Attack Can Benefit: An Adversarial Approach to Recognizing Facial Expressions under Noisy Annotations

Jiawen Zheng[1*], Bo Li[2*‡], Shengchuan Zhang[1‡], Shuang Wu[2], Liujuan Cao[1], Shouhong Ding[2]

[1]Xiamen University
[2]Youtu Lab, Tencent
{jiawenzheng, caoliujuan,zsc_2016}@xmu.edu.cn, {libraboli, calvinwu, ericshding}@tencent.com

Abstract

The real-world Facial Expression Recognition (FER) datasets usually exhibit complex scenarios with coupled noise annotations and imbalanced class distribution, which undoubtedly impede the development of FER methods. To address the aforementioned issues, in this paper, we propose a novel and flexible method to spot noisy labels by leveraging adversarial attack, termed Geometry Aware Adversarial Vulnerability Estimation (GAAVE). Different from existing state-of-the-art methods of noisy label learning (NLL), our method has no reliance on additional information and is thus easy to generalize to the large-scale real-world FER datasets. Besides, the combination of Dataset Splitting module and Subset Refactoring module mitigates the impact of class imbalance, and the Self-Annotator module facilitates the sufficient use of all training data. Extensive experiments on RAF-DB, FERPlus and AffectNet datasets validate the effectiveness of our method. The stabilized enhancement based on different methods demonstrates the flexibility of our proposed GAAVE.

Introduction

As a window to the mind, facial expressions reflect human emotions and intents in daily communication. Automatically recognizing facial expression is important for artificial intelligence to comprehend human behaviors and interact with them. Due to its potential applications in various fields, such as intelligent tutoring systems, service robots, automatic driving and mental health analysis, Automatic Facial Expression Recognition has attracted increasing attention in the computer vision community recently (Li and Deng 2020).

Over the past decades, the community has made significant progress in the recognition of facial expression data, which is collected in a lab-controlled environment. Such in-the-lab datasets like CK+ (Lucey et al. 2010) and RaFD (Langner, Dotsch et al. 2010) have relative small sizes. Recently, with the emerge of large-scale datasets collected in real-world scenarios such as FER2013 (Goodfellow et al. 2013), RAF-DB (Li, Deng, and Du 2017) and Affectnet (Mollahosseini et al. 2017), FER is promoted from lab-controlled to in-the-wild settings (Li and Deng 2020). However, these real-world FER datasets usually exhibit noisy annotations (Li and Deng 2020) due to crowd-sourcing annotations, the subjectivity of the annotators and ambiguity in expressions. As illustrated in Fig. 1, we show some examples in AffectNet dataset with noisy labels. The presence of these noisy annotations is an impediment to the construction of FER methods, especially for the data-driven deep learning based FER.

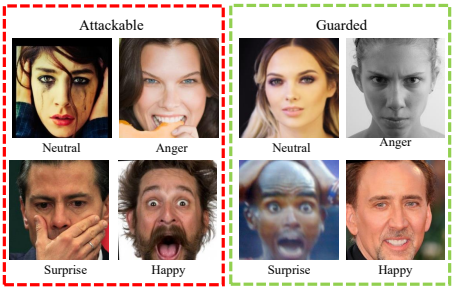

Figure 1: Illustration of the samples with noisy label (left box) and correct label(right box). The samples with noisy label are attackable by the proposed GAAVE, while clean samples are guarded.

Handling noisy annotations in DNNs is a challenging task (Arpit et al. 2017). The ideal way is to correct the noisy labels by estimating the noise transition matrix (Yao et al. 2020; Patrini et al. 2017; Tanno et al. 2019). However, obtaining an accurate noise transition matrix is extremely difficult in real-world scenarios. Another approach is to diminish the influence of corrupted labels during the training phase. These methods (Malach and Shalev-Shwartz 2017; Han et al. 2018; Jiang, Zhou, and Leung 2018; Yu et al. 2019) attempt to spot noisy training data and adjust the training strategy accordingly. Unfortunately, the additional information they generally require, such as noise distribution and clean data, is hard to obtain in real-world scenarios. Additionally, it is worth pointing out that these FER datasets also prevalently exhibit extreme class imbalance. This complicated situation makes training adjustment methods, like reweighting (Wang, Peng, and Yang 2020) and robust regular-

*Both author contributed equally to this work. Work done during Jiawen Zheng's internship at Tencent Youtu Lab and Bo Li is the project lead.

‡Corresponding author.

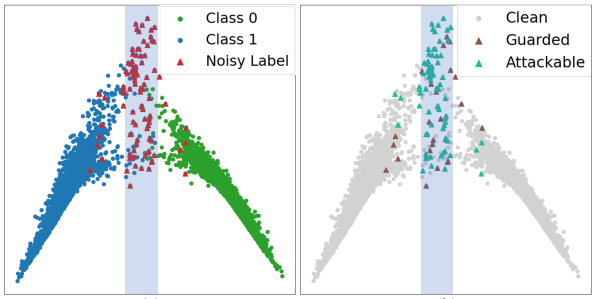

Figure 2: Two-dimensional PCA projections of samples. We randomly selected 440 samples from two classes of AffectNet. (a) Blue and green circles denote the samples of two classes respectively, and the red triangles denote samples with the noisy label. (b) The triangles denote the samples with noisy label, where the light blue triangles denote the samples evaluated as noise candidates through GAAVE.

ization (Zhang 2018), no longer feasible for FER. Specifically, tail-class samples and noisy label samples usually exhibit similar properties, like larger loss value, thus causing confusion between them.

To address the aforementioned issues, in this paper, we propose a novel and flexible method to spot noisy labels with no reliance on additional information by leveraging adversarial attack. It is well known that Deep Neural Networks (DNNs) are surprisingly vulnerable to malicious adversarial attack: adding small, perceptually indistinguishable perturbations to the data can easily cause disastrous misprediction from the model (Szegedy et al. 2013). But, how does adversarial attack help spot noisy labels? From previous works we find two enlightening observations: (1) for a DNN trained with noisy labels, data near the decision boundary is harder to distinguish and more likely to be mislabeled (Zhang et al. 2021); (2) memorization of label noise does cause significant adversarial vulnerability (Sanyal, Dokania et al. 2020). Inspired by these observations we get a deduction: mislabeled data are usually geometrically close to the decision boundary and thus relatively more attackable. This brings a feasible solution for spotting noisy data using adversarial attack: we attack all training data with small adversarial perturbations and the data which are easier attacked are selected as mislabeled. By means of demo examples, we demonstrate how to use adversarial attack to spot noisy labels in Fig. 2.

Specifically, we design a Geometry Aware Adversarial Vulnerability Estimation (GAAVE) method to spot more attackable data in the training set as noise candidates. Then we use the remaining clean data to relabel these candidates. Finally, with the clean and relabeled data, we can diminish the influence of corrupted labels in training and get a more promising model to better handle the FER task in real-world. It is worth noting that to alleviate the influence of imbalanced data distribution, we first divide the dataset and then apply GAAVE to spot the noisy data in each subset.

Our main contributions are summarized as below:

- We propose a novel and feasible adversarial approach called Geometry Aware Adversarial Vulnerability Estimation (GAAVE) to diminish the influence of noisy labels in real-world FER scenarios. The novel GAAVE method can differentiate corrupted labels and correct labels by estimating their adversarial vulnerability without any additional information of the noise distribution.
- To alleviate the complex situation caused by the entanglement of noisy annotations and imbalanced data distribution, we propose a divide-and-conquer strategy to split the whole training set into two relatively balanced subsets and then spot the noisy labels separately.
- The proposed approach does not involve any changes to the network structure and puts no additional burden on inference. With such high flexibility, it can therefore be promptly applied to any state-of-the-art network architecture. We conduct extensive experiments on large-scale real-world FER datasets to demonstrate the improved recognition results with the proposed method.

Related Work

Facial Expression Recognition

Over the past years, with the emergence of large-scale datasets collected in real-world scenarios, FER is promoted from lab-controlled to in-the-wild settings. Following the success of DNNs in computer vision tasks (Li et al. 2019c; Li, Sun, and Guo 2019; Li et al. 2019a,d,b; Tang and Li 2020; Tang et al. 2021; Zhong et al. 2021; Tang et al. 2022; Zhong et al. 2022; Tang and Li 2021; Zhang et al. 2022c; Li et al. 2022; Zhang et al. 2022a), automatic Facial Expression Recognition (FER) has been improved via Deep learning. Consequently, trained with relatively sufficient data the deep learning methods (Vo et al. 2020; Farzaneh and Qi 2020; Zhao 2021; Zhang, Wang, and Deng 2021) have exceeded traditional methods (Zhi et al. 2010; Zhong et al. 2012) by a large margin. Some works (Gera et al. 2021; Wang et al. 2020; Ruan et al. 2021; Shi, Zhu, and Liang 2021) focus on learning better representations to handle the real-world FER scenarios with occlusion, pose and illumination variations. She et al. (She, Hu, and Shi 2021) propose the latent Distribution Mining and the pairwise Uncertainty Estimation (DMUE) to investigate the ambiguity problem in FER. Wang et al. (Wang, Peng, and Yang 2020) propose Self-Cure Network (SCN) to reduce the impact of uncertainties. Zhang et al. (Zhang et al. 2022d) propose Erasing Attention Consistency (EAC) method to suppress the noisy samples during the training process automatically. Despite this, little research has been conducted on the noisy annotation problem in real-world FER scenarios.

Although Noisy Label Learning (NLL) is a classic task in machine learning, directly incorporating the classic NLL techniques is not an ideal cure for this problem. Some methods attempt to correct the noisy labels by estimating the noise transition matrix (Yao et al. 2020; Patrini et al. 2017; Tanno et al. 2019). But they require additional information in general, such as noise ratio or clean validation data, which is extremely difficult to obtain in real-world FER scenarios. Besides, the co-training architectures help reduce the confirmation bias (Malach and Shalev-Shwartz 2017; Han et al.

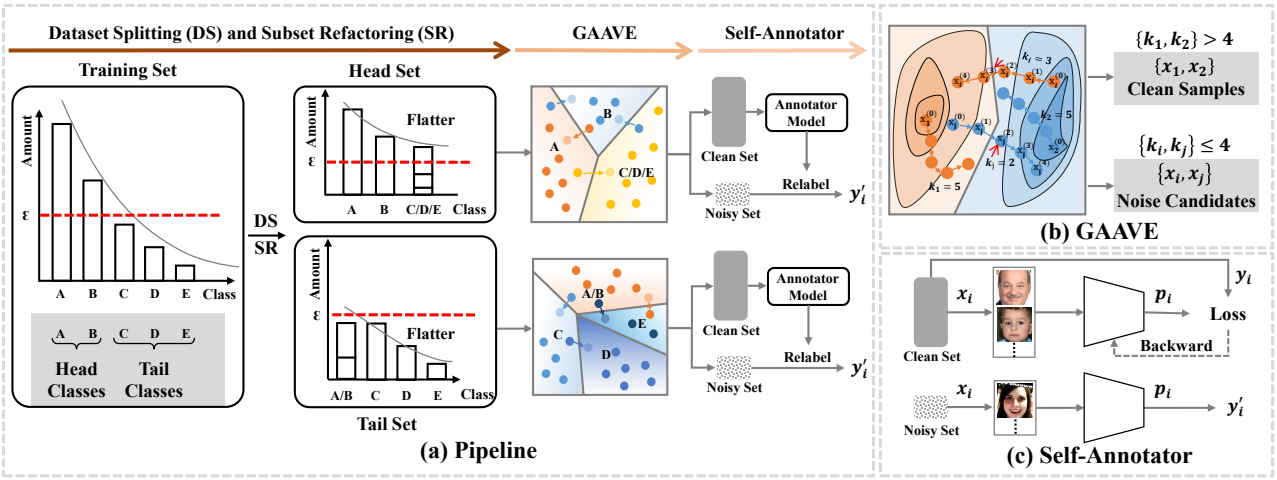

Figure 3: Overview of the proposed method. (a) The complete pipeline consists of three steps, namely Dataset Splitting and Subset Refactoring, GAAVE, and Self-Annotator in that order. (b) Visualization of the adversarial sample generation and the method of spotting the noisy labels by GAAVE. For a better explanation, we take x_1 and x_i as examples. x_1 does not cross the decision boundary. In other words, including the starting point and the four generated adversarial samples, there are five $p_1^{(s)}$ which equals y_1. Referring to Eq.4, then $k_1 = 5$. Similarly, for x_i, three samples do not cross the decision boundary, then $k_i = 3$. (c) With GAAVE dividing the original dataset into four subsets, we train the Self-Annotators using two clean subsets respectively and assign new labels to the samples of noisy subsets.

2018; Jiang, Zhou, and Leung 2018; Yu et al. 2019; Gera and Balasubramanian 2021), which is a hazard of favoring the examples selected at the beginning of training. However, the complex multi-network limit the flexibility of their application, and the small-loss trick they commonly used, which treats a certain number of small-loss training examples as clean examples, further limit them in particular noise scenario (Song et al. 2019). Moreover, some methods (Zhang 2018; Wang et al. 2019) attempt to design robust loss functions to minimize the side effects of noisy labels. They are flexible and usually have no need for inaccessible additional information, but unfortunately, they are no longer feasible facing the complicated situation, *i.e.*, the FER datasets. Specifically, tail-class samples and noisy label samples usually exhibit similar properties, like larger loss value. That is, the methods base on small-loss trick and robust loss function methods have to face the confusion between tail-class samples and noisy label samples, which causes that the clean tail-class samples are misclassified into noisy label samples easily.

Adversarial Attack

In contrast to the aforementioned superiority of Deep learning, the early publications (Szegedy et al. 2013; Nguyen, Yosinski, and Clune 2015) prove that Deep Neural Networks (DNNs) are surprisingly vulnerable to malicious adversarial attack: adding small, perceptually indistinguishable perturbations to the data can easily cause disastrous misprediction from the model (Li et al. 2021; Chen et al. 2022b; Zhang et al. 2022b; Chen et al. 2022a; Zhang et al. 2023). This approach of deceiving a trained model through subtle perturbations is Adversarial Attack. The magnitude of the attack is upper bounded by ε, which means that the attack is limited to a closed ball, centered on the input image x node and with ε as the radius. Specifically, Nguyen et al. (Nguyen, Yosinski, and Clune 2015) propose a classical method, namely Fast Gradient Sign Method (FGSM), to generate adversarial examples with a single gradient step as follows:

$$x_* = x + \varepsilon * \text{sgn}(\bigtriangledown_x L(x, y)), \tag{1}$$

where sgn is the sign function and L is the cost function; $\bigtriangledown_x$ denotes the gradient of the model with respect to a normal input x with label y.

Although FGSM can attack the linear model with ease, it is not always possible to succeed against a complex nonlinear model. Therefore, Madry et al. (Madry et al. 2017) use multi-step variant FGSM^k, *i.e.*, Projected Gradient Descent (PGD), to replace the simple one-step solution of FGSM. Recently, adversarial attack has been widely used in other areas. Several works (Zhu et al. 2021; Sanyal, Dokania et al. 2020) study the combination of adversarial attack and noisy label. Concretely, they use adversarial training to improve the robust accuracy, which avoids overfitting partial label noise, and they theoretically analyzed the relationship between adversarial training and noisy label. However, they lack extensive experiments to validate their hypothesis, and they use adversarial training while we use adversarial attack to filter samples, which is the major difference between us.

Method

The full pipeline of the proposed FER method includes noisy label spotting, noisy label correction, and re-train the FER network with purified datasets. An overview of full FER pipeline of our method is shown in Fig. 3. In this work,

we focus on the noisy label spotting and noisy label correction. In the following we elaborate on the core contributions of our work: (1) Before we spot the noisy candidates, we want to decouple the noisy data from the imbalanced distribution. A dataset splitting (DS) and a subset refactoring (SR) strategy are proposed to alleviate the distribution imbalance; (2) Geometry Aware Adversarial Vulnerability Estimation (GAAVE), which spots noisy labels using adversarial attack without any additional information; (3) Self-Annotator, which provides more plausible pseudo labels by two asymmetric annotators.

Dataset Splitting and Subset Refactoring

Notation. Give a FER dataset $\mathbb{D} = \{(x_i, y_i)\}$, in which each input image x_i corresponds to label y_i over C classes.

Dataset Splitting. FER datasets prevalently exhibit extreme class imbalance, which means that head classes will contain most of the data, while tail classes may have only 2.7% of the former (Mollahosseini et al. 2017). With the limited number of training samples, the feature space of tail classes will be squeezed by the head classes (Zhong et al. 2019), which makes learning with noisy label take more risks. Specifically, partial samples of tail classes usually exhibit similar properties to noisy labels, such as having a relatively high loss, which leads to confusion between noisy labels and tail classes in noisy label learning (Song et al. 2022). Therefore, we need to consider how to confront the complex situation caused by the entanglement of noisy annotations and imbalanced data distribution at first.

Facing the class imbalance problem, two categories of methods are usually used, respectively Re-sampling (Shen, Lin, and Huang 2016; More 2016) and Re-weighting (Khan, Hayat, and Bennamoun 2017; Wang, Ramanan, and Hebert 2017). But, the high entanglement of noisy labels and class imbalance makes some long-tail methods no longer feasible for FER. For example, a mislabeled tail-class sample will be used dozens of times when the traditional Re-sampling methods are used. So, we consider splitting the training set $\mathbb{D}$ into head-set $\mathbb{D}_h$ with C_1 classes and tail-set $\mathbb{D}_t$ with C_2 classes according to the category distribution, where $C = C_1 + C_2$. Dataset Splitting ensures the class distribution within each subset is relatively balanced, minimizing the impact of class imbalance. Meanwhile, compared with Re-weighting and Re-sampling, Dataset Splitting is a more soft strategy, despite its dependence on the current label, since it does not cause error amplification even if a sample is mislabeled.

Subset Refactoring. As we mentioned in the introduction, the core technique of this paper is spotting the noisy data through their adversarial vulnerability. Therefore, we need to provide a complete space in which the samples can be misclassified into arbitrary classes after adversarial attack. However, directly applying Dataset Splitting limits the space, in which a tail-class sample can not be misclassified into head classes and vice versa. Therefore, for each subset (head/tail), we consider introducing an additional category to complete the space, which consists of samples from another subset. For the purpose of making the data distribution within each refactored subset relatively balanced, distinctive strategies are designed to generate the refactored head-set $\mathbb{D}'_h$ and the refactored tail-set $\mathbb{D}'_t$.

Formally, for $\mathbb{D}'_h$, we introduce all samples of $\mathbb{D}_t$ as the $C_1 + 1$ class, because the total number of samples in $\mathbb{D}_t$ is usually closed to that of one of the categories in $\mathbb{D}_h$. For $\mathbb{D}'_t$, we randomly select samples from each class of $\mathbb{D}_h$ in a certain proportion, which follows their original class proportion, as the $C_2 + 1$ class. And the total number of $C_2 + 1$ class samples equals the maximum sample number in a single class of tail-set $\mathbb{D}_t$.

GAAVE

In this section, we introduce the detailed process of spotting noisy labels using adversarial attack.

Why can we use adversarial attack to spot noisy labels? As described in the previous section, for a DNN trained with noisy labels, data near the decision boundary is harder to distinguish and more likely to be mislabeled (Zhang et al. 2021). As shown in Fig. 2 (a), the toy example demonstrates that data with noisy label (red triangle) is closer to the boundary of the two categories. Meanwhile, memorization of label noise does cause significant adversarial vulnerability (Sanyal, Dokania et al. 2020). As we see in Fig. 2 (b), most of the data with noisy label exhibit vulnerability to adversarial attack (light blue triangle). The high overlap in the distribution of attackable samples and noisy-label samples demonstrates the feasibility of adversarial attack in spotting noisy labels.

How can we use adversarial attack to spot noisy labels? After clarifying the consistency in the distribution of attackable samples and noisy-label samples, we convert the problem of spotting noisy labels into estimating the adversarial vulnerability of the sample, termed as Geometry Aware Adversarial Vulnerability Estimation (GAAVE).

First, a relatively stable decision boundary is essential. That is, we need to train models under noisy annotations. We can obtain two sets of model parameters, θ_h and θ_t respectively trained on $\mathbb{D}'_h$ and $\mathbb{D}'_t$. Notably, the models are required to fit the subsets as closely as possible so that they can distinguish between underfitted samples and noisy-label samples. Second, a classical and highly effective adversarial attack method, namely PGD (Madry et al. 2017) is used to attack the trained models, since it provides the number of projected gradient descent steps, *i.e.*, how many PGD iterations are required to produce a misclassified adversarial variant. Therefore, we can use PGD to quantify the difficulty of the attack for each sample. Intuitively, the smaller the number of steps required to produce a misclassified adversarial variant, the more vulnerable the sample is to adversarial attack. Additionally, for the selection of targets for the PGD, there are two settings for adversarial attacks, namely targeted and non-targeted, and the non-targeted PGD is used.

Formally, for a clearer description, we take a sample $(x_i^{(0)}, y_i)$ of head-set $\mathbb{D}'_h$ as the starting point. Given step size α, the method of generating adversarial sample is as follows:

$$x_i^{(s+1)} = \text{Proj}_{(x_i^{(0)}+\varepsilon)}(x_i^{(s)} + \alpha \cdot \text{sgn}(\bigtriangledown_{x_i^{(s)}} L(\theta_h, x_i^{(s)}, y_i))), \tag{2}$$

where the definition of ε, sgn and L are consistent with Eq. 1; α is the step size of PGD attack; $x_i^{(s)}$ is the s-th generated adversarial sample of $x_i^{(0)}$; $\text{Proj}_{(x_i^{(0)}+\varepsilon)}$ is the projection function that projects the adversarial data back into the ε-ball centered at $x_i^{(0)}$. We use the generated adversarial samples $x_i^{(s)}$ to attack the trained model for a predictions $p_i^{(s)}$. Making $p_i^{(s)}$ deviate from y_i is the practical goal of PGD, but we merely need to know at what point the sample crosses the decision boundary, *i.e.*, at what point $p_i^{(s)}$ is not equal to y_i. Therefore, we define the step size of the sample crossing the decision boundary as k_i, named geometric value of adversarial vulnerability, which is calculated by the following formula:

$$k_i = \sum_{s=0}^{\alpha} \mathbb{1}(p_i^{(s)} = y_i), \tag{3}$$

where $\mathbb{1}$ equals 1 if the equation holds. Given the geometric value of adversarial vulnerability k for each sample, we can obtain the statistics of whether the sample is a noise candidate or not according to Criterion 1, which indicates that the smaller k is, more likely to be a noisy candidate. We summarize the process of GAAVE to precisely describe how to use adversarial attack to spot noisy labels in Algorithm 1. Due to space limitations, we omit the parts of tail-set, which is consistent with the head-set.

Criterion 1. The sample x is a noisy label if its deviation step $k < \tau$ (τ_h for head-set, τ_t for tail-set).

Self-Annotator

We have so far partitioned the original dataset into four subsets, separately clean head-set $\mathbb{D}_h^c$, noisy head-set $\mathbb{D}_h^n$, clean tail-set $\mathbb{D}_t^c$ and noisy tail-set $\mathbb{D}_t^n$. It is unwise to reject noisy subsets directly because a lot of data will be wasted. To leverage all training data sufficiently, we consider assigning new labels to the samples of noisy subsets. As shown in Fig. 3 (c), we first train robust annotators, namely Self-Annotator, on two clean subsets, θ_h^c and θ_t^c, respectively. Both annotators use the same structure and initialization parameters. The purpose of training the two models separately is to alleviate the data imbalance. The head model is mainly used to spot and relabel data in the head categories, and the tail model is used to spot and relabel data in the tail categories.

Then, we generate two pseudo labels for each sample through two self-annotators as follows:

$$p_i^h = f(\theta_h^c, x_i) \quad \text{and} \quad p_i^t = f(\theta_t^c, x_i). \tag{4}$$

Finally, we need a trade-off between these two pseudo labels. Formally, given an input image x_i from two noisy subsets, the pseudo label y_i' is as follows:

$$y_i' = \begin{cases} p_i^h & p_i^h \neq C_1 + 1 \text{ and } p_i^t = C_2 + 1 \\ p_i^t & p_i^t \neq C_2 + 1 \text{ and } p_i^h = C_1 + 1 \\ -1 & \text{else} \end{cases}. \tag{5}$$

By leveraging the consistency confidence of both networks, when the results of the two models contradict each other, the sample will be discarded, which can reduce the risk of introducing new errors. Our relabel algorithm has two advantages: (1) training the annotators in the purified datasets reduces the effect of noisy labels; (2) two asymmetric annotators provide more plausible pseudo labels. So far, we obtain the relabeled noisy subsets and clean subsets. The final step is to merge these subsets into a new training set and re-train it. We use the performance of the model, which is trained on the new training set from scratch, on the test set to measure the performance of our method.

Algorithm 1: Geometry Aware Adversarial Vulnerability Estimation (GAAVE)

Input: head-set $\mathbb{D}_h'$ and tail-set $\mathbb{D}_t'$
Output: clean-sets $\mathbb{D}_h^c$, $\mathbb{D}_t^c$; noisy-sets$\mathbb{D}_h^n$, $\mathbb{D}_t^n$

#1 Training two base models
Initialization: the network parameters θ_h, θ_t; learning rate η_h, η_t; total epoch T_h and T_t.
for epoch=1,...,T_h **do**
  for mini-batch=1,..., total batch **do**
    Sample: a mini-batch$\{(x_i, y_i)\}_{i=1}^m$ from $\mathbb{D}_h'$,
    Update $\theta_h \leftarrow \theta_h - \eta_h \bigtriangledown_{\theta_h} \{\ell(f_{\theta_h}(x_i), y_i)\}$.
  Update η_h.
Update θ_t by the same process as above

#2 Adversarial Attack
Initialization: learning rate η; total epoch T_h' and T_t'.
for epoch=1,...,T_h' **do**
  for mini-batch=1,..., total batch **do**
    Sample: a mini-batch$\{(x_i, y_i)\}_{i=1}^m$ from $\mathbb{D}_h'$
    Update $\theta_h \leftarrow \theta_h - \eta_h \bigtriangledown_{\theta_h} \{\ell(f_{\theta_h}(x_i), y_i)\}$
    Calculate $k_i$via Eq. 2 and Eq. 3
    Record k_i of every sample
*** Collecting k_i of every sample on $\mathbb{D}_t'$ ***

#3 Splitting the dataset a second time.
Initialization: τ_h and τ_t.
for i=1,..., sample size of $\mathbb{D}_h'$ **do**
  if $k_i \leq \tau_h$ **then**
    Add the i-th sample as a noise candidate to $\mathbb{D}_h^n$
  else
    Add the i-th sample to $\mathbb{D}_h^c$
*** Splitting $\mathbb{D}_t'$ into $\mathbb{D}_t^c$ and $\mathbb{D}_t^n$ ***
return $\mathbb{D}_h^c$, $\mathbb{D}t^n$,$\mathbb{D}_t^c$, $\mathbb{D}_t^n$

Experiments

Evaluation on FER Datasets

Datasets. RAF-DB (Li, Deng, and Du 2017) contains 30,000 facial images annotated with basic or compound expressions by 40 trained human coders. In our experiments, we use 12,271 images for training and 3,068 images for testing, following the previous works. FERPlus (Barsoum et al. 2016) is extended from Fer2013 (Goodfellow et al. 2013). It consists of 28,709 training images, 3,589 validation images and 3,589 test images, and these images are all resized to 48×48. Following the previous works, we just use training

Method	Backbone	RAFDB		FERPlus		AffectNet-7	AffectNet-8
		Acc.(%)	Avg.Acc.(%)	Acc.(%)	Avg.Acc.(%)	Acc.(%)	Acc.(%)
SCN (Wang, Peng, and Yang 2020)	ResNet-18	87.03	78.09 †	88.01	66.87 †	63.40†	60.23
PSR (Vo et al. 2020)	VGG-16	88.98	80.78	89.75	-	63.77	60.68
SCAN (Gera et al. 2021)	ResNet-50	89.02	-	89.42	-	65.14	61.73
DACL (Farzaneh and Qi 2021)	ResNet-18	87.78	80.44	88.39 †	69.20 †	65.20	60.75 †
DMUE (She, Hu, and Shi 2021)	ResNet-18	88.76	81.02 ‡	88.64	69.89 ‡	65.85 ‡	62.84
DAN (Wen et al. 2021)	ResNet-18	89.70	85.32	-	-	65.69	62.09
EAC (Zhang et al. 2022d)	ResNet-18	89.99	-	89.64	-	65.32	-
GCE (Zhang 2018)	ResNet-18	86.30 †	79.71 †	87.14 †	67.90 †	61.71 †	60.25 †
Co-teaching (Han et al. 2018)	ResNet-18	87.32 †	77.09 †	87.03 †	65.02 †	62.31 †	60.98 †
Dual t (Yao et al. 2020)	ResNet-18	88.07 †	80.39†	88.33 †	65.32 †	62.69 †	61.05†
Baseline (He et al. 2016)	ResNet-18	86.93	76.04	88.20	68.48	60.34	59.38
+ Ours	ResNet-18	**89.29(+2.36)**	**81.69(+2.08)**	**89.83(+1.63)**	**71.41(+2.93)**	**65.60(+5.26)**	**62.78(+3.40)**
MA-Net (Zhao 2021)	ResNet-18	88.40	79.73	88.19 †	68.78 †	64.53	60.96
+ Ours	ResNet-18	**89.51(+1.11)**	**81.92(+2.19)**	**89.10(+0.91)**	**70.10(+1.32)**	**65.94(+1.41)**	**62.85(+1.89)**
ARM (Shi, Zhu, and Liang 2021)	ResNet-18	90.55	82.29	88.25 †	69.16 †	64.49	59.75
+ Ours	ResNet-18	**91.53(+0.98)**	**83.70(+1.41)**	**89.29(+1.04)**	**70.91(+1.75)**	**66.11(+1.62)**	**63.25(+3.50)**

Table 1: Comparison with the state-of-the-art results on the FER datasets. Accuracy (Acc.) and mean class accuracy (Avg.Acc.) are used. † denotes our implementation and ‡ denotes the results provided by the authors.

images and test images in our experiments. AffectNet (Mollahosseini et al. 2017) is currently one of the largest and most challenging FER datasets with 440,000 images collected from the Internet. Following the previous works, we conduct two separate sets of experiments on the AffectNet dataset, AffectNet-7, and AffectNet-8.

Experimental setup. We demonstrate the effectiveness and generalization of our approach based on three methods: baseline (He et al. 2016), which is ResNet-18 without any modification; MA-Net (Zhao 2021); ARM (Shi, Zhu, and Liang 2021).

For images on RAFDB and FERPlus, the aligned face region is obtained through MobileFace (Chen et al. 2018), and for ones on AffectNet, the aligned face region is obtained through the landmarks provided. Then, the face region is resized to 224×224. In all experiments, we use ResNet-18 (He et al. 2016) as backbone, which is pre-trained on the face recognition dataset MS-Celeb-1M (Guo et al. 2016). By default, we apply the Adam optimizer (Kingma and Ba 2014) with weight decay 1e-5 and initial learning rate 1e-3. Besides, following (Wang, Peng, and Yang 2020; Wang et al. 2020), oversampling is used in the AffectNet.

Comparison with state of the arts. We compare the proposed method to several SOTA FER methods on RAFDB, FERPlus and AffectNet datasets. Besides, following the description, we also add the NLL methods, respectively GCE (Zhang 2018), Co-teaching (Han et al. 2018) and Dual t (Yao et al. 2020). They are representative of the work in robust loss methods, noisy label spotting methods and label correction methods in NLL. Moreover, most of the methods use ResNet-18 as the backbone, except PSR (Vo et al. 2020) and SCAN (Gera et al. 2021) using larger backbones. We report the comparison results in Table. 1. From the results, our method outperforms the baseline (He et al. 2016) by 2.36, 1.63, 5.26 and 3.40 accuracy (%) on RAFDB, FERPlus, AffectNet-7 and AffectNet-8. The above experimental results show the effectiveness of our proposed method.

Evaluation on Synthetic Data

Synthetic Data. CIFAR-10 (Krizhevsky, Hinton et al. 2009) is one of the most commonly used benchmarks for NLL methods. To simulate the imbalanced distribution in FER datasets, we generate long-tailed CIFAR-10 following (Cui et al. 2019). We use the imbalance ratio μ to denote the ratio between the sample sizes of the most frequent and least frequent classes. In our experiments, we set μ as 50, denoted as CIAFR-10-LT. Following (Han et al. 2018), we introduce two types of noisy label, respectively Pair flipping and Symmetry flipping. We use noise rate ν to denote the probability of the label being flipped. In our experiments, we just set $\nu \in \{0.1, 0.2, 0.3\}$.

Baselines. We compare our method with the following state-of-art methods: (1) GCE (Zhang 2018), (2) SCE (Wang et al. 2019), (3) Co-teaching (Han et al. 2018), and (4) O2U-Net (Huang et al. 2019) and (5) Jo-SRC (Yao et al. 2021). All the baselines are re-implemented on their open-source codes with minor modifications to fit our setting. We use ResNet-18 (He et al. 2016) as the backbone without pre-training.

Experimental setup. We apply SGD optimizer with a momentum factor of 0.9 and weight decay 1e-5. The learning rate is initialized as 0.1 and uses a cosine annealing schedule to update it. The batch size is 128. The epoch is 200.

Comparison Results. We report the comparison results in Table. 2. On the CIFAR-10-LT, almost all of the comparison

Methods	Pair			Symmetry		
	10%	20%	30%	10%	20%	30%
Baseline	60.02 ±0.17	55.14 ±0.27	46.93 ±0.26	58.77 ±0.21	53.41 ±0.18	46.93 ±0.18
GCE	52.11 ±0.17	49.29 ±0.17	41.79 ±0.28	56.77 ±0.25	51.13 ±0.17	42.96 ±0.14
SCE	58.85 ±0.29	52.67 ±0.25	47.39 ±0.16	56.89 ±0.17	51.14 ±0.19	45.23 ±0.15
Co-teaching	56.48 ±0.17	48.64 ±0.17	36.55 ±0.16	56.95 ±0.26	51.49 ±0.16	46.57 ±0.17
O2UNet	55.92 ±0.21	52.79 ±0.29	47.63 ±0.22	56.31 ±029	51.56 ±0.33	46.09 ±0.15
Jo-SRC	57.47 ±0.23	51.23 ±0.20	44.46 ±0.22	58.10 ±0.27	52.30 ±0.37	46.23 ±0.29
Ours	**62.51** ±0.51	**57.12** ±0.51	**49.45** ±0.19	**61.17** ±0.22	**54.40** ±0.19	**48.32** ±0.16

Table 2: Average test accuracy (%) on CIFAR-10-LT over the last 10 epochs.

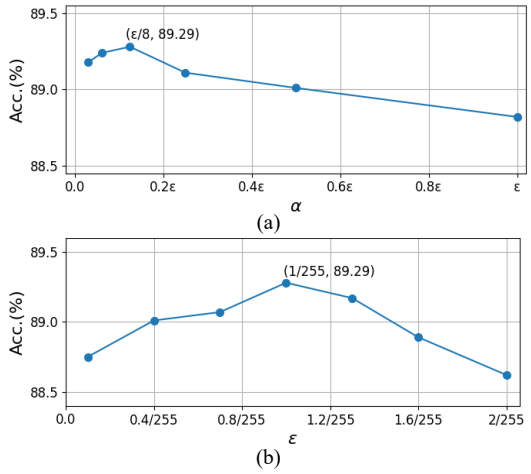

Figure 4: The accuracy (%) with different (a) α and (b) ε.

methods have side effects, which indicates that the imbalanced class distribution has a significant negative impact on the methods of noisy label learning. But, our method generally has a 1.0 to 2.5 accuracy (%) improvement.

Ablation Study

Influence of the adversarial hyper-parameters. There are two major hyper-parameters for the adversarial sample selection, which are step size α and maximum disturbance range ϵ in Eq. 2. We evaluate the influence of the adversarial hyper-parameters on the RAF-DB dataset. We report the experimental results in Fig. 4.

Specifically, for the common setting in adversarial attacks (Sanyal, Dokania et al. 2020; Madry et al. 2017), ε and α are set as 4/255 and ε/4. However, our GAAVE is designed to estimate the adversarial vulnerability of samples in a fine-grained manner. Large value of ε impairs the discrimination between noisy and clean samples, while too small value leads to incorrect passing of noisy samples. Therefore, we first set the maximum disturbance range ε as $1/255$ and set the value of α from $\varepsilon/32$ to ε. As shown in Fig. 4 (a), our methods obtain best performance when α is $\varepsilon/8$. Then, we set the step size α as $\varepsilon/8$. From the results of Fig. 4 (b), our methods obtain best performance when ε is $1/255$. Empirically, we set ε as $1/255$ and α as $\varepsilon/8$ in other experiments.

GAAVE	DS+SR	SA	RAF-DB	AffectNet-7
×	×	×	86.93	60.34
✓	×	×	88.04	63.37
×	✓	×	87.79	61.92
✓	×	✓	88.46	64.37
✓	✓	×	88.85	63.93
✓	✓	✓	89.29	65.60

Table 3: Evaluation of the key modules.

Evaluation of the key modules. To evaluate the effectiveness of each key module of our methods, we conduct an ablation study to GAAVE, Dataset Splitting and Subset Refactoring (DS+SR), and Self-Annotator (SA) on RAF-DB and AffectNet-7. From the experimental results in Table. 3, we can obtain several conclusions. First, the key modules GAAVE, DS+SR, and SA all deliver a general performance boost. Notably, on the AffectNet, SA brings significant improvement. AffectNet has a higher percentage of noisy labels, so removing the SA module would remove a lot of samples.

Conclusion

In this paper, we propose a novel and feasible method for differentiating noisy labels and clean labels for the FER datasets, which consists of Dataset splitting and Subset Refactoring, Geometry Aware Adversarial Vulnerability Estimation and Self-Annotator. The Dataset Splitting and Subset Refactoring first split and refactor the imbalanced dataset into two relatively balanced subsets. Then, the samples of two subsets are grouped into clean samples and noise candidates by GAAVE with no dependence on any additional information. Finally, Self-Annotators trained on the clean data assign new annotations for the noise candidates. Extensive experiments on large-scale real-world FER datasets prove the effectiveness and flexibility of our approach.

Acknowledgements

This work was supported by the National Science Fund for Distinguished Young Scholars (No.62025603), the National Natural Science Foundation of China (No. U21B2037, No. 62176222, No. 62176223, No. 62176226, No. 62072386, No. 62072387, No. 62072389, and No. 62002305).

References

Arpit, D.; et al. 2017. A closer look at memorization in deep networks. In *Proceedings of the ICML*, 233–242.

Barsoum, E.; Zhang, C.; Ferrer, C. C.; and Zhang, Z. 2016. Training deep networks for facial expression recognition

with crowd-sourced label distribution. In *Proceedings of the 18th ACM ICMI*, 279–283.

Chen, S.; Liu, Y.; Gao, X.; and Han, Z. 2018. Mobilefacenets: Efficient cnns for accurate real-time face verification on mobile devices. In *Chinese Conference on Biometric Recognition*, 428–438.

Chen, Z.; Li, B.; Wu, S.; Xu, J.; Ding, S.; and Zhang, W. 2022a. Shape Matters: Deformable Patch Attack. In *Computer Vision - ECCV 2022*. Springer.

Chen, Z.; Li, B.; Xu, J.; Wu, S.; Ding, S.; and Zhang, W. 2022b. Towards Practical Certifiable Patch Defense With Vision Transformer. In *Proceedings of the IEEE/CVF Conference on CVPR*, 15148–15158.

Cui, Y.; Jia, M.; Lin, T.-Y.; and Song. 2019. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF conference on CVPR*, 9268–9277.

Farzaneh, A. H.; and Qi, X. 2020. Discriminant distribution-agnostic loss for facial expression recognition in the wild. In *CVPR Workshops*, 406–407.

Farzaneh, A. H.; and Qi, X. 2021. Facial expression recognition in the wild via deep attentive center loss. In *Proceedings of the IEEE/CVF WACV*, 2402–2411.

Gera, D.; and Balasubramanian, S. 2021. Consensual Collaborative Training And Knowledge Distillation Based Facial Expression Recognition Under Noisy Annotations. *arXiv preprint arXiv:2107.04746*.

Gera, D.; et al. 2021. Landmark guidance independent spatio-channel attention and complementary context information based facial expression recognition. *Pattern Recognition Letters*, 145: 58–66.

Goodfellow, I. J.; Erhan, D.; Carrier, P. L.; et al. 2013. Challenges in representation learning: A report on three machine learning contests. In *CONIP*, 117–124.

Guo, Y.; Zhang, L.; Hu, Y.; He, X.; and Gao, J. 2016. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In *Proceedings of the ECCV*, 87–102.

Han, B.; Yao, Q.; Yu, X.; Niu, G.; Xu, M.; and Sugiyama, M. 2018. Co-teaching: Robust training of deep neural networks with extremely noisy labels. *NeurIPS*, 31.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on CVPR*, 770–778.

Huang, J.; Qu, L.; Jia, R.; and Zhao, B. 2019. O2u-net: A simple noisy label detection approach for deep neural networks. In *Proceedings of the IEEE/CVF ICCV*, 3326–3334.

Jiang, L.; Zhou, Z.; and Leung, T. 2018. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *Proceedings of the ICML*, 2304–2313.

Khan, S. H.; Hayat, M.; and Bennamoun, M. 2017. Cost-sensitive learning of deep feature representations from imbalanced data. *IEEE transactions on NeurIPS*, 29(8).

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Krizhevsky, A.; Hinton, G.; et al. 2009. *Learning multiple layers of features from tiny images*. Master's thesis, Department of Computer Science, University of Toronto.

Langner, O.; Dotsch, R.; et al. 2010. Presentation and validation of the Radboud Faces Database. *Cognition and emotion*, 24(8): 1377–1388.

Li, B.; Sun, Z.; and Guo, Y. 2019. SuperVAE: Superpixelwise Variational Autoencoder for Salient Object Detection. In *The Thirty-Third AAAI Conference*.

Li, B.; Sun, Z.; Li, Q.; Wu, Y.; and Hu, A. 2019a. Group-Wise Deep Object Co-Segmentation With Co-Attention Recurrent Neural Network. In *2019 IEEE/CVF International Conference on ICCV*, 8518–8527. IEEE.

Li, B.; Sun, Z.; Tang, L.; and Hu, A. 2019b. Two-B-real Net: Two-branch Network for Real-time Salient Object Detection. In *IEEE International Conference on ICASSP*. IEEE.

Li, B.; Sun, Z.; Tang, L.; Sun, Y.; and Shi, J. 2019c. Detecting Robust Co-Saliency with Recurrent Co-Attention Neural Network. In Kraus, S., ed., *IJCAI*.

Li, B.; Tang, L.; Kuang, S.; Song, M.; and Ding, S. 2022. Toward Stable Co-Saliency Detection and Object Co-Segmentation. *IEEE Trans. Image Process.*, 31: 6532–6547.

Li, B.; Xu, J.; Wu, S.; Ding, S.; Li, J.; and Huang, F. 2021. Detecting Adversarial Patch Attacks through Global-local Consistency. In *ADVM '21*, 35–41. ACM.

Li, B.; et al. 2019d. Co-saliency Detection Based on Hierarchical Consistency. In Amsaleg, L.; et al., eds., *Proceedings of the 27th ACM International Conference on MM*, 1392–1400. ACM.

Li, S.; and Deng, W. 2020. Deep facial expression recognition: A survey. *IEEE transactions on affective computing*.

Li, S.; Deng, W.; and Du, J. 2017. Reliable crowdsourcing and deep locality-preserving learning for expression recognition in the wild. In *CVPR*, 2852–2861.

Lucey, P.; et al. 2010. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In *2010 CVPR-workshops*, 94–101.

Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; and Vladu, A. 2017. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*.

Malach, E.; and Shalev-Shwartz, S. 2017. Decoupling" when to update" from" how to update". *NeurIPS*, 30.

Mollahosseini, A.; et al. 2017. Affectnet: A database for facial expression, valence, and arousal computing in the wild. *IEEE Transactions on Affective Computing*, 10(1): 18–31.

More, A. 2016. Survey of resampling techniques for improving classification performance in unbalanced datasets. *arXiv preprint arXiv:1608.06048*.

Nguyen, A.; Yosinski, J.; and Clune, J. 2015. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Proceedings of the IEEE conference on CVPR*, 427–436.

Patrini, G.; Rozza, A.; Krishna Menon, A.; Nock, R.; and Qu, L. 2017. Making deep neural networks robust to label noise: A loss correction approach. In *CVPR*.

Ruan, D.; Yan, Y.; Lai, S.; Chai, Z.; and Shen, C. 2021. Feature Decomposition and Reconstruction Learning for Effective Facial Expression Recognition. In *CVPR*.

Sanyal, A.; Dokania, P. K.; et al. 2020. How benign is benign overfitting? *arXiv preprint arXiv:2007.04028*.

She, J.; Hu, Y.; and Shi, H. 2021. Dive into Ambiguity: Latent Distribution Mining and Pairwise Uncertainty Estimation for Facial Expression Recognition. In *CVPR*.

Shen, L.; Lin, Z.; and Huang, Q. 2016. Relay backpropagation for effective learning of deep convolutional neural networks. In *Proceedings of the ECCV*, 467–482.

Shi, J.; Zhu, S.; and Liang, Z. 2021. Learning to amend facial expression representation via de-albino and affinity. *arXiv preprint arXiv:2103.10189*.

Song, H.; Kim, M.; Park, D.; and Lee, J.-G. 2019. How does Early Stopping Help Generalization against Label Noise? *arXiv preprint arXiv:1911.08059*.

Song, H.; Kim, M.; Park, D.; Shin, Y.; and Lee, J.-G. 2022. Learning from noisy labels with deep neural networks: A survey. *IEEE TNNLS*.

Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; and Fergus, R. 2013. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Tang, L.; and Li, B. 2020. CLASS: Cross-Level Attention and Supervision for Salient Objects Detection. In Ishikawa, H.; Liu, C.; Pajdla, T.; and Shi, J., eds., *ACCV 2020*.

Tang, L.; and Li, B. 2021. CoSformer: Detecting co-salient object with transformers. *arXiv preprint arXiv:2104.14729*.

Tang, L.; Li, B.; Kuang, S.; Song, M.; and Ding, S. 2022. Rethinking the relations in co-saliency detection. *IEEE Transactions on Circuits and Systems for Video Technology*.

Tang, L.; Li, B.; Zhong, Y.; Ding, S.; and Song, M. 2021. Disentangled High Quality Salient Object Detection. In *2021 IEEE/CVF ICCV*, 3560–3570. IEEE.

Tanno, R.; et al. 2019. Learning from noisy labels by regularized estimation of annotator confusion. In *Proceedings of the IEEE/CVF Conference on CVPR*, 11244–11253.

Vo, T.-H.; Lee, G.-S.; Yang, H.-J.; and Kim, S.-H. 2020. Pyramid with super resolution for In-the-Wild facial expression recognition. *IEEE Access*, 8: 131988–132001.

Wang, K.; Peng, X.; and Yang, J. 2020. Suppressing uncertainties for large-scale facial expression recognition. In *Proceedings of the IEEE/CVF Conference on CVPR*.

Wang, K.; Peng, X.; Yang, J.; Meng, D.; and Qiao, Y. 2020. Region attention networks for pose and occlusion robust facial expression recognition. *IEEE TIP*.

Wang, Y.; Ma, X.; Chen, Z.; Luo, Y.; Yi, J.; and Bailey, J. 2019. Symmetric cross entropy for robust learning with noisy labels. In *Proceedings of the IEEE/CVF ICCV*.

Wang, Y.-X.; Ramanan, D.; and Hebert, M. 2017. Learning to model the tail. In *NeurIPS*, 7032–7042.

Wen, Z.; Lin, W.; Wang, T.; and Xu, G. 2021. Distract your attention: multi-head cross attention network for facial expression recognition. *arXiv preprint arXiv:2109.07270*.

Yao, Y.; Liu, T.; Han, B.; Gong, M.; Deng, J.; Niu, G.; and Sugiyama, M. 2020. Dual t: Reducing estimation error for transition matrix in label-noise learning. *NeurIPS*.

Yao, Y.; et al. 2021. Jo-SRC: A Contrastive Approach for Combating Noisy Labels. In *Proceedings of the IEEE/CVF Conference on CVPR*, 5192–5201.

Yu, X.; Han, B.; Yao, J.; Niu, G.; and Tsang, I. 2019. How does disagreement help generalization against label corruption? In *Proceedings of the ICML*, 7164–7173.

Zhang, J.; Chen, C.; Li, B.; Lyu, L.; Wu, S.; Ding, S.; Shen, C.; and Wu, C. 2022a. DENSE: Data-Free One-Shot Federated Learning. In *Advances in NeurIPS*.

Zhang, J.; Li, B.; Chen, C.; Lyu, L.; Wu, S.; Ding, S.; and Wu, C. 2023. Delving into the Adversarial Robustness of Federated Learning. In *AAAI Conference*.

Zhang, J.; Li, B.; Xu, J.; Wu, S.; Ding, S.; Zhang, L.; and Wu, C. 2022b. Towards Efficient Data Free Black-Box Adversarial Attack. In *Proceedings of the CVPR*.

Zhang, J.; Li, Z.; Li, B.; Xu, J.; Wu, S.; Ding, S.; and Wu, C. 2022c. Federated Learning with Label Distribution Skew via Logits Calibration. In *Proceedings of the ICML*. PMLR.

Zhang, Y.; Wang, C.; and Deng, W. 2021. Relative Uncertainty Learning for Facial Expression Recognition. *Advances in NeurIPS*, 34: 17616–17627.

Zhang, Y.; Wang, C.; Ling, X.; and Deng, W. 2022d. Learn from all: Erasing attention consistency for noisy label facial expression recognition. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVI*, 418–434. Springer.

Zhang, Y.; Zheng, S.; Wu, P.; Goswami, M.; and Chen, C. 2021. Learning with feature-dependent label noise: A progressive approach. *arXiv preprint arXiv:2103.07756*.

Zhang, Z. 2018. Generalized cross entropy loss for training deep neural networks with noisy labels. In *NeurIPS*.

Zhao, Z. 2021. Learning Deep Global Multi-scale and Local Attention Features for Facial Expression Recognition in the Wild. *IEEE Trans. Image Process.*

Zhi, R.; et al. 2010. Graph-preserving sparse nonnegative matrix factorization with application to facial expression recognition. *IEEE Transactions on Systems*, 41(1): 38–52.

Zhong, L.; Liu, Q.; Yang, P.; Liu, B.; Huang, J.; and Metaxas, D. N. 2012. Learning active facial patches for expression analysis. In *2012 IEEE Conference on CVPR*.

Zhong, Y.; Li, B.; Tang, L.; Kuang, S.; Wu, S.; and Ding, S. 2022. Detecting Camouflaged Object in Frequency Domain. In *Proceedings of the CVPR*, 4504–4513.

Zhong, Y.; Li, B.; Tang, L.; Tang, H.; et al. 2021. Highly Efficient Natural Image Matting. *CoRR*, abs/2110.12748.

Zhong, Y.; et al. 2019. Unequal-training for deep face recognition with long-tailed noisy data. In *Proceedings of the IEEE/CVF conference on CVPR*, 7812–7821.

Zhu, J.; Zhang, J.; Han, B.; Liu, T.; and Sugiyama, M. 2021. Understanding the Interaction of Adversarial Training with Noisy Labels. *arXiv preprint arXiv:2102.03482*.