism," *IEEE Trans. Image Process.*, vol. 28, no. 5, pp. 2439–2450, May 2019.

[42] B. Islam, F. Mahmud, and A. Hossain, "Facial region segmentation based emotion recognition using extreme learning machine," in *Proc. Int. Conf. Advancement Electr. Electron. Eng. (ICAEEE)*, Nov. 2018, pp. 1–4.

[43] P. Jiang, G. Liu, Q. Wang, and J. Wu, "Accurate and reliable facial expression recognition using advanced softmax loss with fixed weights," *IEEE Signal Process. Lett.*, vol. 27, pp. 725–729, 2020.

[44] I. Oztel, G. Yolcu, and C. Oz, "Performance comparison of transfer learning and training from scratch approaches for deep facial expression recognition," in *Proc. 4th Int. Conf. Comput. Sci. Eng. (UBMK)*, Sep. 2019, pp. 1–6.

[45] B. Islam, F. Mahmud, A. Hossain, M. S. Mia, and P. B. Goala, "Human facial expression recognition system using artificial neural network classification of Gabor feature based facial expression information," in *Proc. 4th Int. Conf. Electr. Eng. Inf. Commun. Technol. (iCEEiCT)*, Sep. 2018, pp. 364–368.

[46] C. Turan, K. Lam, and X. He, "Soft locality preserving map (SLPM) for facial expression recognition," Jan. 2018, *arXiv:1801.03754*. [Online]. Available: https://arxiv.org/abs/1801.03754
[47] S. T. Ly, N.-T. Do, G.-S. Lee, S.-H. Kim, and H.-J. Yang, "A 3D face modeling approach for in-the-wild facial expression recognition on image datasets," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2019, pp. 3492–3496.
[48] X. Wu, R. He, Z. Sun, and T. Tan, "A light CNN for deep face representation with noisy labels," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 11, pp. 2884–2896, Nov. 2018.
[49] K. Cao, Y. Rong, C. Li, X. Tang, and C. C. Loy, "Pose-robust face recognition via deep residual equivariant mapping," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5187–5196.

Siyue Xie received the B.E. degree from the College of Science and Engineering, Jinan University, China, in 2016, and the M.E. degree from the School of Electronics and Information Technology, Sun Yat-sen University, China, in 2019. He is currently pursuing the Ph.D. degree with the Department of Information Engineering, The Chinese University of Hong Kong. His major research interests include computer vision and pattern recognition.

Haifeng Hu (Member, IEEE) received the Ph.D. degree from Sun Yat-sen University in 2004. He is currently a Professor with the School of Electronics and Information Technology, Sun Yat-sen University. He has published over 140 articles since 2000. His research interests include computer vision, pattern recognition, image processing, and neural computation.

Yizhen Chen received the B.E. degree in communication engineering from Sun Yat-sen University, Guangzhou, China, in 2018, where he is currently pursuing the M.E. degree in information and communication engineering with the School of Electronics and Information Technology. His current research interests include computer vision, facial expression recognition, and semantic segmentation.