Dual Learning for Facial Action Unit Detection Under Nonfull Annotation

Shangfei Wang, *Senior Member, IEEE*, Heyan Ding, and Guozhu Peng

***Abstract*—Most methods for facial action unit (AU) recognition typically require training images that are fully AU labeled. Manual AU annotation is time intensive. To alleviate this, we propose a novel dual learning framework and apply it to AU detection under two scenarios, that is, semisupervised AU detection with partially AU-labeled and fully expression-labeled samples, and weakly supervised AU detection with fully expression-labeled samples alone. We leverage two forms of auxiliary information. The first is the probabilistic duality between the AU detection task and its dual task, in this case, the face synthesis task given AU labels. We also take advantage of the dependencies among multiple AUs, the dependencies between expression and AUs, and the dependencies between facial features and AUs. Specifically, the proposed method consists of a classifier, an image generator, and a discriminator. The classifier and generator yield face–AU–expression tuples, which are forced to coverage of the ground-truth distribution. This joint distribution also includes three kinds of inherent dependencies: 1) the dependencies among multiple AUs; 2) the dependencies between expression and AUs; and 3) the dependencies between facial features and AUs. We reconstruct the inputted face and AU labels and introduce two reconstruction losses. In a semisupervised scenario, the supervised loss is also incorporated into the full objective for AU-labeled samples. In a weakly supervised scenario, we generate pseudo paired data according to the domain knowledge about expression and AUs. Semisupervised and weakly supervised experiments on three widely used datasets demonstrate the superiority of the proposed method for AU detection and facial synthesis tasks over current works.**

***Index Terms*—Adversarial learning, dual learning, facial action unit (AU) detection, semisupervised, weakly supervised.**

I. Introduction

FACIAL behavior analysis is one of the fastest growing research areas in affective computing and computer vision research. We can learn about people's emotions through facial behavior. There are two commonly used ways to describe facial behavior: 1) facial expression and 2) facial action unit (AU). Facial expression is an intuitive description of facial behavior, most commonly identified as one or more of six expressions (i.e., anger, disgust, fear, happiness, sadness, and surprise) [1]. These labels, however, are not complex enough to describe the full range of emotions. There are many other expressions, including pain, awe, embarrassment, hatred, etc. The number and definition of expressions are not universally agreed upon by researchers.

AUs are patterns of muscular activation as defined in Ekman's facial action coding system (FACS) [2]. Compared to expressions, which describe the global facial behavior, AUs describe facial behavior in more detail and subtlety. AU detection is a basic affective computing problem that has been studied for decades [3]–[5]. The successful recognition of AUs could greatly assist the analysis of human facial behavior and expression. Traditional supervised AU detection methods need a large number of AU-annotated facial images. However, AUs represent subtle local facial changes and, thus, should be annotated by experts. AU labeling is time consuming and expensive. To reduce reliance on AU labels, we propose semisupervised and weakly supervised AU detection methods, in which we train AU classifiers from images with expressions and partial AUs, and images annotated with expressions only, respectively.

Expression labels are easier and less time consuming to annotate than AUs. Expressions are also strongly associated with AUs. For example, Du *et al.* [6] found that people almost always lower their jaws (AU26) when they show surprise, and the lip corner puller (AU12) rarely appears in sad faces. Many expression-dependent AU combinations are detailed in the emotion FACS (EMFACS) [7]. Prkachin and Solomon [8] found that pain intensity is mainly determined by six AUs (AU4, AU6, AU7, AU9, AU10, and AU43). In addition, multiple AUs are closely related because of the structure and anatomy of the human face. Each AU is controlled by at least one facial muscle. For example, the inner and outer brow raisers (AU1 and AU2) typically appear together, since they are both related to the muscle group frontalis. Lip corner puller (AU12) and lip corner depressor (AU15) rarely appear together, since they are formed by the contraction or relaxation

Manuscript received October 16, 2019; revised March 13, 2020; accepted June 14, 2020. This work was supported in part by the projects from Anhui Science and Technology Agency under Grant 1804a09020038; and in part the National Natural Science Foundation of China under Grant 91748129 and Grant 61727809. This article was recommended by Associate Editor W. Hu. *(Corresponding author: Shangfei Wang.)*

Shangfei Wang is with the Key Laboratory of Computing and Communication Software of Anhui Province, School of Computer Science and Technology, School of Data Science, University of Science and Technology of China, Hefei 230027, China (e-mail: sfwang@ustc.edu.cn).

Heyan Ding is with the School of Data Science, University of Science and Technology of China, Hefei 230027, China (e-mail: dhy0513@mail.ustc.edu.cn).

Guozhu Peng is with the Key Laboratory of Computing and Communication Software of Anhui Province, School of Computer Science and Technology, University of Science and Technology of China, Hefei 230027, China (e-mail: gzpeng@mail.ustc.edu.cn).

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

Digital Object Identifier 10.1109/TCYB.2020.3003502

of the same muscle. These dependencies exist whether or not a facial image has been annotated. They are crucial for learning the AU classifier, especially when samples are partially annotated or not annotated at all.

Many machine-learning tasks emerge in dual form [9]. For AU detection, the dual task is facial synthesis using the AU labels. There are intrinsic probabilistic connections between the recognition of AUs and the facial synthesis tasks, and such connections are independent of annotation status. Dual tasks help each other when they are trained together. However, there has not been much research on the simultaneous recognition of AUs and face synthesis. In this article, we leverage the connections between these tasks to improve the results of both. Generative adversarial network (GAN) [10] is also used in many fields [11], [12]. In this article, we also use GAN to learn joint distributions.

We suggest a dual GAN (DGAN) to jointly learn an AU classifier and a face generator. DGAN regularizes the learning process by exploring the probabilistic duality between the dual tasks [9]. The joint distribution of the input and output of the primal and dual models should be equal to the distribution of the paired data. DGAN utilizes an adversarial framework to force convergence between the joint distribution of the input and output of the AU classifier and face generator and the distribution of the paired face–AUs data. Such distribution includes not only the dependencies among multiple AUs but also dependencies between facial features and AUs. Furthermore, we leverage the assistance of expression labels as extra conditional information. In addition, we reconstruct the inputted face or AU labels and introduce two reconstruction losses since the AU classifier and face generator form a closed loop. We apply DGAN to semisupervised and weakly supervised learning scenarios. In the former scenario, we also utilize supervised loss for AU-labeled samples. There are no face–AUs data in the latter scenario, so we generate pseudo paired data through domain knowledge about expression and AUs.

A preliminary version of this article appeared as [13], in which a semisupervised dual learning framework for AU detection with partially labeled data is proposed. Compared to the previous version, the present version applies the proposed dual learning framework to weakly supervised AU detection scenarios when only expression-labeled data are available. We also add experiments and comparisons of the weakly supervised AU detection task on three benchmark sets of data, demonstrating the effectiveness of the proposed method in this scenario.

This article is organized in the following manner. Section II reviews relevant works on dual learning and AU detection without full AU annotations. In Section III, we present the dual learning method DGAN. In Sections IV and V, we apply the proposed DGAN to semisupervised and weakly supervised AU detection problems, respectively. In Section VI, we conduct experiments on three benchmark databases. Finally, Section VII summarizes our work.

II. Related Work

A. Dual Learning

Machine-learning tasks often have a primal and dual form [9]. For example, an image generation task is the dual task of an image classification task. The primal task and the dual task form a closed loop, generating informative feedback signals that benefit both tasks.

He *et al.* [14] first proposed unsupervised dual learning for neural machine translation with unpaired monolingual corpora. They trained two translators through a reinforcement learning process. In this process, training consists of two agents: each understands one language. One sentence is translated from the first agent by the primal translator and then sent to the second agent. The sentence is then evaluated by the second agent and returned to the first by the dual translator. The two translators can be iteratively updated based on the feedback signals (i.e., the pretrained language model likelihood of the translator output and the post-translation reconstruction error of the original sentence) until convergence.

Yi *et al.* [15] proposed another unsupervised dual learning method, applying a GAN to image-to-image translation. They introduced two image discriminators as the evaluation model for two image domains. Unlike He *et al.*'s work [14], in which models are pretrained, two image discriminators and two image translators are simultaneously trained using adversarial learning.

Unlike the aforementioned unsupervised dual learning methods, Xia *et al.* [9] proposed dual supervised leaning (DSL) from paired data. They minimized the empirical risk of dual tasks under a necessary condition, that is, the probabilistic duality between the dual tasks. They explored probabilistic duality by minimizing the KL-divergence of the joint distribution of the inputs and outputs of the primal and dual models.

The above three works explore dualities at the data level. In contrast, Xia *et al.* [16] suggested a model-level dual learning method to explore dualities between the dual tasks by sharing partial parameters of dual models.

In Xia *et al.*'s work [9], the marginal distribution of the input is estimated to represent the joint distribution of the input and output. This may lead to errors in the learning process. To avoid the estimation of marginal distribution, in this article, we adopt an adversarial manner to make the input and output joint distributions of the primal and dual models close by forcing them to converge to the distribution of the paired data. Unlike DSL [9], we introduce reconstruction loss.

Until now, there has not been any research on dual learning in semisupervised scenarios or that applies dual learning to AU detection. Here, we formulate AU detection and face synthesis as dual tasks, which can be trained at the same time. The proposed dual learning method for this face–AU dual task works when there are only partially AU-annotated samples or samples lacking AU annotation entirely.

B. AU Detection

The background and details of facial AU recognition are detailed in [17]–[19]. This section is limited in scope to recognition methods that learn the AU classifier from images without full AU annotations. Specifically, we briefly review the semisupervised and weakly supervised AU detection works.

1) Semisupervised Works: The current semisupervised AU detection works can be categorized into two approaches based on whether or not they use the help of expression labels.

For semisupervised AU detection without expressions, missing AUs are handled using label smoothness or AU dependencies. Niu *et al.* [20] proposed a multilabel co-regularization method for semisupervised facial AU recognition via co-training. This method uses the graph convolutional network (GCN) to embed AU relationships and optimizes multiview feature generation and AU classification via multiview loss and multilabel co-regularization loss. Song *et al.* [21] suggested a Bayesian group-sparse-compressed sensing (BGCS) model to encode AU co-occurrence structure and sparsity for AU detection. During the inference procedure, this method marginalizes over the unobserved values to handle partially observed labels. Wu *et al.* [22] put forth a multilabel learning method with missing labels (MLMLs). They handled the missing labels by examining how consistent the predicted labels were with the provided labels. They also looked at local smoothness among the assigned labels. Unlike the above work, which uses the same features for all AUs, Li *et al.* [23] proposed an improved version of MLML that classified each AU using the most related features. Wu *et al.* [24] adopted a restricted Boltzmann machine (RBM) model with the goal of capturing the joint distribution of all AUs using the given AU labels as the prior distribution. This was accomplished by minimizing errors between predicted and ground truth AUs for AU-labeled samples while simultaneously maximizing the log likelihood of the AU classifier with regard to the learned prior distribution.

In semisupervised AU detection scenarios enhanced by expressions, dependencies between AUs and expressions are exploited to handle missing AUs. A Bayesian network (BN) was proposed by Wang *et al.* [25] to capture the relations among AUs and the relations between facial expressions and AUs. Hidden knowledge in the form of expression labels complement any AU labels that are missing. In the testing phase, the AU–expression relationships encoded in the BN are combined with the AU measurements obtained from a basic classifier (SVM) to infer the AU labels. The previous version of this article [13] proposed a dual semisupervised GAN (DSGAN) for semisupervised AU detection. The expression labels act as auxiliary information and are fed into the discriminator. This article extends DSGAN to weakly supervised AU detection scenarios.

2) Weakly Supervised Scenarios: The above AU detection methods successfully reduce the reliance on AU labels but still depend on AU-labeled samples. Recently, some AU detection methods learned AU classifiers when expression-labeled samples are available but AU-labeled samples are not. All of them utilized domain knowledge about expressions and AUs. Some works used domain knowledge as the constraints and incorporated some losses into the full objective; others used domain knowledge to generate pseudo AU labels under each expression category. All of these weakly supervised methods can be extended to semisupervised methods by adding supervised loss for AU-labeled samples.

Specifically, Ruiz *et al.* [26] suggested hidden-task learning (HTL), which uses facial image features to develop the AU classifier. They introduced the visible task to learn the expression classifier from AUs in advance. Pseudo samples generated under the expression condition act as the training set of the visible task. They concatenated two tasks as hidden–visible tasks and used expression labels to update the parameters of the hidden task. They extended HTL to semi-HTL (SHTL), assuming that the AU labels of partial samples are also provided. Visible and hidden tasks are trained separately, so any errors made by the expression classifier affect the AU classifier as well. Furthermore, HTL only uses the probability of one AU, given expressions, and requires an extra-large set of expression-annotated facial images.

Wang *et al.* [27] proposed a similar model to [24]. They suggested an RBM prior (RBM-P) model to learn the prior joint distribution of all AUs. Unlike [24], Wang *et al.* learned the prior distribution under each expression and used the generated pseudo AU labels according to the domain knowledge. Like HTL, RBM-P leverages the probability of AUs under each expression.

Peng and Wang [28] used the same domain knowledge as Wang *et al.* [27], and also sampled the pseudo samples under each expression. They employed adversarial learning to minimize the distance between the predicted AU label distribution and the pseudo AU label distribution, thus avoiding any error caused by the estimation of AU prior distribution.

Zhang *et al.* [4] put forth a multiple AU classifier learning method (LP-SM) using SVM as the basic classifier. They introduced five kinds of extra loss according to the inequality relations among the AU probabilities and incorporated them into the full objective. They used an iterative optimization algorithm to learn multiple AU classifiers and AU labels of training samples simultaneously.

Wang *et al.* [29] leveraged the rank relations among the probabilities of all AUs, given expression. They formulated weakly supervised AU detection as a multilabel ranking problem, proposing a rank loss for it. They only considered relations between expressions and AUs, and the rank relations among the AU probabilities miss a lot of useful information.

Compared to related works, this work offers the following primary contributions.

1) We formulate a novel dual learning framework that explores the probabilistic duality through a GAN, which is proven a necessary condition [9] when training the primal and dual models using paired data. The proposed method avoids the errors caused by the estimation of the marginal distribution.
2) Currently, there are few works considering both AU detection and face synthesis. The proposed dual learning method is applied to AU detection in semisupervised and weakly supervised scenarios.
3) The proposed method is evaluated on widely used benchmark datasets. At the submission of this article, the results of our experiments exceed those of other methods.

III. Dual Generative Adversarial Network

Given two spaces $\mathcal{X}$ and $\mathcal{Y}$, a general dual learning scheme can be created as the primal task $C : \mathcal{X} \to \mathcal{Y}$, which maps a sample taken from space $\mathcal{X}$ to space $\mathcal{Y}$, and its dual task $G : \mathcal{Y} \to \mathcal{X}$, which maps a sample taken from space $\mathcal{Y}$ to space $\mathcal{X}$.

Let $(\mathbf{x}, \mathbf{y})$ denote the paired data in $\mathcal{X} \times \mathcal{Y}$. Inputting $\mathbf{x}$ to C, we can obtain $\hat{\mathbf{y}} = C(\mathbf{x})$, and $(\mathbf{x}, \hat{\mathbf{y}}) \in \mathcal{X} \times \hat{\mathcal{Y}}$. Similarly, inputting $\mathbf{y}$ to G can obtain $\hat{\mathbf{x}} = G(\mathbf{y})$, and $(\hat{\mathbf{x}}, \mathbf{y}) \in \hat{\mathcal{X}} \times \mathcal{Y}$. Let P denote the distribution of paired data in $\mathcal{X} \times \mathcal{Y}$. P_C signifies the distribution of data in $\mathcal{X} \times \hat{\mathcal{Y}}$. P_G denotes the distribution of data in $\hat{\mathcal{X}} \times \mathcal{Y}$. According to the probabilistic duality [9], for any paired data $(\boldsymbol{x}, \boldsymbol{y})$, the primal model C and the dual model G should make the following equality valid:

$$P(\boldsymbol{x}, \boldsymbol{y}) = P_C(\boldsymbol{x}, \boldsymbol{y}) = P_G(\boldsymbol{x}, \boldsymbol{y}). \tag{1}$$

In order to avoid the estimation of the marginal distribution of data in $\mathcal{X}$ or $\mathcal{Y}$, we propose a DGAN to make both P_C and P_G converge to P. Specifically, we introduce a discriminator D. $(\mathbf{x}, \hat{\mathbf{y}}) \in \mathcal{X} \times \hat{\mathcal{Y}}$ and $(\hat{\mathbf{x}}, \mathbf{y}) \in \hat{\mathcal{X}} \times \mathcal{Y}$, generated by C and G, respectively, are regarded as "fake" and sent to discriminator D for judgment. $(\mathbf{x}, \mathbf{y}) \in \mathcal{X} \times \mathcal{Y}$ is regarded as "real" and also sent to D. Then, the adversarial loss of DGAN is as follows:

$$\begin{aligned}\mathcal{L}_{\text{adv}} &= \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{X}\times\mathcal{Y}}\big[\log D(\mathbf{x}, \mathbf{y})\big] \\ &\quad + \alpha\mathbb{E}_{(\mathbf{x},\hat{\mathbf{y}})\sim\mathcal{X}\times\hat{\mathcal{Y}}}\big[\log\big(1 - D(\mathbf{x}, \hat{\mathbf{y}})\big)\big] \\ &\quad + (1-\alpha)\mathbb{E}_{(\hat{\mathbf{x}},\mathbf{y})\sim\hat{\mathcal{X}}\times\mathcal{Y}}\big[\log\big(1 - D(\hat{\mathbf{x}}, \mathbf{y})\big)\big]\end{aligned} \tag{2}$$

where $\alpha \in (0, 1)$ weighs the importance of the distribution of data in $\mathcal{X} \times \hat{\mathcal{Y}}$ in the mixed distribution.

Since the primal and dual models form a closed loop, they impose constraints on one another. In consideration of this, we introduce two reconstruction losses as Yi *et al.* [15] and Zhu *et al.* [30] did. $\hat{\mathbf{y}}$ is inputted into G and the output $G(\hat{\mathbf{y}})$ is the reconstruction of $\mathbf{x}$. Similarly, $\hat{\mathbf{x}}$ is inputted into C and the output $C(\hat{\mathbf{x}})$ is the reconstruction of $\mathbf{y}$. The reconstruction loss for C ($\mathcal{L}^c_{\text{rec}}$) and G ($\mathcal{L}^g_{\text{rec}}$) is as follows:

$$\begin{aligned}\mathcal{L}^c_{\text{rec}} &= \mathbb{E}_{\mathbf{x}\sim\mathcal{X}}\text{Dis}(\mathbf{x}, G(C(\mathbf{x}))) \\ \mathcal{L}^g_{\text{rec}} &= \mathbb{E}_{\mathbf{y}\sim\mathcal{Y}}\text{Dis}(\mathbf{y}, C(G(\mathbf{y})))\end{aligned} \tag{3}$$

where Dis is the distance measurement, which is different for C and G, and varies by the specific task.

Compared to other recent dual learning frameworks, DualGAN [15] is the most similar to the proposed DGAN as both are based on GAN. However, there are some significant differences. First, DualGAN is an unsupervised dual learning framework that trains with unpaired data in two domains, while DGAN explores the duality inherent in paired data. Second, DualGAN contains two discriminators corresponding to two data domains, while DGAN only has one discriminator and the input is the paired data. Third, through adversarial learning, DualGAN makes the distribution of the generated data from one domain converge with the distribution of the true data of another domain. DGAN forces the distributions of the input and output of the primal and dual models to converge to the distribution of the true paired data. Finally, DualGAN only uses reconstruction loss to explore the duality between the dual tasks, while DGAN also explores the probabilistic duality between the dual tasks.

IV. Semisupervised AU Detection

The AU detection task and face synthesis are dual tasks. This section applies the proposed DGAN to semisupervised AU detection. All samples have expression labels and some also have AU labels. The face space is space $\mathcal{X}$, the AU label space is space $\mathcal{Y}$, and the expression is conditional information. Facial feature points represent the face.

A. Problem Statement

Let $\Omega = T \cup U$ denote the training set, where $T = \{(x^i, y^i, E^i_{xy})\}_{i=1}^N$ contains N AU-labeled training samples with feature vectors $x \in \mathbb{R}^d$, AU labels $y \in \{1, 0\}^l$, and expression label $E_{xy} \in \{1, 2, \ldots, P\}$. d represents the dimension of x, l stands for the number of AUs, and P is the number of expressions. $U = \{(x^j, E^j_x)\}_{j=1}^M$ contains M training samples annotated with expression labels only. $X = \{(x^i, E^i_x)\}_{i=1}^N$ stores all feature vectors and their corresponding expression labels in T, and $B = \{(y^i, E^i_y)\}_{i=1}^N$ stores all AU labels and their corresponding expression labels in T. $A = X \cup U$ denotes the subset of $\mathcal{X}$ storing all training feature vectors and their corresponding expression labels. This section's goal is to jointly train the AU classifier $C : \mathbb{R}^d \to \{1, 0\}^l$ and a facial image generator $G : \{1, 0\}^l \to \mathbb{R}^d$ with the training set Ω, thus exploring the connections between the dual tasks to boost the performance of both tasks.

B. Proposed Approach

The framework of the proposed DGAN for semisupervised AU detection is shown as Fig. 1. Since we consider the assistance of expression, the input of discriminator D is a feature—AU–expression tuple, that is, the real (x, y, E_{xy}) from T, and the fake $(x, \hat{y}, E_x)$ and $(\hat{x}, y, E_y)$, generated by C and G, respectively. When generating the face, we introduce the Gaussian noise $z \sim p_z(z)$, so $\hat{x} = G(y, z)$. Then, the adversarial loss of DGAN for AU detection and face synthesis is as follows:

$$\begin{aligned}\mathcal{L}_{\text{adv}} &= \mathbb{E}_{(x,y,E_{xy})\sim T}\big[\log D\big(x, y, E_{xy}\big)\big] \\ &\quad + \alpha\mathbb{E}_{(x,E_x)\sim A}\big[\log(1 - D(x, C(x), E_x))\big] \\ &\quad + (1-\alpha)\mathbb{E}_{(y,E_y)\sim B, z\sim p_z(z)}\big[\log\big(1 - D\big(G(y, z), y, E_y\big)\big)\big].\end{aligned} \tag{4}$$

We set $\alpha = 0.5$ in our experiments to balance the distributions of pseudo-tuples generated from C and G. The classifier C and generator G try to minimize this loss; discriminator D attempts to maximize it. Since the objectives for D, C, and G are different, we define $\mathcal{L}^d_{\text{adv}} = \mathcal{L}_{\text{adv}}$, and $\mathcal{L}^c_{\text{adv}}$ for C and $\mathcal{L}^g_{\text{adv}}$ for G as follows:

$$\begin{aligned}\mathcal{L}^c_{\text{adv}} &= -\mathbb{E}_{(x,E_x)\sim A}\big[\log D(x, C(x), E_x)\big] \\ \mathcal{L}^g_{\text{adv}} &= -\mathbb{E}_{(y,E_y)\sim B, z\sim p_z(z)}\big[\log D\big(G(y, z), y, E_y\big)\big].\end{aligned} \tag{5}$$

For the reconstructed feature and AU labels, we adopt the L1 distance and cross-entropy loss (CE) to calculate the difference between the reconstruction and the original, respectively. So the reconstruction losses for C ($\mathcal{L}^c_{\text{rec}}$) and G ($\mathcal{L}^g_{\text{rec}}$) are as

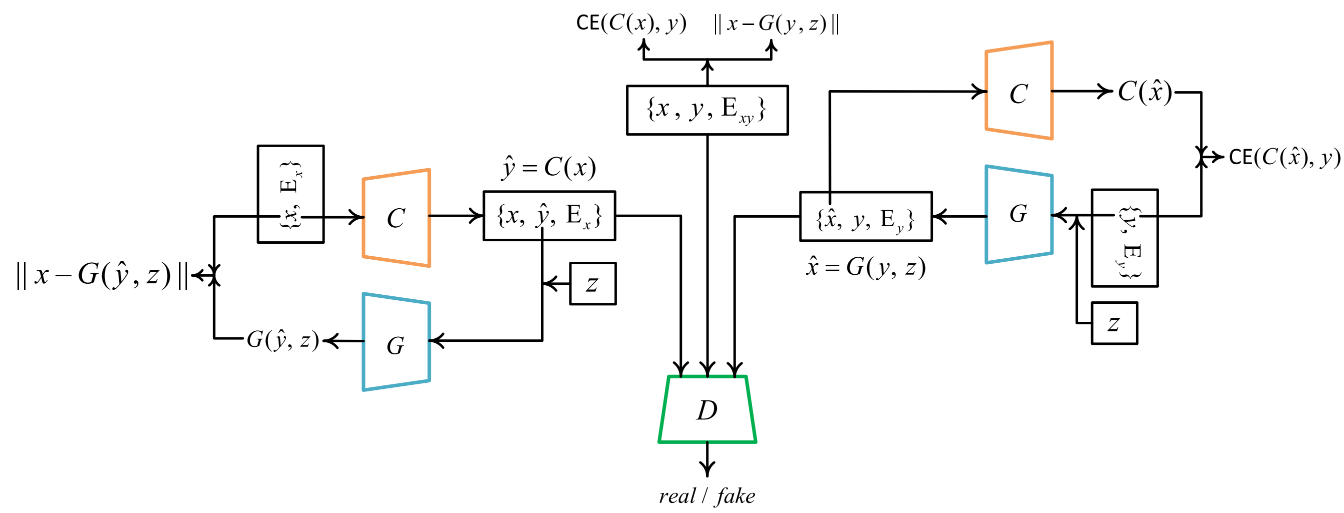


Fig. 1. Framework of the proposed DGAN, which is applied to semisupervised AU detection. Since there are true paired data, we introduce two supervised losses for AU-annotated samples.

follows:

$$\mathcal{L}^c_{\text{rec}} = \mathbb{E}_{(x,E_x)\sim A, z\sim p_z(z)}\big[\|x - G(C(x), z)\|_1\big]$$
$$\mathcal{L}^g_{\text{rec}} = \mathbb{E}_{(y,E_y)\sim B, z\sim p_z(z)}\big[\text{CE}(C(G(y, z), y)\big]. \tag{6}$$

In the semisupervised scenario, there are some AU-labeled samples (training set T), so the standard supervised loss must be included in the full objective for AU-labeled data (x, y, E_{xy}). The supervised losses for C ($\mathcal{L}_{cl}$) and G ($\mathcal{L}_{\text{reg}}$) are defined as

$$\mathcal{L}_{cl} = \mathbb{E}_{(x,y,E_{xy})\sim T}\big[\text{CE}(C(x), y)\big]$$
$$\mathcal{L}_{\text{reg}} = \mathbb{E}_{(x,y,E_{xy})\sim T, z\sim p_z(z)}\big[\|x - G(y, z)\|_1\big]. \tag{7}$$

Finally, the objectives for D, C, and G are to minimize $\mathcal{L}_D$, $\mathcal{L}_C$, and $\mathcal{L}_G$, respectively, and are written as

$$\mathcal{L}_D = -\mathcal{L}^d_{\text{adv}}$$
$$\mathcal{L}_C = \mathcal{L}^c_{\text{adv}} + \lambda_c \mathcal{L}^c_{\text{rec}} + \lambda_{cl}\mathcal{L}_{cl}$$
$$\mathcal{L}_G = \mathcal{L}^g_{\text{adv}} + \lambda_g \mathcal{L}^g_{\text{rec}} + \lambda_{\text{reg}}\mathcal{L}_{\text{reg}} \tag{8}$$

where λ_c and λ_g are weight coefficients of reconstruction loss for C and G, respectively, and λ_{cl} and λ_{reg} are weight coefficients of supervised loss for C and G, respectively. Like the training procedure of Vanilla GAN [10], D, C, and G are updated alternately like so: first, update D while fixing C and G, then update C while fixing D and G, and then update G while fixing D and C. The process is repeated until convergence. Algorithm 1 outlines the detailed training procedure.

All of the networks of discriminator D, classifier C, and generator G are parameterized through a four-layer feedforward network since the dimensions of feature and AU labels are not very high. We use the TensorFlow [31] framework to implement the proposed DGAN. The Adam [32] optimization method is the best choice to update the parameters of GAN. A validation set and grid search strategy are used to determine other hyperparameters, such as weight coefficients λ_c, λ_{cl}, λ_g, and λ_{reg}, training step K, and batch size s, which varies by database.

V. Weakly Supervised AU Detection

This section applies the proposed DGAN to weakly supervised AU detection scenarios in which no true paired data exit. We generate pseudo paired data as the real sample according to the summarized domain knowledge about expressions and AUs.

A. Problem Statement

$A = \{(x^i, E^i_x)\}^N_{i=1}$ denotes the training set annotated with expression labels only. The aim is to jointly train an AU classifier $C : \mathbb{R}^d \rightarrow \{1, 0\}^l$ and a facial image generator $G : \{1, 0\}^l \rightarrow \mathbb{R}^d$ with the training set A only.

B. Proposed Approach

1) Domain Knowledge: In our previous work [28], we detailed the current domain knowledge regarding facial expressions and AUs. Here, we briefly introduce the domain knowledge. The domain knowledge is represented as the conditional probability of any single AU. The likelihood of an AU can be categorized into one of the three kinds according to the conditions.

The first is the likelihood of any one AU, given one expression. From Du *et al.*'s work [6], we can obtain the conditional probability of each AU given one of the six basic expressions. From Prkachin and Solomon's work about Prkachin and Solomon pain intensity (PSPI) [8], [33], we can obtain the domain knowledge regarding the likely presence of six AUs (AU4, AU6, AU7, AU9, AU10, and AU43) given the pain expression.

The second is the conditional probability of an AU given one expression as well as another AU. From EMFACS [7], we obtain many AU combinations that are often noticed during one of the six basic expressions, from which we can summarize the conditional probabilities.

The third is the conditional probability of a single AU when another AU is absent or present. This shows the co-existent and mutually exclusive relations between two AUs. FACS and

Algorithm 1 Training of DGAN in Semisupervised Scenario

```
Require: Training sets $T$, $A$, and $B$; max number of training steps $K$, batch size $s$; weight coefficients $\lambda_c$, $\lambda_{cl}$, $\lambda_g$, and $\lambda_{reg}$.
Ensure: Classifier $C$ and generator $G$
Randomly initialize parameters $\theta_d$, $\theta_c$, and $\theta_g$ of discriminator $D$, classifier $C$, and generator $G$, respectively.
for $k = 1, 2, ..., K$ do
    Sample mini-batch of $s$ samples $\{(x_i^d, y_i^d, E_{xy_i^d})\}_{i=1}^s$ from $T$, sample mini-batch of $s$ samples $\{(x_i^c, E_{x_i^c})\}_{i=1}^s$ from $A$, sample mini-batch of $s$ samples $\{(y_i^g, E_{y_i^g})\}_{i=1}^s$ from $B$, and sample mini-batch of $s$ noise samples $\{z_i\}_{i=1}^s$ from $p_z(z)$.
    Update discriminator $D$ by descending the gradient:
        $\nabla_{\theta_d}\left[-\frac{1}{s}\sum_{i=1}^{s}\Big(\log D(x_i^d, y_i^d, E_{xy_i^d}) + \alpha\log(1 - D(x_i^c, C(x_i^c), E_{x_i^c})) + (1-\alpha)\log(1 - D(G(y_i^g, z_i), y_i^g, E_{y_i^g}))\Big)\right]$
    Sample mini-batch of $s$ samples $\{(x_i^c, E_{x_i^c})\}_{i=1}^s$ from $A$, $s_1 (s_1 \leq s)$ of which are annotated with AU labels, $\{(x_j^c, y_j^c, E_{x_j^c})\}_{j=1}^{s_1}$. Sample mini-batch of $s$ noise samples $\{z_i\}_{i=1}^s$ from $p_z(z)$.
    Update classifier $C$ by descending its gradient:
        $\nabla_{\theta_c}\left[-\frac{1}{s}\sum_{i=1}^{s}\log D(x_i^c, C(x_i^c), E_{x_i^c}) + \frac{\lambda_c}{s}\sum_{i=1}^{s}||x_i^c - G(C(x_i^c), z_i)||_1 + \frac{\lambda_{cl}}{s_1}\sum_{j=1}^{s_1}\mathrm{CE}(C(x_j^c), y_j^c)\right]$
    Sample mini-batch of $s$ samples $\{(x_i^g, y_i^g, E_{y_i^g})\}_{i=1}^s$ from $T$ and sample mini-batch of $s$ noise samples $\{z_i\}_{i=1}^s$ from $p_z(z)$.
    Update generator $G$ by descending its gradient:
        $\nabla_{\theta_g}\left[-\frac{1}{s}\sum_{i=1}^{s}\log D(G(y_i^g, z_i), y_i^g, E_{y_i^g}) + \frac{\lambda_g}{s}\sum_{i=1}^{s}\mathrm{CE}(C(G(y_i^g, z_i)), y_i^g) + \frac{\lambda_{reg}}{s}\sum_{i=1}^{s}||x_i^g - G(y_i^g, z_i)||_1\right]$
end for
```

Li *et al.*'s work [34] pinpoint some AUs that usually or rarely appear together, from which we can summarize the third type of AU conditional probability (see [28] for more details).

2) Pseudo Data Generation: According to Section IV, we can see that in addition to the training set A, which stores features, we also need the training set B to store labels and the paired training set T in order to learn DGAN. In this section, we generate pseudo B and T according to the summarized kinds of AU conditional probability.

From [28], we find that for many AU conditional probabilities, we only know the range. For example, P(AU7 $=$ 1|anger) $\geq$ 0.7, P(AU2 $=$ 1|happiness) $<$ 0.2, P(AU6 $=$ 1|pain) $\geq$ 0.5, P(AU7 $=$ 1|AU9 $=$ 1) $>$ 0.5, P(AU12 $=$ 1|AU15 $=$ 1) $<$ 0.2, and P(AU6 $=$ 1|AU12 $=$ 1,happiness) $>$ 0.5. So before generating A and B, we use uniform random sampling to sample a certain value for these AU conditional probabilities. After that, we adopt the sampling algorithm used by Peng and Wang [28] to create pseudo AU labels for every expression.

For each expression $E^p, p \in \{1, 2, \ldots, P\}$, we sample a pseudo AU label set $\mathrm{Y}^p = \{y_i^p\}_{i=1}^Q$. Q is the sampling number, set to $Q = 5000$ in our experiments. Then, the pseudo training set B can be represented as $B = \bigcup_{p=1}^P \bigcup_{i=1}^Q \{(y_i^p, E^p)\}$. After obtaining B, we generate pseudo T by connecting the samples in A and B using the expression label as the bridge. Specifically, the pseudo T can be represented as $T = \{(x, y, E_{xy}) : (x, E_{xy}) \in A, (y, E_{xy}) \in B\}$.

3) AU Classifier Learning With DGAN: After obtaining pseudo training sets B and T, we can learn DGAN as outlined in Section IV. We do see adversarial loss as (4) and reconstruction loss as (6). However, there is no supervised loss in the full objective (8), since there are no true paired samples in the training set. So the new objectives for D, C, and G are to minimize $\mathcal{L}'_D$, $\mathcal{L}'_C$, and $\mathcal{L}'_G$, respectively, written as

$$\begin{aligned}\mathcal{L}'_D &= -\mathcal{L}^d_{\mathrm{adv}}\\ \mathcal{L}'_C &= \mathcal{L}^c_{\mathrm{adv}} + \lambda_c\mathcal{L}^c_{\mathrm{rec}}\\ \mathcal{L}'_G &= \mathcal{L}^g_{\mathrm{adv}} + \lambda_g\mathcal{L}^g_{\mathrm{rec}}.\end{aligned} \tag{9}$$

The training process is similar to Algorithm 1. The detailed training procedure is described in Algorithm 2.

VI. Experiments

A. Experimental Conditions

Three benchmark databases are used in our experiments: 1) the extended Cohn-Kanade database (CK+) [35]; 2) the MMI database [36]; and 3) the UNBC-McMaster shoulder pain expression archive database (Pain) [37]. Several samples from the three databases are shown in Fig. 2.

The CK+ database consists of 593 facial sequences from the posed expressions of 123 subjects. From 106 of those subjects, 309 sequences annotated with six basic expressions are selected. The apex frame (the last frame) of each of 309 sequences is used. Similar to [28], we consider 12 AUs (AU1, AU2, AU4, AU5, AU6, AU7, AU9, AU12, AU17, AU23, AU24, and AU25) with occurrence frequencies greater than 10%.

The MMI database is also composed of posed expressions and contains 2900 videos from 27 subjects. Among them, apex frames are taken from 171 sequences of 27 subjects.

Algorithm 2 Training of DGAN in Weakly Supervised Scenario

```
Require: Training sets $T$, $A$, and $B$; max number of training steps $K$; batch size $s$; weight coefficients $\alpha$, $\lambda_c$, and $\lambda_g$.
Ensure: Classifier $C$ and generator $G$
Randomly initialize parameters $\theta_d$, $\theta_c$, and $\theta_g$ of discriminator $D$, classifier $C$, and generator $G$, respectively.
for $k = 1, 2, ..., K$ do
    Sample mini-batch of $s$ samples $\{(x_i^d, y_i^d, E_{xy_i^d})\}_{i=1}^s$ from $T$, sample mini-batch of $s$ samples $\{(x_i^c, E_{x_i^c})\}_{i=1}^s$ from $A$, sample mini-batch of $s$ samples $\{(y_i^g, E_{y_i^g})\}_{i=1}^s$ from $B$, and sample mini-batch of $s$ noise samples $\{z_i\}_{i=1}^s$ from $p_z(z)$.
    Update discriminator $D$ using gradient descent:
        $\nabla_{\theta_d}\left[-\frac{1}{s}\sum_{i=1}^{s}\Big(\log D(x_i^d, y_i^d, E_{xy_i^d}) + \alpha\log(1 - D(x_i^c, C(x_i^c), E_{x_i^c})) + (1-\alpha)\log(1 - D(G(y_i^g, z_i), y_i^g, E_{y_i^g}))\Big)\right]$
    Sample mini-batch of $s$ samples $\{(x_i^c, E_{x_i^c})\}_{i=1}^s$ from $A$ and sample mini-batch of $s$ noise samples $\{z_i\}_{i=1}^s$ from $p_z(z)$.
    Update classifier $C$ using gradient descent:
        $\nabla_{\theta_c}\left[-\frac{1}{s}\sum_{i=1}^{s}\log D(x_i^c, C(x_i^c), E_{x_i^c}) + \frac{\lambda_c}{s}\sum_{i=1}^{s}||x_i^c - G(C(x_i^c), z_i)||_1\right]$
    Sample mini-batch of $s$ samples $\{y_i^g, E_{y_i^g})\}_{i=1}^s$ from $B$ and sample mini-batch of $s$ noise samples $\{z_i\}_{i=1}^s$ from $p_z(z)$.
    Update generator $G$ via gradient descent:
        $\nabla_{\theta_g}\left[-\frac{1}{s}\sum_{i=1}^{s}\log D(G(y_i^g, z_i), y_i^g, E_{y_i^g}) + \frac{\lambda_g}{s}\sum_{i=1}^{s}\mathrm{CE}(C(G(y_i^g, z_i)), y_i^g)\right]$
end for
```

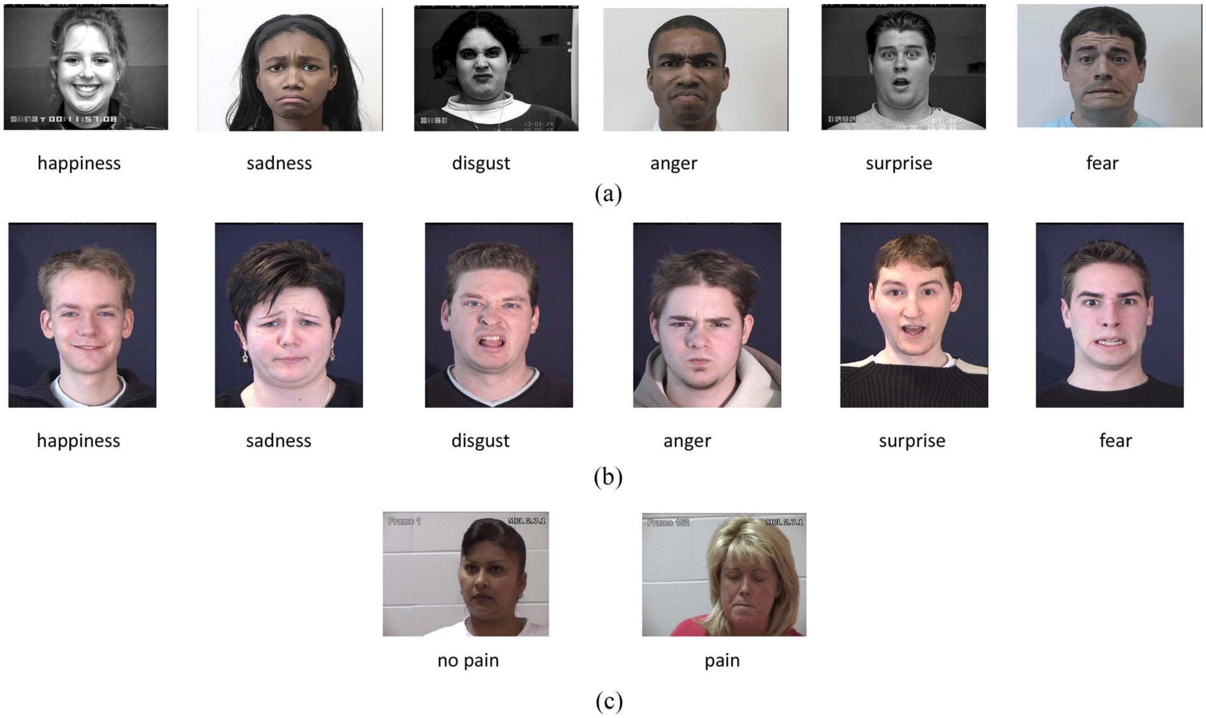


Fig. 2. Several samples from the three databases. (a) Samples from the CK+ database. (b) Samples from the MMI database. (c) Samples from the Pain database.

Frames are annotated with AUs and one of the six expressions. Thirteen AUs (AU1, AU2, AU4, AU5, AU6, AU7, AU9, AU10, AU12, AU17, AU23, AU25, and AU26) with occurrence frequencies greater than 10% are considered.

The Pain database is spontaneous and contains 200 video sequences from 25 patients with shoulder pain as they exhibit "pain" or "no pain" expressions. Each frame is coded with PSPI, used to evaluate pain intensity. Similar to [28], frames with PSPI > 4 are regarded as pain, and frames with PSPI $= 0$ are regarded as no pain. We select all pain and no pain frames (7319 frames in total) from 30 sequences of 17 subjects. Six AUs related to pain expression are considered.

Although the representations of pretrained deep nets may carry more information for facial appearance, such representations could dramatically increase the complexity of the joint probability of facial features, expressions, and AUs compared to that of facial feature points, expressions, and AUs. Therefore, 2-D positions of landmarks are used as the features in this work. They effectively capture face shape and are crucial for facial expression analyses. On the CK+ database, we use 49 feature points and on the Pain database, we use 66 feature points. The feature points for these databases are provided by database constructors. The MMI database does not provide feature points, so we detect 49 feature points through IntraFace [38]. An affine transformation is used to make the centers of the eye fall on the appropriate positions, and Gaussian normalization is performed for each feature dimension. The average F1 score of all AUs is used to evaluate the performance of AU detection. Face synthesis is evaluated by the root mean-square error (RMSE). Higher $F1$ scores and lower RMSE indicate better performance on the dual tasks.

In both semisupervised and weakly supervised scenarios, within-database experiments are conducted using five-fold subject-independent cross-validation. We also conduct cross-database experiments, and each of the experiments is performed five times. In cross-database experiments, we use common AUs for experiments. The common AUs between the MMI dataset and the CK + dataset are AU1, AU2, AU4, AU5, AU6, AU7, AU9, AU12, AU17, AU23, and AU25. The common AUs between the Pain dataset and the CK + dataset are AU4, AU6, AU7, and AU9. The common AUs between the Pain dataset and the MMI dataset are AU4, AU6, AU7, AU9, and AU10. To simulate semisupervised scenarios, AU labels are randomly missed according to certain probabilities: 0.1, 0.2, 0.3, 0.4, and 0.5.

Ablation studies are performed on within-database experiments to demonstrate the impacts of expression and reconstruction loss. The proposed method is compared to a method that does not consider the assistance of expression labels, referred to as DGAN$_{ne}$, which removes the expression label in feature–AU–expression tuple. This comparison is only for semisupervised scenarios since the expression labels are necessary for weakly supervised scenarios. We also compare the results of our proposed method to one that removes the reconstruction loss, referred to as DGAN$_{nr}$, by setting $\lambda_c = \lambda_g = 0$ in both semisupervised and weakly supervised scenarios.

The results of the proposed method are compared to those achieved by state-of-the-art works. For semisupervised scenarios, the compared methods are BGCS, MLML, BN, SHTL, RBM-P, RAN, and Wang *et al.*'s work Rank [29] on within-database experiments; and to SHTL, RBM-P, and RAN on cross-database experiments. For weakly supervised scenarios, the compared methods are HTL, RBM-P, RAN, Rank, and SVM (only on cross-database experiments). We copy the results of RBM-P, RAN, and Rank from the original papers, since the experimental conditions of these three methods are as same as ours, except that Wang *et al.* [29] used all frames with PSPI>0 and considered ten AUs (AU4, AU6, AU7, AU9, AU10, AU12, AU20, AU25, AU26, and AU43) and 13 AUs (AU1, AU2, AU4, AU5, AU6, AU7, AU9, AU12, AU17, AU23, AU24, AU25, and AU27) on the Pain and CK+ database respectively. The comparison to Rank on the Pain and CK+ database is only for reference. Wang *et al.* [29] only conducted semisupervised experiments with missing rates set to 0.5 and only conducted cross-experiments between the CK+ and MMI databases. For completeness of comparison, we conduct the semisupervised experiment as the author's condition. The results of common AUs are outside parentheses, and the results of all AUs are in parentheses. Since the experimental conditions of these three works are different from ours, the results of BGCS, MLML, and HTL (SHTL) are copied from [27], as the authors reconducted those experiments. We do not compare the proposed method to LP-SM or Wu *et al.*'s work [24], since Zhang *et al.* [4] considered different AUs on the MMI and CK+ databases and different frames on the Pain database, and Wu *et al.* [24] did not perform experiments on any of those databases.

For the face synthesis task, the proposed method is compared to the discriminative RBM (DRBM) [39] in which feature and AU label vectors constitute the visible layer. A Gibbs sampling method is used to infer facial features from input AU labels.

B. Experimental Results and Analyses

1) Experimental Results of Semisupervised AU Detection: Table I shows the results of semisupervised AU detection on within-database experiments with five missing rates on the three databases. From this table, we can observe the following.

First, when comparing methods using the assistance of expressions to methods ignoring expressions, the methods considering expressions perform better overall. For example, DGAN performs better than DGAN$_{ne}$ in all cases, and RAN and RBM-P perform better than MLML and BGCS in all cases. This demonstrates that expression is definitely helpful for AU detection due to the strong dependencies between expressions and AUs. When AUs are missing, expression labels can provide weak supervisory information.

Second, compared to the other two methods that do not consider the help of expressions, DGAN$_{ne}$ achieves the best performance in every scenario, proving the supremacy of the proposed method. For example, when the missing rate is 0.1 on the CK+ database, DGAN$_{ne}$ achieves 12.99% and 32.33% improvements over BGCS and MLML, respectively. Although expression labels are not present, DGAN$_{ne}$ captures the joint distribution of features and AUs to leverage the weak supervisory information inherent in the dependencies between features and AUs in addition to the dependencies among AUs for samples without AU labels. BGCS and MLML do not capture this distribution. DGAN$_{ne}$ also considers and trains the AU detection task and the face synthesis task simultaneously, while BGCS and MLML only consider the AU detection task.

Third, compared to BN, SHTL, RBM-P, RAN, and Rank, the proposed DGAN performs best in most cases. For example, when the missing rate is 0.1 on the MMI database, DGAN achieves improvements of 17.05%, 5.43%, 4.43%, and 4.03% over BN, SHTL, RBM-P, and RAN, respectively, demonstrating the effectiveness of DGAN. BN can only explore pairwise

TABLE I
RESULTS OF WITHIN-DATABASE EXPERIMENTS OF SEMISUPERVISED AU DETECTION WITH FIVE MISSING RATES ON THE THREE DATABASES (BOLD NUMBERS INDICATE THE BEST PERFORMANCE)

	methods	0.1	0.2	0.3	0.4	0.5
CK+	MLML	.6152	.6115	.6052	.6278	.6515
	BGCS	.7205	.7178	.7117	.7032	.6957
	DGAN$_{ne}$	.8141	.8018	.7923	.7844	.7824
	BN	.7738	.7835	.7837	.7817	.7808
	SHTL	.5997	.5958	.5931	.5957	.5940
	RBM-P	.8186	.8148	.7948	**.8053**	.7868
	RAN	.8114	.8059	.7986	.7993	.7916
	Rank	.7803 (.7921)	.7637 (.7738)	.7653 (.7789)	.7719 (.7848)	.7689 (.7810)
	DGAN	**.8287**	**.8184**	**.8057**	.8015	**.7917**
	DGAN$_{nr}$	.8131	.8062	.8032	.7929	.7838
MMI	MLML	.5063	.4806	.4793	.4651	.4323
	BGCS	.4667	.4559	.4491	.4350	.4466
	DGAN$_{ne}$	.5672	.5583	.5489	.5349	.5218
	BN	.4897	.4792	.4725	.4659	.4378
	SHTL	.5437	.5331	.5332	.5317	.5301
	RBM-P	.5489	.5348	.5355	.5344	.5312
	RAN	.5510	.5392	.5405	.5328	.5299
	Rank	**.5848**	**.5829**	**.5802**	**.5750**	**.5720**
	DGAN	.5732	.5609	.5550	.5513	.5484
	DGAN$_{nr}$	.5634	.5519	.5467	.5368	.5264
Pain	MLML	.2101	.2222	.1786	.1566	.1461
	BGCS	.4700	.4621	.4787	.4647	.4497
	DGAN$_{ne}$	.5145	.5002	.4970	.5026	.4939
	BN	.2654	.3027	.2505	.2445	.1831
	SHTL	.3266	.3184	.3091	.3005	.2929
	RBM-P	.5288	.5155	.5101	.5087	.5020
	RAN	.5072	.5034	.4955	.4854	.4724
	Rank	.4384 (.3815)	.4175 (.3710)	.4258 (.3730)	.4235 (.3736)	.3968 (.3550)
	DGAN	**.5368**	**.5279**	**.5187**	**.5161**	**.5195**
	DGAN$_{nr}$	.4992	.4965	.4784	.4535	.4355

dependencies among AUs, while DGAN can explore global relations among all AUs since DGAN captures their joint distribution. For SHTL, any error of the expression classifier propagates to the AU classifier, since they are trained separately rather than simultaneously, as DGAN does. Although both RBM-P and RAN explore global relations among AUs, they ignore relations between features and AUs. DGAN captures the joint distribution of features, AUs, and expression. Rank uses the rank relations but not the specific value of the AU probabilities, and ignores AU condition probability given another AU. Furthermore, all four methods only handle the AU detection task, ignoring the helpful intrinsic connections between AU detection and face synthesis. We optimize the dual tasks, thus achieving better performance.

Finally, comparing DGAN$_{ne}$ to methods considering expressions, we find that DGAN$_{ne}$ performs better than some of them in some cases. For example, DGAN$_{ne}$ performs better than BN, SHTL, and Rank on the CK+ database; better than BN, SHTL, RBM-P, and RAN on the MMI database; and better than BN, SHTL, RAN, and Rank on the Pain database. Although DGAN$_{ne}$ does not use the assistance of expression, DGAN$_{ne}$ successfully exploits the duality between the tasks to improve the AU classifier.

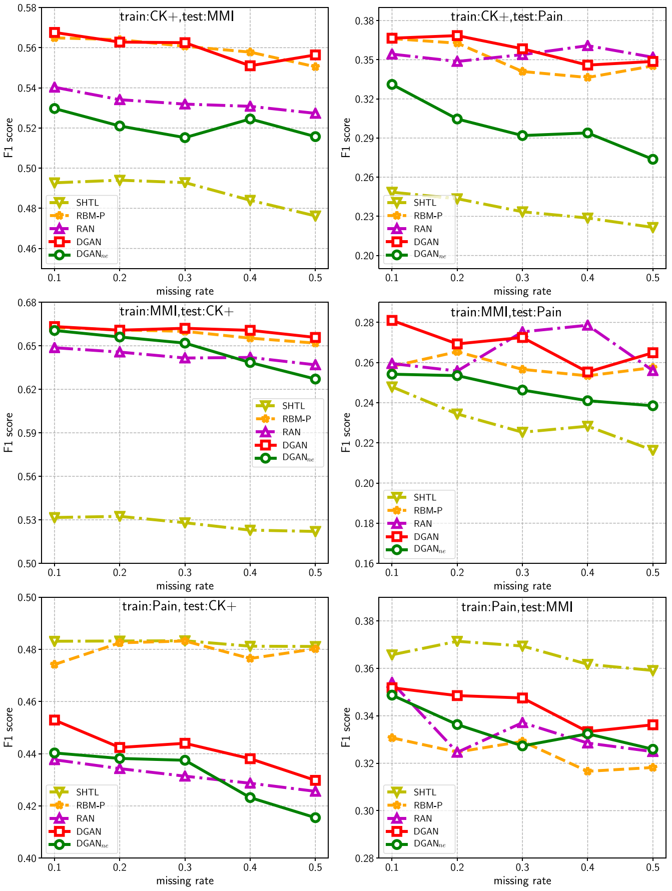


Fig. 3. Cross-database experimental results of semisupervised AU detection.

In order to visually see the specific AU value of the DGAN variant, we also list AU-specific semisupervised results of variants of DGAN with 0.5 missing rate as Table II. From Table II, we can see that the F1 score of most AUs in DGAN is better than DGAN$_{ne}$ and DGAN$_{nr}$. This is because DGAN adds expression labels as auxiliary information and uses dual structures.

Fig. 3 shows the results of cross-database experiments on the semisupervised AU detection task. DGAN performs best in most cases when training is performed on the CK+ or MMI databases (the first two rows of Fig. 3). The experiments testing on the Pain database are particularly difficult for the AU detection task. The CK+ and MMI databases use the six basic expressions, so pain is not included. Thus, there are large biases between the training and testing sets. The superior performances of DGAN demonstrate the better generalization ability of DGAN. DGAN$_{ne}$ also performs better than SHTL in all cases. Although DGAN$_{ne}$ does not use expression labels, it makes use of the duality between the tasks. However, DGAN performs poorly when training on the

TABLE II
WITHIN-DATABASE EXPERIMENTAL RESULTS OF SEMISUPERVISED FOR DGAN VARIANTS WITH 0.5 MISSING RATE ON THE THREE DATABASES

AU	CK+			MMI			Pain		
	DGAN	DGAN$_{ne}$	DGAN$_{nr}$	DGAN	DGAN$_{ne}$	DGAN$_{nr}$	DGAN	DGAN$_{ne}$	DGAN$_{nr}$
1	**.8948**	.8708	.8672	.6000	.5395	**.6536**	-	-	-
2	**.9020**	.8709	.8815	**.6943**	.6841	.6899	-	-	-
4	**.8327**	.7363	.7668	.5334	.5605	**.6172**	.4989	**.5841**	.4323
5	.8105	**.8313**	.8179	.6702	**.7160**	.6575	-	-	-
6	.7344	.7520	**.7603**	.4644	.4247	**.4712**	**.7663**	.7433	.6417
7	.6235	**.6641**	.5999	**.4610**	.3985	.4170	**.5484**	.4909	.4664
9	.8287	**.8594**	.8244	**.4901**	.4873	.3502	.4118	.3625	**.4223**
10	-	-	-	**.4650**	.3255	.3353	**.1292**	.1127	.0610
12	.8873	**.8954**	.8484	.6617	.6826	**.7341**	-	-	-
17	**.8631**	.8183	.8230	.4177	.3942	**.4799**	-	-	-
23	.6036	.6016	**.7079**	**.1723**	.0876	.1470	-	-	-
24	.5556	.5521	**.5646**	-	-	-	-	-	-
25	**.9652**	.9366	.9436	**.8727**	.8396	.7863	-	-	-
26	-	-	-	.6257	**.6437**	.5040	-	-	-
43	-	-	-	-	-	-	**.7625**	.6702	.5894
Avg.	**.7917**	.7824	.7838	**.5484**	.5218	.5264	**.5195**	.4939	.4355

TABLE III
WITHIN-DATABASE EXPERIMENTAL RESULTS OF WEAKLY SUPERVISED AU DETECTION ON THE THREE DATABASES

AU	CK+						MMI						Pain					
	HTL	RBM-P	RAN	Rank	DGAN	DGAN$_{nr}$	HTL	RBM-P	RAN	Rank	DGAN	DGAN$_{nr}$	HTL	RBM-P	RAN	Rank	DGAN	DGAN$_{nr}$
1	.6071	.9129	**.9365**	.8450	.9120	.8908	.3857	.6947	.6754	.6850	.6622	**.7069**	-	-	-	-	-	-
2	.6776	.8928	.8987	.8560	**.9011**	.8950	.6000	.6994	.5967	**.7270**	.7007	.7151	-	-	-	-	-	-
4	.4593	.7008	**.7414**	.6760	.7372	.7355	.3407	.6052	.6097	.6440	.6540	**.6781**	.2732	.4261	**.4416**	.1610	.4304	.4413
5	.7522	.8189	.7989	.7470	**.8334**	.8255	.5882	.6720	**.7092**	.6940	.6714	.6699	-	-	-	-	-	-
6	.5079	.1116	.5337	.6140	.5979	**.6242**	**.3922**	.2766	.3430	.3810	.3892	.3539	.3532	.4552	.5019	**.6810**	.5719	.5385
7	.3750	.4091	.4465	**.6400**	.5682	.6085	.3220	.4713	.4258	**.5000**	.4309	.4737	.1955	**.4221**	.3632	.3980	.4059	.3555
9	.3679	**.9036**	.8861	.7470	.8552	.8122	.3495	.3075	.4088	**.4880**	.4879	.4728	.1854	.3352	.3068	.0830	**.3736**	.3047
10	-	-	-	-	-	-	.4118	.2850	.2355	.3640	**.4792**	.3974	.1563	.1675	.0689	.0130	**.2523**	.2051
12	.4444	.8412	.8292	**.9100**	.8785	.8740	.5439	.5905	.6881	.7360	**.7376**	.7270	-	-	-	.6720	-	-
17	.3731	.8560	.6789	**.8580**	.8465	.8460	.2167	.5056	**.5110**	.4840	.4560	.4980	-	-	-	-	-	-
23	.2739	.4157	.4122	**.7040**	.5080	.3726	.2187	**.2493**	.2239	.1090	.1196	.0773	-	-	-	-	-	-
24	.2412	**.6448**	.4879	.4710	.5516	.4653	-	-	-	-	-	-	-	-	-	-	-	-
25	.6852	**.9529**	.9369	.9450	.9517	.9489	.6631	.7521	.7025	.7240	.7611	**.7630**	-	-	-	.2020	-	-
26	-	-	-	-	-	-	.5714	.6004	**.6379**	.4670	.5534	.5508	-	-	-	.2140	-	-
43	-	-	-	-	-	-	-	-	-	-	-	-	-	.3006	**.5754**	.3900	.5504	.5104
Avg.	.4698	.7050	.7156	.7510	**.7618**	.7413	.4310	.5161	.5206	.5390	**.5464**	.5449	.2352	.3510	.3763	.3130	**.4308**	.3926

Pain database and testing on the CK+ and MMI datasets (the last row of Fig. 3). SHTL performs best in these two scenarios since SHTL trains with the Pain database as well as an extra-large facial image database annotated with six basic expressions, reducing database biases. Furthermore, only six AUs are considered on the Pain database. This might not carry enough information to generate faces, so the assistance of face synthesis is less significant.

2) Experimental Results of Weakly Supervised AU Detection: Table III shows the results of weakly supervised within-database experiments. DGAN achieves the best average F1 scores on all datasets. For the CK+ database, the average F1 scores of DGAN are 62.15%, 8.06%, 6.46%, and 1.44% higher than those of HTL, RBM-P, RAN, and Rank, respectively.

Both HTL and Rank only consider the probability of a single AU given one expression. The performance of HTL is constrained by the accuracy of the expression classifier. Rank only uses the rank relations between the AU probabilities. RBM-P needs the pretrained RBM model. None of the compared methods consider the face synthesis task. However, DGAN considers the dual task of AU detection, takes advantage of the probabilistic duality and reconstruction loss, makes use of comprehensive AU condition probabilities, and uses adversarial learning to explore the joint distribution and avoid the complex distribution estimation process, thus achieving the best performance.

Tables IV–VI show the results of cross-database experiments in weakly supervised scenarios. Table IV shows generally improved results compared to Tables V and VI, since the CK+ and MMI databases use the six basic expressions, but the Pain dataset uses the pain expression. Although SVM is fully supervised, the proposed DGAN performs better. For example, DGAN achieves 34.5% improvement over

TABLE IV
CROSS-DATABASE EXPERIMENTAL RESULTS OF WEAKLY SUPERVISED AU DETECTION BETWEEN THE CK+ AND MMI DATABASES

AU	CK+ → MMI						MMI → CK+					
	SVM	HTL	RBM-P	RAN	Rank	DGAN	SVM	HTL	RBM-P	RAN	Rank	DGAN
1	.6463	.3857	.6800	.5455	.6670	**.7220**	.6215	.6071	**.8377**	.7459	.7000	.7637
2	.5546	.6000	.6174	.6748	.6490	**.7560**	.6638	.6776	.7881	.7845	**.8020**	.7811
4	.3857	.3407	.6014	**.6395**	.5780	.4916	.5799	.4593	.5683	.5959	**.6500**	.6161
5	.4355	.5882	.5942	.6316	.6290	**.6990**	.7393	**.7522**	.7321	.6279	.7070	.7172
6	.2811	.3922	.2830	.3485	.3660	**.4169**	.5244	.5079	.5852	.5240	.5060	**.6016**
7	.2373	.3220	**.5455**	.4430	.4710	.4041	.4397	.3750	.5103	.4504	.5520	**.6210**
9	**.4127**	.3495	.3721	.3288	.2710	.3877	.3907	.3679	**.5463**	.4298	.4550	.5252
12	.4156	.5439	.6286	.5354	.5980	**.6542**	.4434	.4444	.4661	**.7634**	.7450	.7056
17	.1765	.2167	.3729	.4478	.3910	**.4706**	.3636	.3731	.5926	.6543	.6620	**.6889**
23	.0702	.2178	.0943	**.2203**	.0680	.1370	.2206	.2739	.3054	.2037	**.4190**	.2430
25	.7984	.6631	.6368	.7577	.7590	**.7993**	.6870	.6852	.6787	.8421	**.8700**	.8380
Avg.	.4014	.4200	.4943	.5066	.4950	**.5399**	.5453	.5021	.6011	.6020	.6430	**.6456**

CK+ → MMI represents training on the CK+ database and testing on the MMI database.

TABLE V
CROSS-DATABASE EXPERIMENTAL RESULTS OF WEAKLY SUPERVISED AU DETECTION BETWEEN THE CK+ AND PAIN DATABASES

AU		4	6	7	9	Avg.
CK+ ↓ Pain	SVM	.1900	.2919	.2405	**.3453**	.2699
	HTL	**.2732**	.3532	.1995	.1854	.2518
	RBM-P	.2234	.4114	.3453	.3017	**.3204**
	RAN	.2572	.4526	**.4311**	.0381	.2948
	DGAN	.1088	**.5431**	.1816	.1882	.2554
Pain ↓ CK+	SVM	.2412	.3438	.1522	.2899	.2567
	HTL	**.4593**	.5079	.3750	.3679	**.4275**
	RBM-P	.2312	.4148	.1849	.3366	.3366
	RAN	.4470	.4066	**.4036**	.1714	.3572
	DGAN	.2840	**.5104**	.2903	**.3774**	.3655

TABLE VI
CROSS-DATABASE EXPERIMENTAL RESULTS OF WEAKLY SUPERVISED AU DETECTION BETWEEN THE MMI AND PAIN DATABASES

AU		4	6	7	9	10	Avg.
MMI ↓ Pain	SVM	.0987	.2627	.1041	**.5342**	.0534	.2106
	HTL	**.2732**	.3532	.1955	.1854	.1563	.2325
	RBM-P	.0224	**.3926**	**.2536**	.3657	.1949	.2458
	RAN	.1647	.2899	.0123	.4543	.3212	.2482
	DGAN	.0731	.3276	.0524	.4356	**.3617**	**.2501**
Pain ↓ MMI	SVM	.2804	.2432	.2198	**.4308**	.1111	.2571
	HTL	**.3407**	**.3922**	**.3220**	.3495	.4118	**.3632**
	RBM-P	.2812	.0400	.2078	.4262	.4000	.2712
	RAN	.2927	.2727	.2833	.3800	.2376	.2933
	DGAN	.2608	.3455	.2836	.3264	**.4152**	.3258

SVM when training is performed on the CK+ database and testing is performed on the MMI database. SVM learns in a data-driven manner, so it is less generalizable. DGAN uses domain knowledge, which is not dependent on the database.

Compared to RBM-P, RAN, and Rank, DGAN achieves superior performance in most cases, as it has greater generalization ability. For example, when the CK+ dataset is used for training and the MMI dataset is used for testing, DGAN achieves 9.23%, 6.57%, and 9.07% improvement over RBM-P, RAN, and Rank, respectively.

Compared to HTL, DGAN performs better in most cases, apart from two scenarios that train on the Pain database. This is similar to the cross-database experiments in semisupervised scenarios. HTL uses the facial images of the pain expression as well as the six basic expressions.

3) Evaluation of Reconstruction Loss: To determine the relevance of the reconstruction loss, the ablation study is conducted by removing the reconstruction loss in the full objective (DGAN_{nr}) in both semisupervised and weakly supervised scenarios, and then comparing it to DGAN on the three databases. Tables I and III show the results of within-database experiments of DGAN_{nr} in semisupervised and weakly supervised scenarios, respectively.

DGAN_{nr} performs worse than DGAN in all cases, demonstrating the contribution of the reconstruction loss. The reconstruction loss reflects the constraint of the dual task to the primal task. When reconstruction loss is removed, AU detection performance typically decreases.

4) Comparisons to Fully Supervised Methods: The results of performances of the proposed method in semisupervised scenarios (with 0.5 missing rate) and weakly supervised scenarios are compared to fully supervised methods and displayed in Table VII. On the CK+ and Pain databases, the proposed method is compared to MC-LVM [40] and HRBM [41]. For the MMI dataset, the results achieved using the proposed method are compared to the results of SVM-HMM [42] and FFD [43]. The results of HRBM are from [40]. These comparisons are only for reference since their experimental conditions differ from ours. Please note that the compared supervised

TABLE VII
COMPARISON TO THE FULLY SUPERVISED METHODS

	CK+	MMI	Pain
MC-LVM	.7707	-	**.6345**
SVM-HMM	-	**.6712**	-
HRBM	.7147	-	.5942
FFD	-	.6652	-
DGAN (semi-supervised)	**.7917**	.5484	.5195
DGAN (weakly supervised)	.7618	.5464	.4308
DGAN_{nf}	.7761	.5389	.4745

TABLE VIII
RMSE OF DRBM AND THE PROPOSED METHODS FOR FACE SYNTHESIS

	CK+	MMI	Pain
DRBM	1.3866	1.8061	3.0192
DGAN (semi-supervised)	**0.9687**	**0.9866**	**2.5238**
DGAN (weakly supervised)	1.3894	1.4119	2.8680

approaches maybe not state of the art. Still, the comparison demonstrates the effectiveness of the proposed method since it achieves comparable performance with several supervised approaches.

Table VII shows that on the MMI and Pain databases, DGAN performs worse than fully supervised methods. This is expected since DGAN uses the training set. Only half of the samples in this set are annotated with AUs in semisupervised scenarios, and none are labeled in weakly supervised scenarios. The fully supervised methods use complete supervisory information. However, on the CK+ database, DGAN performs better than other methods when half of the samples have AU labels. Surprisingly, DGAN also performs better than HRBM even if samples lack AU labels. This demonstrates the ability of the suggested method to consider the assistance of expression labels as well as the face synthesis task.

To illustrate the effect of the proposed semisupervised DGAN comprehensively, we also compare the results of DGAN with missing rate 0.5 to one whose training set contains only the labeled part of the training dataset of DGAN, referred to as DGAN_{nf}. DGAN_{nf} performs worse than DGAN in all cases, demonstrating the contribution of the unlabeled data.

5) Experimental Results and Analyses of Face Synthesis: This section analyzes the performance of the face generator G. Table VIII shows the results of the RMSE of the proposed DGAN (under semisupervised conditions with a missing rate of 0.2, and weakly supervised scenarios) and the compared method, DRBM.

From Table VIII, we can see the following. First, performances on the Pain database are worse than those on the other two databases. This may be because we consider fewer AUs but more feature points on the Pain database. Six AUs may be insufficient to generate 66 feature points. Second, compared to DRBM, DGAN performs better in semisupervised scenarios on the three databases, despite the fact that DRBM utilizes fully AU-labeled samples. This shows that DGAN is more suitable for face synthesis tasks. Third, the performances of DGAN in semisupervised scenarios are better than those in weakly supervised scenarios. This is reasonable since additional AU-labeled samples could improve AU detection performance, which in turn, benefits the dual task.

VII. CONCLUSION

This work proposed DGAN, a novel dual learning method that explores probabilistic duality through a GAN while leveraging the reconstruction loss. DGAN considered the joint distribution of features, AUs, and expressions, simultaneously capturing the dependencies between expressions and AUs, among AUs, and between features and AUs. The proposed DGAN was applied to weakly supervised and semisupervised AU detection. In the latter scenario, we also minimized the supervised loss for AU-labeled samples. In weakly supervised scenarios, we sampled pseudo paired data according to the summarized domain regarding AUs and expressions. The proposed method achieved better results than the foremost works in both semisupervised and weakly supervised scenarios.

REFERENCES

[1] P. Ekman, W. V. Friesen, and P. Ellsworth, *Emotion in the Human Face: Guidelines for Research and Anintegration of Findings*. New York, NY, USA: Pergamon Press, 1972.

[2] E. Friesen and P. Ekman, *Facial Action Coding System: A Technique for the Measurement of Facial Movement*. New York, NY, USA: Consult. Psychol. Press, 1978.

[3] Y. Tong, W. Liao, and Q. Ji, "Facial action unit recognition by exploiting their dynamic and semantic relationships," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 10, pp. 1683–1699, Oct. 2007.

[4] Y. Zhang, W. Dong, B.-G. Hu, and Q. Ji, "Classifier learning with prior probabilities for facial action unit recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 5108–5116.

[5] X. Niu, H. Han, S. Yang, Y. Huang, and S. Shan, "Local relationship learning with person-specific shape regularization for facial action unit detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 11917–11926.

[6] S. Du, Y. Tao, and A. M. Martinez, "Compound facial expressions of emotion," *Proc. Nat. Acad. Sci. USA*, vol. 111, no. 15, pp. E1454–E1462, 2014.

[7] W. V. Friesen and P. Ekman, *Emfacs-7: Emotional Facial Action Coding System*, Univ. California at San Francisco, San Francisco, CA, USA, 1983.

[8] K. M. Prkachin and P. E. Solomon, "The structure, reliability and validity of pain expression: Evidence from patients with shoulder pain," *Pain*, vol. 139, no. 2, pp. 267–274, 2008.

[9] Y. Xia, T. Qin, W. Chen, J. Bian, N. Yu, and T.-Y. Liu, "Dual supervised learning," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 3789–3798.

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

[11] X. Xu, H. Lu, J. Song, Y. Yang, H. T. Shen, and X. Li, "Ternary adversarial networks with self-supervision for zero-shot cross-modal retrieval," *IEEE Trans. Cybern.*, vol. 50, no. 6, pp. 2400–2413, Jun. 2020.

[12] Y. Yang *et al.*, "Video captioning by adversarial LSTM," *IEEE Trans. Image Process.*, early access, Jul. 12, 2018, doi: 10.1109/TIP.2018.2855422.

[13] G. Peng and S. Wang, "Dual semi-supervised learning for facial action unit recognition," in *Proc. 33rd AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 8827–8834.

[14] D. He *et al.*, "Dual learning for machine translation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 820–828.

[15] Z. Yi, H. Zhang, P. Tan, and M. Gong, "DualGAN: Unsupervised dual learning for image-to-image translation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2849–2857.

[16] Y. Xia, X. Tan, F. Tian, T. Qin, N. Yu, and T.-Y. Liu, "Model-level dual learning," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 5383–5392.

[17] M. Pantic and L. J. M. Rothkrantz, "Automatic analysis of facial expressions: The state of the art," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 12, pp. 1424–1445, Dec. 2000.

[18] E. Sariyanidi, H. Gunes, and A. Cavallaro, "Automatic analysis of facial affect: A survey of registration, representation, and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 6, pp. 1113–1133, Jun. 2015.

[19] B. Martinez, M. F. Valstar, B. Jiang, and M. Pantic, "Automatic analysis of facial actions: A survey," *IEEE Trans. Affect. Comput.*, vol. 10, no. 3, pp. 325–347, Jul.–Sep. 2019.

[20] X. Niu, H. Han, S. Shan, and X. Chen, "Multi-label co-regularization for semi-supervised facial action unit recognition," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 907–917.

[21] Y. Song, D. McDuff, D. Vasisht, and A. Kapoor, "Exploiting sparsity and co-occurrence structure for action unit recognition," in *Proc. 11th IEEE Int. Conf. Workshops Autom. Face Gesture Recognit.*, vol. 1, 2015, pp. 1–8.

[22] B. Wu, S. Lyu, B.-G. Hu, and Q. Ji, "Multi-label learning with missing labels for image annotation and facial action unit recognition," *Pattern Recognit.*, vol. 48, no. 7, pp. 2279–2289, 2015.

[23] Y. Li, B. Wu, B. Ghanem, Y. Zhao, H. Yao, and Q. Ji, "Facial action unit recognition under incomplete data based on multi-label learning with missing labels," *Pattern Recognit.*, vol. 60, pp. 890–900, Dec. 2016.

[24] S. Wu, S. Wang, B. Pan, and Q. Ji, "Deep facial action unit recognition from partially labeled data," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 3971–3979.

[25] S. Wang, Q. Gan, and Q. Ji, "Expression-assisted facial action unit recognition under incomplete AU annotation," *Pattern Recognit.*, vol. 61, pp. 78–91, Jan. 2017.

[26] A. Ruiz, J. Van de Weijer, and X. Binefa, "From emotions to action units with hidden and semi-hidden-task learning," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3703–3711.

[27] S. Wang, G. Peng, and Q. Ji, "Exploring domain knowledge for facial expression-assisted action unit activation recognition," *IEEE Trans. Affect. Comput.*, early access, Apr. 2, 2018, doi: 10.1109/TAFFC.2018.2822303.

[28] G. Peng and S. Wang, "Weakly supervised facial action unit recognition through adversarial training," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 2188–2196.

[29] S. Wang, G. Peng, S. Chen, and Q. Ji, "Weakly supervised facial action unit recognition with domain knowledge," *IEEE Trans. Cybern.*, vol. 48, no. 11, pp. 3265–3276, Nov. 2018.

[30] 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.*, 2017, pp. 2242–2251.

[31] M. Abadi *et al.*, "TensorFlow: A system for large-scale machine learning." in *Proc. OSDI*, vol. 16, 2016, pp. 265–283.

[32] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Represent.*, 2015, pp. 118–126.

[33] K. M. Prkachin, "The consistency of facial expressions of pain: a comparison across modalities," *Pain*, vol. 51, no. 3, pp. 297–306, 1992.

[34] Y. Li, J. Chen, Y. Zhao, and Q. Ji, "Data-free prior model for facial action unit recognition," *IEEE Trans. Affect. Comput.*, vol. 4, no. 2, pp. 127–141, Mar. 2013.

[35] 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. Workshops*, 2010, pp. 94–101.

[36] M. Pantic, M. Valstar, R. Rademaker, and L. Maat, "Web-based database for facial expression analysis," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2005, p. 5.

[37] P. Lucey, J. F. Cohn, K. M. Prkachin, P. E. Solomon, and I. Matthews, "Painful data: The UNBC-mcmaster shoulder pain expression archive database," in *Proc. 9th IEEE Int. Conf. Workshops Autom. Face Gesture Recognit.*, 2011, pp. 57–64.

[38] F. De la Torre, W.-S. Chu, X. Xiong, F. Vicente, X. Ding, and J. F. Cohn, "Intraface," in *Proc. 11th IEEE Int. Conf. Workshops Autom. Face Gesture Recognit.*, 2015, pp. 1–8.

[39] H. Larochelle and Y. Bengio, "Classification using discriminative restricted Boltzmann machines," in *Proc. ACM 25th Int. Conf. Mach. Learn.*, 2008, pp. 536–543.

[40] S. Eleftheriadis, O. Rudovic, and M. Pantic, "Multi-conditional latent variable model for joint facial action unit detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3792–3800.

[41] Z. Wang, Y. Li, S. Wang, and Q. Ji, "Capturing global semantic relationships for facial action unit recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 3304–3311.

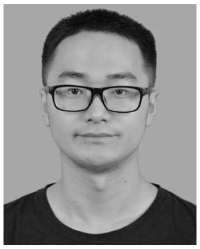
[42] M. F. Valstar and M. Pantic, "Fully automatic recognition of the temporal phases of facial actions," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 1, pp. 28–43, Feb. 2012.

[43] S. Koelstra, M. Pantic, and I. Patras, "A dynamic texture-based approach to recognition of facial actions and their temporal models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 11, pp. 1940–1954, Mar. 2010.

Shangfei Wang (Senior Member, IEEE) received the B.S. degree in electronic engineering from Anhui University, Hefei, China, in 1996, and the M.S. degree in circuits and systems and the Ph.D. degree in signal and information processing from the University of Science and Technology of China (USTC), Hefei, in 1999 and 2002, respectively.

From 2004 to 2005, she was a Postdoctoral Research Fellow with Kyushu University, Fukuoka, Japan. From 2011 to 2012, she was a Visiting Scholar with Rensselaer Polytechnic Institute, Troy, NY, USA. She is currently a Professor with the School of Computer Science and Technology and the School of Data Science, USTC. She has authored or coauthored over 90 publications. Her research interests cover affective computing and probabilistic graphical models.

Prof. Wang is a member of ACM.



Heyan Ding received the B.S. degree in mathematics science from the University of Science and Technology of China, Hefei, China, in 2019, where he is currently pursuing the M.S. degree in data science.

His research interest is affective computing.

Guozhu Peng received the B.S. degree in mathematics from the South China University of Technology, Guangzhou, China, in 2016, and the M.S. degree in computer science from the University of Science and Technology of China, Hefei, China, in 2019.

His research interest is affective computing.