Dive into Ambiguity: Latent Distribution Mining and Pairwise Uncertainty Estimation for Facial Expression Recognition

Jiahui She[*1,2] Yibo Hu[*2] Hailin Shi[2] Jun Wang[2] Qiu Shen[†1] Tao Mei[2]
[1]Nanjing University [2]JD AI Research
jh.she@foxmail.com, huyibo871079699@gmail.com, {wangjun492, shihailin}@jd.com
shenqiu@nju.edu.cn, tmei@live.com

Abstract

*Due to the subjective annotation and the inherent inter-class similarity of facial expressions, one of key challenges in Facial Expression Recognition (FER) is the annotation ambiguity. In this paper, we proposes a solution, named DMUE, to address the problem of annotation ambiguity from two perspectives: the latent **D**istribution **M**ining and the pairwise **U**ncertainty **E**stimation. For the former, an auxiliary multi-branch learning framework is introduced to better mine and describe the latent distribution in the label space. For the latter, the pairwise relationship of semantic feature between instances are fully exploited to estimate the ambiguity extent in the instance space. The proposed method is independent to the backbone architectures, and brings no extra burden for inference. The experiments are conducted on the popular real-world benchmarks and the synthetic noisy datasets. Either way, the proposed DMUE stably achieves leading performance.*

1. Introduction

Facial expression plays an essential role in human's daily life. Automatic Facial Expression Recognition (FER) is crucial in real world applications, such as service robots, driver fragile detection and human computer interaction. In recent years, with the emerge of large-scale datasets, *e.g.* AffectNet [28], RAF-DB [24] and EmotioNet [6], many deep learning based FER approaches [11, 38, 49] have been proposed and achieved promising performance.

However, the ambiguity problem remains an obstacle that hinders the FER performance. Usually, facial images are annotated to one of several basic expressions for training the FER model. Yet the definition with respect to the expression category may be inconsistent among different people. For better understanding, we randomly pick two images from AffectNet [28] and conduct a user study. As shown in Fig. 1, for the image annotated with *Anger*, the most possible class decided by volunteers is *Neutral*. For the other image, the confidence gap between the most and secondary possible classes is only 20%, which means annotating it to a specific class is not suitable. In other words, a label distribution that depicts the possibility belonging to each class can better describe the visual feature. There are two reasons leading to the above phenomenon: (1) It is subjective for people to define which type of expression a facial image is. (2) With a large amount of images in large-scale FER datasets, it is expensive and time-consuming to provide label distribution of images. As there exists a considerable portion of ambiguous samples in large-scale datasets, the models are prevented from learning the robust visual features with respect to a certain type of expression, thus the performance has reached a bottleneck. The previous approaches tried to address this issue by introducing label distribution learning [11] or suppressing uncertain samples [38]. However, they still suffer from the ambiguity problem revealed in data that cannot be directly solved from the single instance perspective.

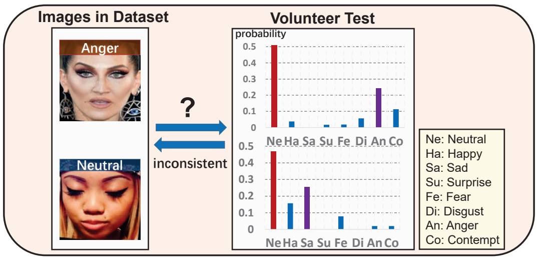


Figure 1: User study results by 50 volunteers on two randomly picked images. The red (purple) bar represents the most (secondary) possible class given by the volunteers. The results provide insights that the annotations may be inconsistent among the users.

In this paper, we propose a solution to address the ambiguity problem in FER from two perspectives, *i.e.* the latent **D**istribution **M**ining and the pairwise **U**ncertainty

[*] These authors contributed equally to this work.
[†] Corresponding author

Estimation (DMUE). For the former, several temporary auxiliary branches are introduced to discover the label distributions of samples in an online manner. The iteratively updated distributions can better describe the visual features of expression images in the label space. Thus, it can provide the model informative semantic features to flexibly handle ambiguous images. For the latter, we design an elaborate uncertainty estimation module based on pairwise relationships between samples. It jointly utilizes the original annotations and the statistics of relationships to reflect the ambiguity extent of samples. The estimated uncertain level encourages the model to dynamically adjust learning focus between the mined label distribution and original annotations. Note that our proposed framework is end-to-end training and has no extra cost for inference. All the auxiliary branches and the uncertainty estimation module will be removed during deployment. Overall, the main contributions can be summarized as follows:

- We propose a novel end-to-end solution to investigate the ambiguity problem in FER by exploring the latent label distribution of the given sample, without introducing extra burden on inference.
- An elaborate uncertainty estimation module is designed based on the statistics of relationships, which provides guidance for the model to dynamically adjust learning focus between the mined label distribution and annotations from sample level.
- Our approach is evaluated on the popular real-world benchmarks and synthetic noisy datasets. Particularly, it achieves the best performance by 89.42% on RAF-DB and 63.11% on AffectNet, setting new records.

2. Related Work

2.1. Facial Expression Recognition

Numerous FER algorithms [2, 25, 29, 37] have been proposed, which can be grouped into *handcraft and learning-based* methods. Early attempts [5, 29, 31] rely on handcraft features that reflect folds and geometry changes caused by expression. With the development of deep learning, *learning-based* methods [36, 37, 46] become the majority, such as decoupling the identity information [37] or exploiting the difference between expressive images [46].

In recent years, several attempts try to address ambiguity problem in FER. Zeng *et al.* [49] consider annotation inconsistency and introduce multiple training phases. Chen *et al.* [11] build nearest neighbor graphs for training data in advance and investigate label distribution of samples in a semi-online way. Previous leading performance has been achieved by Wang *et al.* [38]. They focus on finding the confidence weight and the latent truth of each sample to suppress harmful influence from ambiguous data. However, the compound expressions [24] and the original annotations could be jointly considered in estimating ambiguity.

2.2. Learning with Ambiguity Label

Mislabelled annotations and low data quality may result in ambiguity problem. For the former, learning with noisy label [3] is one of the most popular directions. Another direction is the uncertainty estimation [33, 41], such as MentorNet [19] and CleanLab [30]. In recent years, a promising way to handle mislabelled annotations is to find the latent truth [13], such as utilizing the prediction of model [7, 13, 15, 23, 43] or introducing auxiliary embeddings [17, 47]. For the latter, an universal way is to enhance the label [44, 45] of low quality images by the temperature softmax [7] or inject the artificial uncertainty [9, 32, 34]. Unlike prior methods [44, 45], the ambiguity problem in FER, *i.e.* compound expressions [24] exists in a more subjective way. The label description of a compound expression image is various among the users.

3. Method

Notation. Given a FER dataset $(\mathcal{X}, \mathcal{Y})$ in which each image $\boldsymbol{x}$ belongs to one of C classes, we denote $y_x \in \{1, 2, \cdots, C\}$ as its annotated deterministic class. However, as shown in Fig. 1, the exact type of $\boldsymbol{x}$ is inapparent or uncertain. We employ *latent distribution* $\widetilde{\boldsymbol{y}}_x$ to represent the probability distribution for $\boldsymbol{x}$ belonging to all possible classes except y_x. That is, $\widetilde{\boldsymbol{y}}_x \in \mathbb{R}^{C-1}$ is a distribution vector, $\|\widetilde{\boldsymbol{y}}_x\|_1 = 1$.

3.1. Overview of DMUE

To address the annotation ambiguity, we mine $\widetilde{\boldsymbol{y}}_x$ for each $\boldsymbol{x}$ and regularize the model to learn jointly from $\widetilde{\boldsymbol{y}}_x$ and y_x. Benefited from the semantic features of ambiguous samples, the performance of the model can be greatly improved.

For a toy experiment, we train a ResNet-18 on AffectNet [28] and present its prediction for a mislabelled training sample in Fig. 5, where a crying baby (*Sad*) is labelled with *Neutral*. We can observe that the predicted distribution reflects the truth class of the mislabelled sample. It inspires us to employ the predictions from a trained model to help a new model in training phase, where such mislabelled image may be tagged by a distribution reflecting its true class. By imposing the latent distribution $\widetilde{\boldsymbol{y}}_x$ as the additional supervision, model can utilize the latent semantic features to better deal with ambiguity samples, thus improve the performance. We employ the classifier trained by samples from negative classes of $\boldsymbol{x}$, *i.e.* samples from other $C-1$ classes except for y_x, to find its $\widetilde{\boldsymbol{y}}_x$, based on qualitative and quantitative analyses in Section 4.7. Moreover, to balance the

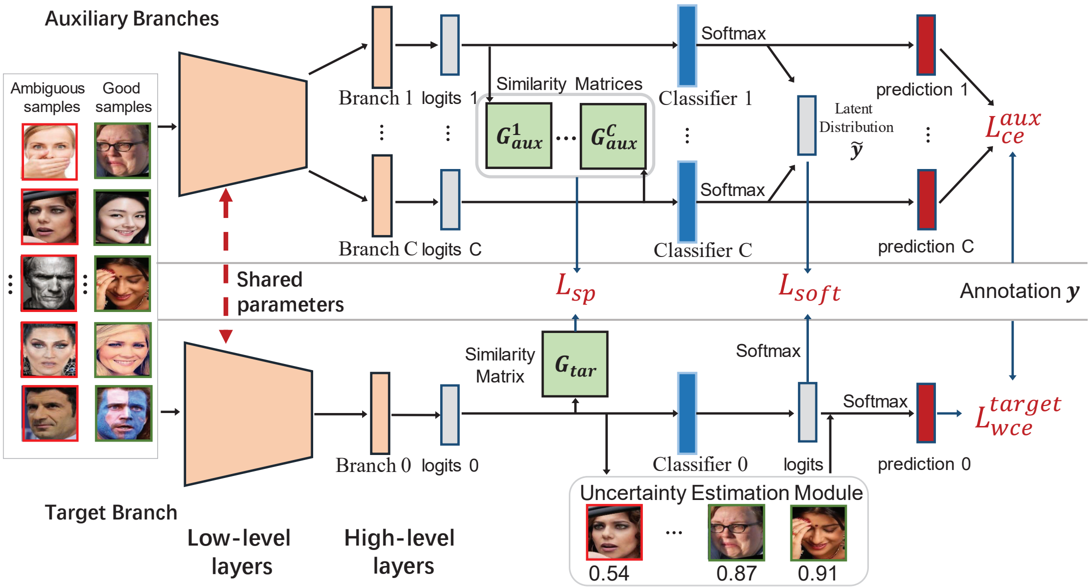


Figure 2: Overview of the DMUE. $\boldsymbol{y}$ denotes the set of annotations of images in a batch. $\widetilde{\boldsymbol{y}}$ denotes the set of mined latent distributions of images in a batch.

learning between the annotation and the mined $\widetilde{\boldsymbol{y}}_x$, an uncertainty estimation module is elaborately designed to guide the model to learn more from $\widetilde{\boldsymbol{y}}_x$ than y_x for those ambiguous samples.

An overview of DMUE is depicted in Fig. 2. The DMUE contains: (1) latent distribution mining with C auxiliary branches and one target branch that have the same architecture (*e.g*. the last stage of ResNet), and (2) pairwise uncertainty estimation, where an uncertainty estimation module is established by two fully connected (FC) layers. Each auxiliary branch is served as an individual $(C-1)$-class classifier aiming to find $\widetilde{\boldsymbol{y}}_x$ for the corresponding $\boldsymbol{x}$. $\widetilde{\boldsymbol{y}}_x$ and y_x are joint together to guide the target branch. Furthermore, we regularize the branches to predict consistent relationships of images by their similarity matrices. Note that all auxiliary branches and the uncertainty estimation module will be removed, and only the target branch will be reserved for deployment. *Therefore, our framework is end-to-end and can be flexibly integrated into existing network architectures without extra cost on inference.*

3.2. Latent Distribution Mining

As $\widetilde{\boldsymbol{y}}_x$ is predicted by the classifier trained with samples from negative classes of $\boldsymbol{x}$, If there are total C classes, then C classifiers need to be trained to predict the latent distribution of each sample. Considering the computational efficiency and the shared low-level features [10, 22], we propose a multi-branch architecture to construct these classifiers. As shown in Fig. 2, C auxiliary branches are introduced to predict the latent distributions and a target branch is employed for final prediction. Given a batch, the j-th branch predicts $\widetilde{\boldsymbol{y}}_x$ for $\boldsymbol{x}$ annotated to the j-th class. Thus, we can obtain $\widetilde{\boldsymbol{y}}_x$ for each $\boldsymbol{x}$ in batch by C auxiliary branches. Note all branches have the same structure (*e.g*. the last stage of ResNet) and share the common lower layers (*e.g*. the first three stages of ResNet). Classifier $j, j \in \{1, \cdots, C\}$ is $(C-1)$-class and Classifier 0 (the target classifier) is C-class for final deployment.

A comprehensive description of mini-batch training is presented in Algorithm 1. Given a batch, we use images not annotated to the j-th category to train the j-th auxiliary branch. In other words, each image $\boldsymbol{x}$ is utilized to train other $C-1$ auxiliary branches than the y_x-th branch. The Cross-Entropy(CE) loss L^{aux}_{CE} is employed for optimization:

$$\mathrm{L}^{aux}_{CE} = \frac{1}{C}\sum_{j=1}^{C} L^{aux_j}_{CE}, \tag{1}$$

$$\mathrm{L}^{aux_j}_{CE} = -\frac{1}{N_j}\sum_{p=1}^{N_j}\sum_{k=1,k\neq j}^{C} y_{x_p,k}\log f_j(\boldsymbol{x}_p;\theta)_k, \tag{2}$$

where $L^{aux_j}_{CE}$ is the CE loss for training the j-th branch, N_j is the number of $\boldsymbol{x}$ not annotated to j in the batch and p is index. $y_{x_p,k}$ is the label of $\boldsymbol{x}_p$ belonging to the k-th class and $f_j(\boldsymbol{x}_p;\theta)_k$ is the possibility of $\boldsymbol{x}_p$ belonging to the k-th class predicted by the j-th branch.

As described above, the prediction of the j-th auxiliary branch for $\boldsymbol{x}$ with annotation j, is used as its latent distribution $\widetilde{\boldsymbol{y}}_x \in \mathbb{R}^{C-1}$. One additional step, called Sharpen [7, 23, 42], is adopted before regularizing the target branch:

$$Sharpen(\widetilde{\boldsymbol{y}}_x, T)_i = \widetilde{y}_{x,i}^{\frac{1}{T}} / \sum\nolimits_j^{C-1} \widetilde{y}_{x,j}^{\frac{1}{T}}, \tag{3}$$

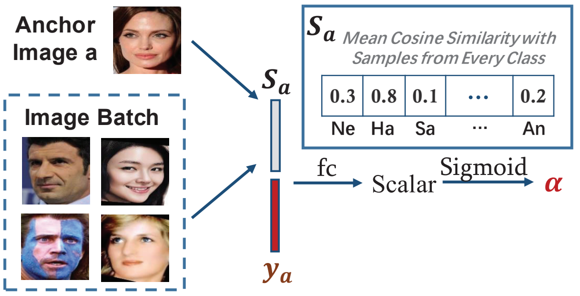


Figure 3: Uncertainty estimation module. y_a is the one-hot form of anchor image's annotation. S_a and y_a are concatenated to reflect how ambiguous the anchor image is.

where $\widetilde{y}_{x,i}$ the i-th element of $\widetilde{\boldsymbol{y}}_x$ and T is the temperature. Sharpen function provides the flexibility to slightly adjust the entropy of $\widetilde{\boldsymbol{y}}_x$. When $T > 1$, the output $Sharpen(\widetilde{\boldsymbol{y}}_x, T)$ will be more flatten than the original $\widetilde{\boldsymbol{y}}_x$.

After sharpening, we utilize L_2 loss to minimize the deviation between the prediction of target branch and the sharpened $\widetilde{\boldsymbol{y}}_x$, which is defined as:

$$L_{soft} = \frac{1}{N(C-1)} \sum_{p=1}^{N} \sum_{k=1,k\neq i}^{C} (\widetilde{y}_{x_p,k} - f_{target}(\boldsymbol{x}_p;\theta)_k)^2, \tag{4}$$

where N is the batch size. $\widetilde{y}_{x_p,k}$ is the possibility of $\boldsymbol{x}_p$ belonging to the k-th class in the latent distribution $\widetilde{\boldsymbol{y}}_{x_p}$, and $f_{target}(\boldsymbol{x}_k;\theta)_j$ is the prediction of target branch. The reason employing L2 loss is that unlike Cross-Entropy, the L2 loss is bounded and less sensitive to inaccurate predictions. We do not back propagate gradients through computing $\widetilde{\boldsymbol{y}}$.

Similarity Preserving. Inspired by [35], we find it is beneficial to regularize all the branches to predict consistent relationship when given a pair of images. This is because CE loss only utilizes samples individually from the label space. However, the relationship between samples is another knowledge paradigm. For instance, given a pair of *smiling* images, besides telling network their annotations *Happy*, the similarities of their semantic features extracted by different branches should be consistent. Thus, we generalize [35] to the context of multi-branch architecture as multi-branch similarity preserving (MSP), defined as:

$$L_{sp} = MSP(G^1_{aux}, \cdots, G^C_{aux}, G_{tar}), \tag{5}$$

where $G^i_{aux} \in \mathbb{R}^{B_i \times B_i}$ and $G_{tar} \in \mathbb{R}^{B \times B}$ are the similarity matrices calculated by the semantic features in auxiliary and target branches, respectively. Their elements reflect the pairwise relationships between samples. L_{sp} aims at sharing the relation information across branches. *Specific computation is provided in the supplementary material.*

3.3. Pairwise Uncertainty Estimation

To handle the ambiguous samples, we introduce latent distribution mining. However, the target branch should also be benefitted from clean samples. Directly employing CE loss may lead to improvement degradation due to the existing of ambiguous samples. Accordingly, we impose a modulator term into the standard CE loss to trade-off between the latent distribution and annotation in the sample space. Specifically, we estimate the confidence scores of the samples based on the statistics of their relationships. Lower score will be assigned to more ambiguous samples, further reducing the CE loss. Thus, the latent distribution will provide more guidance.

For better understanding, we choose an anchor image in a given batch to illustrate the uncertainty estimation module. As shown in Fig. 3, we denote the semantic feature and one-hot label of the anchor image as $(\boldsymbol{f}_a, \boldsymbol{y}_a)$, while others in the batch as $(\boldsymbol{f}_i, \boldsymbol{y}_i)$, where $\boldsymbol{f}$ is the feature before classifier in the target branch, i is the index. We calculate the average cosine similarity of $\boldsymbol{f}_a$ with each of $\boldsymbol{f}_i$ annotated with j-th category as $S_{a,j}$, and vector $\boldsymbol{S}_a = [S_{a,1}, \cdots, S_{a,C}]$. After that, $\boldsymbol{S}_a$ is concatenated with $\boldsymbol{y}_a \in \mathbb{R}^C$ to form $\boldsymbol{SV}_a \in \mathbb{R}^{2C}$, which reflects how ambiguous the anchor sample is:

$$\boldsymbol{SV}_a = concat(\boldsymbol{S}_a, \boldsymbol{y}_a), \tag{6}$$

$$S_{a,j} = \frac{1}{N_j} \sum_{i=1}^{N_j} \frac{\langle \boldsymbol{f}_a, \boldsymbol{f}_i \rangle}{\|\boldsymbol{f}_a\| \, \|\boldsymbol{f}_i\|}, \tag{7}$$

where $< \boldsymbol{f}_a, \boldsymbol{f}_i >$ is the dot product of $\boldsymbol{f}_a$ and $\boldsymbol{f}_i$. N_j is the number of samples whose annotation is j-th class in the batch and i is the index.

Here, We provide two perspectives to understand this delicate design: (**1**) For a mislabelled sample $(\boldsymbol{x}, y_x)$ in the given batch (*e.g.* the semantic feature of $\boldsymbol{x}$ belongs to i-th class but $y_x = j$), the average similarity of semantic features between $\boldsymbol{x}$ and the images in i-th class should be high. However, the concatenated y_x indicates $\boldsymbol{x}$ is annotated with i-th class. (**2**) A *clear* $\boldsymbol{x}(y_x = i)$ should only capture the typical semantic feature of i-th class. The average similarity of its semantic feature with other types of images should be discriminatively lower than with i-th class samples. Thus, $\boldsymbol{SV}_x$ can reveal the ambiguity information of $\boldsymbol{x}$.

Let $\boldsymbol{SV} = [\boldsymbol{SV}_1, \boldsymbol{SV}_2, \cdots, \boldsymbol{SV}_N] \in \mathbb{R}^{2C \times N}$ denotes the ambiguity information feature of a batch, the uncertainty estimation module takes $\boldsymbol{SV}$ as the input and outputs a confidence scalar $\alpha_i \in (0, 1)$ for each image. The module consists of two FC layers with a PRelu non-linear function and a sigmoid activation:

$$\alpha = Sigmoid(\boldsymbol{W}_2^T \sigma(\boldsymbol{W}_1^T \boldsymbol{SV}), \tag{8}$$

where $\boldsymbol{W}_1 \in \mathbb{R}^{2C \times C}$ and $\boldsymbol{W}_2 \in \mathbb{R}^{C \times 1}$ are the parameters of two FC layers,σ is the PReLU activation.

With the estimated confidence score, we perform weighted training in the target branch. Directly multiplying the score with CE loss may obstruct the uncertainty estimation, because it will make the estimated score to be zero [38]. Therefore, we alternatively multiply the score with the output logit of the classifier in the target branch. The weighted CE loss [16, 38] is formulated as:

$$L_{WCE}^{t\arg et} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{\alpha_i W_{y_i}^T f_i}}{\sum_{j=1}^{C} e^{\alpha_i W_j^T f_i}}. \tag{9}$$

Obviously, L_{WCE}^{target} has positive correlation with the score α [26]. Thus, for ambiguous samples, the estimated scores are small, reducing the impact of CE loss, and the target branch learns more from the mined latent distributions.

3.4. Overall Loss function

The overall objective of DMUE is:

$$L_{total} = w_u(e)(L_{WCE}^{target} + \omega L_{soft} + \gamma L_{sp}) + w_d(e)L_{CE}^{aux}, \tag{10}$$

where ω, γ are the hyperparameters. w_u and w_d are the weighted ramp functions [21] w.r.t. the epoch e, which is formulated as:

$$w_u(e) = \begin{cases} \exp(-(1-\frac{e}{\beta})^2) & e \le \beta \\ 1 & e > \beta \end{cases}, \tag{11}$$

$$w_d(e) = \begin{cases} 1 & e \le \beta \\ \exp(-(1-\frac{\beta}{e})^2) & e > \beta \end{cases}, \tag{12}$$

where β is the epoch threshold for functions. where β is the epoch threshold. The Eq. 11 and 12 are introduced to benefit training from two aspects: (1) At the beginning of training, the latent distributions mined by auxiliary branches are not stable enough. Thus, we focus on training the auxiliary branches. (2) When the auxiliary branches are well trained, we then divert our attention to train the target branch.

It worth noting that we remove all the auxiliary branches and the uncertainty estimation module for deployment. *Our framework is end-to-end and can be flexibly integrated with existing network architectures, without extra cost on inference.*

4. Experiments

We verify the effectiveness of DMUE on synthetic noisy datasets and 4 popular in-the-wild benchmarks, and further validate the contribution of each component of DMUE. Extensive ablation studies with respect to the hyperparameters and the different backbone architectures are carried out to confirm the advantage of our method.

4.1. Datasets and Metrics

RAF-DB [24] is constructed by 30,000 facial images with basic or compound annotations. In the experiment, we choose the images with seven basic expressions (*i.e.* neutral, happiness, surprise, sadness, anger, disgust and fear), of which 12,271 are used for training, and the remaining 3,068 for testing. **AffectNet** [28] is currently the largest FER dataset, including 440,000 images. The images are collected from the Internet by querying the major search engines with 1,250 emotion-related keywords. Half of the images are annotated with eight basic expressions, providing 280K training images and 4K testing images. **FERPlus** [4] is an extension of FER2013 [14], including 28,709 training images and 3,589 testing images resized to 48×48 gray-scale pixels. Each image is labelled by 10 crowd-sourced annotators to one of eight categories. For a fair comparison, the most voting category is picked as the annotation for each image following [4, 18, 38, 39]. **SFEW** [12] contains the images from movies with seven basic emotions, including 958 images for training and 436 images for testing. For each dataset, we report the overall accuracy on the testing set.

Algorithm 1: DMUE.

```
Input: Training Images $\mathcal{X}$ and annotations $\mathcal{Y}$ with
       $C$ classes, $MaxEpoch$, $num\_iters$
Output: Trained model with target branch $\theta^0$ and $C$
        auxiliary branches $\theta^j, j \in \{1, \cdots, C\}$
/* Training */
Initialize $\theta^0$ and $\theta^j$ with random values,
  $j \in \{1, 2, \cdots, C\}, e = 1$
while $e < MaxEpoch$ do
    for $k = 0$ to $num\_iters$ do
        From $(\mathcal{X}, \mathcal{Y})$, sample a batch $set_{batch}$;
          // Note samples in $j$-th class as $set_j$
        Compute $\mathcal{L}_{ce}^{aux}$ ;    // use $set_{batch} \backslash set_j$ to
          compute $\mathcal{L}_{ce}^{aux_j}$ for $\theta^j, j \in \{1, \cdots, C\}$
        Compute latent distribution for $set_{batch}$ ;
          // Use $\theta^j$ predict for $set_j$, $j \in \{1, \cdots, C\}$
        Compute $\mathcal{L}_{soft}$ and $\mathcal{L}_{sp}$
        Compute $\mathcal{L}_{wce}^{target}$ in $\theta^0$ ;    // use $set_{batch}$
        Update all branches $\theta^j, j \in \{0, 1, 2, \cdots, C\}$
    $e = e + 1$;
/* Testing */
Deploy model only with the target branch $\theta^0$
```

4.2. Implementation Details

By default, we use ResNet-18 as the backbone network pretrained on MS-Celeb-1M with the standard routine [38, 39] for a fair comparison. The last stage and the classifier of ResNet-18 are separated tor form auxiliary branches, while the remaining low-level layers are shared across auxiliary and target branches. The facial images are aligned and cropped with three landmarks [40], resized to 256×256 pixels, augmented by random cropping to 224×224 pixels and horizontal flipped with a probability of 0.5. During training, the batch size is 72, and each batch is constructed to ensure every class is included. We use Adam with weight decay of 10^{-4}. The initial learning rate is 10^{-3}, which is further divided by 10 at epoch 10 and 20. The training ends at epoch 40. Only the target branch is kept during testing. By default, the hyperparameters are set as $T = 1.2, \omega = 0.5, \beta = 6$ and $\gamma = 10^3$, according to the ablation studies. All experiments are carried out on a single Nvidia Tesla P40 GPU which takes 12 hours to train AffectNet with 40 epochs.

4.3. Evaluation on Synthetic Ambiguity

The annotation ambiguity in FER mainly lies in two aspects: mislabelled annotations and uncertain visual representation. We quantitatively evaluate the improvement of DMUE against the mislabelled annotations on RAF-DB and AffectNet. Specifically, a portion (*e.g*. 10%, 20% and 30%) of the training samples are randomly chosen, of which the labels are flipped to other random categories. We choose ResNet-18 as the baseline and the backbone of DMUE, and compare the performance with SCN [38], which is the state-of-the-art noise-tolerant FER method. SCN reckons uncertainty in each sample by its visual feature, and aims to find their deterministic latent truth. Each experiment is repeated three times, then the mean accuracy and standard deviation on the testing set are reported. To make fair comparison, SCN is pretrained on MS-Celeb-1M with the backbone of ResNet-18.

As shown in Table 1, the DMUE outperforms each baseline and SCN [38] consistently. With noise ratio of 30%, DMUE improves the accuracy by 4.29% and 4.21% on RAF-DB and AffectNet, respectively. This attributes to the mined latent distribution that can flexibly describe both synthetic noisy samples and compound expressions in the label space. Thus, it guides the model to overcome the harmful influence from noisy annotations.

Visualization of $\widetilde{y}$. Qualitative results are presented in the supplementary material to demonstrate that our approach can obtain the latent truth for mislabelled samples, and thereby achieve performance improvement.

4.4. Component Analysis

We conduct experiments on RAF-DB and AffectNet to analyse the contribution of latent distribution mining, uncertainty estimation and similarity preserving. As shown in Table 2, some observations can be found: **(1)** Latent distribution mining plays a more important role than others. When only one component employed, it outperforms similarity preserving and uncertainty estimation by 2.09% and 0.4% on AffectNet, 1.19% and 0.13% on RAF-DB, respectively. It proves the benefits provided by the latent distribution, as the semantic features of ambiguous images are well utilized. **(2)** When combining uncertainty estimation and latent distribution, we achieve performance improvement by 0.74% and 0.91% over only using the latent distribution on AffectNet and RAF-DB, respectively. It attributes to the uncertainty estimation module providing guidance for the target branch. Thus, the target branch can flexibly adjust the learning focus between the annotation and the latent distribution, according to the ambiguous extent of samples. **(3)** Similarity preserving also brings some improvements, while its contribution is relatively small than others. As it benefits the learning mainly by making different branches predict consistent relationships for image pairs, speeding up the training convergence. *We present more results of similarity preserving in the supplementary material.*

Table 1: Mean Accuracy and standard deviation (%) on RAF-DB and AffectNet with synthetic noisy annotations.

Method	Noisy(%)	RAF-DB	AffectNet
Baseline	10	80.43±0.72	57.21±0.31
SCN [38]	10	81.92±0.69	58.48±0.62
DMUE	10	**83.19±0.83**	**61.21±0.36**
Baseline	20	78.01±0.29	56.21±0.31
SCN [38]	20	80.02±0.32	56.98±0.28
DMUE	20	**81.02±0.69**	**59.06±0.34**
Baseline	30	75.12±0.78	52.67±0.45
SCN [38]	30	77.46±0.64	55.04±0.54
DMUE	30	**79.41±0.74**	**56.88±0.56**

Table 2: Accuracy (%) comparison of the different components. SP denotes the similarity preserving. Confidence denotes involving the uncertainty estimation module for the weighted training in target branch.

Latent distribution	SP	Confidence	AffectNet	RAF-DB
-	-	-	58.85	86.33
✓	-	-	61.76	87.84
-	✓	-	59.67	86.65
-	-	✓	61.36	87.71
✓	✓	-	62.34	88.23
-	✓	✓	61.65	87.98
✓	-	✓	62.50	88.45
✓	✓	✓	**62.84**	**88.76**

4.5. Comparison with the State-of-the-art

We compare DMUE with existing state-of-the-art methods on 4 popular in-the-wild benchmarks in Table 3.

Results. In Table 3, both CAKE [20], SCN [38] and RAN [28] utilize ResNet-18 as the backbone. SCN and RAN are pretrained on MS-Celeb-1M according to their original papers. RAN mainly deals with the occlusion and head pose problem in FER. As shown in Tabel 3, DMUE achieves current leading performance on AffectNet. For RAF-DB, all three LDL-ALSG, IPA2LT and SCN are noise-tolerant FER methods considering ambiguity, among

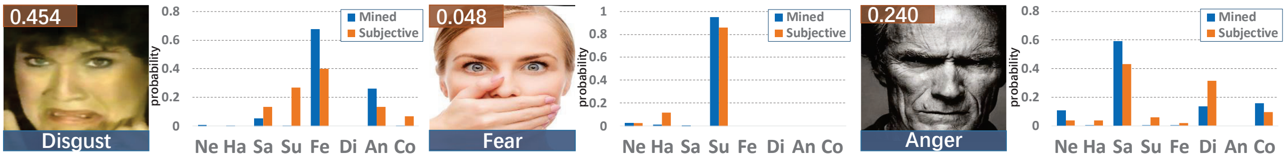


Figure 4: The mined latent distribution is compared with the subjective results. Each image is tagged with annotation and the KL-divergence between two distributions. The generated latent distribution is consistent with intuition. Best viewed in color. Zoom in for better view. (Ne=Neutral,Ha=Happy,Sa=Sad,Su=Surprise,Fe=Fear,Di=Disgust,An=Anger,Co=Contempt).

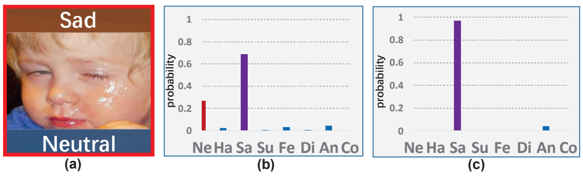


Figure 5: (a) A *Sad* training image mislabelled with *Neutral* in AffectNet. (b) The prediction from model trained on AffectNet can show the right class. (c) The prediction from model trained on all classes except for the *Neutral* better reflects the truth discriminatively. Best viewed in color. Zoom in for better view.

which SCN achieves state-of-the-art results. We further improve the performance of ambiguous FER by mining latent distribution and considering annotations in uncertainty estimation. Table 3 also shows the results on FERPlus and SFEW, respectively. Without bells and whistles, our method achieves better performance than the counterparts.

4.6. Visualization Analysis

To further diagnose our method, we conduct visualizations of the discovered latent distribution and the estimated confidence score.

Latent Distribution. In Section 4.3, we quantitatively demonstrate the effectiveness of DUME to deal with mislabelled images. In this section, we further conduct user study for qualitative analysis of how latent distribution cope with uncertain expressions. Specifically, 20 images are randomly picked from RAF-DB and AffectNet, and labelled by 50 voters. As latent distribution reflects the sample's probability distribution among its negative classes, we set the number of votes on sample's positive class to be zero. The normalized subjective results are compared with the mined latent distribution.

In Fig. 4, KL-divergence between the subjective result and the latent distribution is reported for reference. It is interesting to see people have different views of the specific type of expressions. Furthermore, our approach obtains qualitatively consistent results with human intuition. Although there exists differences in details, it is worth noting that the results can already qualitatively explain that the latent distribution benefits the model by reinforcing the supervision information.

Confidence Score. To corporate with latent distribution mining, a confidence score is estimated by the uncertainty estimation module for each image given a batch. The more ambiguous a sample is, the lower its confidence score will be. Thus, the target branch will learn more from its latent distribution. We qualitatively analyse the uncertainty estimation module by visualizing images with the original annotation and the scaled confidence score. Moreover, we rank images by their confidence scores and report their ranks in a batch of 72 images.

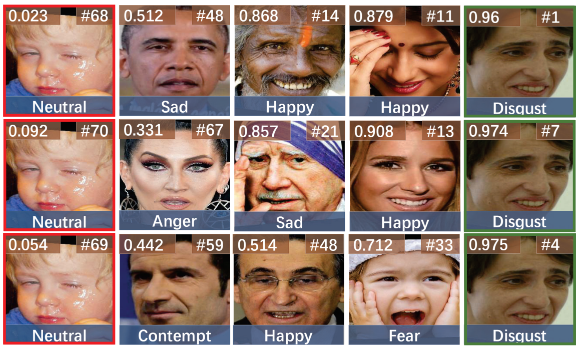


Figure 6: From top to bottom: images from three different batches with their annotations. Red (Green) bounding box denotes bad (good) anchor image. The upper left (right) corner of each picture is tagged with its confidence score (rank) in the batch. The estimated score is robust and consistent with intuition. Best viewed in color. Zoom in for better view.

In Fig. 6, we choose two typical anchor images and report their results in three different batches. The confident samples are assigned with higher score, while the ambiguous ones are the opposite. Furthermore, both the scores and ranks of anchor images are consistent within three different batches. It shows the robustness of our pairwise uncertainty estimation module. *More analyses are provided in the supplementary material.*

4.7. Ablation Study

We conduct extensive ablation studies on AffectNet, as it is the largest dataset. *Some of them are provided in the supplementary material, due to the page limitation.*

Mining latent distribution. Quantitative and qualitative experiments on AffectNet are conducted to analyze the way of mining latent distribution. For the former, given a batch,

Table 3: Comparison with the state-of-the-art results. Res denotes ResNet. $^+$ denotes both AffectNet and RAF-DB are used as the training set. * means using extra distribution data instead of single category annotation. † denotes the method is trained and tested with 7 classes on AffectNet.

(a) Comparison on AffectNet

Method	Acc.
Upsample [28]	47.01
IPA2LT$^+$ [49]	55.71
RAN [28]	59.50
CAKE† [20]	61.70
SCN [38]	60.23
Ours(Res-18)	**62.84**
Ours(Res-50IBN)	**63.11**

(b) Comparison on RAF-DB

Method	Acc.
gaCNN [25]	85.07
LDL-ALSG$^+$ [11]	85.53
IPA2LT$^+$ [49]	86.77
SCN [38]	87.03
SCN$^+$ [38]	88.14
Ours(Res-18)	**88.76**
Ours(Res-50IBN)	**89.42**

(c) Comparison on FERPlus

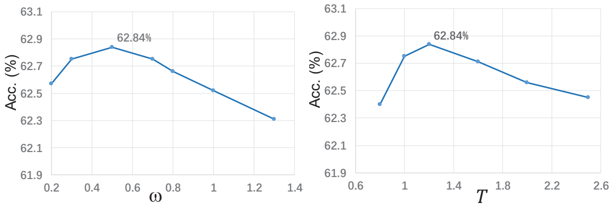
Method	Acc.
PLD* [4]	85.10
Res+VGG [18]	87.40
SCN	88.01
SeNet50* [2]	88.80
RAN [39]	88.55
Ours(Res-18)	**88.64**
Ours(Res-50IBN)	**89.51**

(d) Comparison on SFEW

Method	Acc.
IdentityCNN [27]	50.98
Island loss [8]	52.52
Incept-ResV1 [1]	51.90
MultiCNNs [48]	55.96
RAN [39]	56.40
Ours(Res-18)	**57.12**
Ours(Res-50IBN)	**58.34**

Table 4: Ablation study of ways to mine latent distribution.

Methods	Baseline	LD-A	LD-N
Acc. (%)	58.85	60.03	61.32



(a)

(b)

Figure 7: (a) The accuracy (%) with different ω. (b) The accuracy (%) with different T.

we train each auxiliary branch with all the samples, where the $(C-1)$-class classifier is switched to C-class. To make the latent distribution, their predictions are averaged to increase the robustness. For simplification, we denote latent distribution mined in this way as LD-A, while the original in DUME as LD-N.

As shown in Table 4, LD-N guides the target branch better. Because it can reflect more discriminative latent truth. *More analyses are provided in supplementary material.*

Trade-off Weight ω. ω balances the learning of target branch between $\widetilde{\boldsymbol{y}}_x$ and annotation. Fig. 7 shows that too small ω causes trouble for target branch to learn $\widetilde{\boldsymbol{y}}_x$. When ω is too large, it is hard for uncertain estimation module to adjust learning focus, as the sensitivity to $\widetilde{\boldsymbol{y}}_x$ is enlarged.

Sharpen Temperature T. T provides the flexibility to slightly modify the entropy of $\widetilde{\boldsymbol{y}}_x$. Fig. 7 shows the effect with different T. When $T < 1$, the distribution becomes steep quickly, damaging the fine-grained label information. Using $T > 1$ flattens $\widetilde{\boldsymbol{y}}_x$, relieving model's sensitivity to incorrect predictions. Yet, the performance will be degraded if T is too large, as the pattern of $\widetilde{\boldsymbol{y}}_x$ is suppressed.

Epoch Threshold β. The first β-th epoch is dedicated to pretraining the auxiliary branches in prior, to make them provide stable latent distribution. After the β-th epoch, attention is paid more on optimizing the target branch. Table 5 shows the accuracy with different β.

Table 5: The accuracy (%) with different β.

β	2	3	6	10	14
Acc. (%)	62.28	62.54	62.84	62.50	62.41

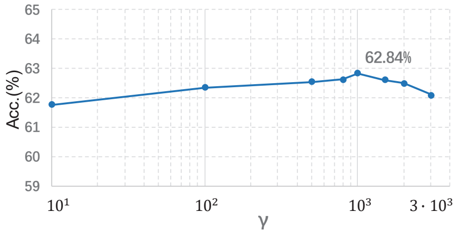


Figure 8: Accuracy(%) sensitivity to γ.

Similarity Preserving factor γ. We generalized the similarity preserving to the context of multi-branch architecture. γ adjusts the contribution ratio of the mechanism. Fig. 8 reflects the performance of model with different γ.

5. Conclusion

In order to address the ambiguity problem in FER, we propose DMUE, with the design of latent distribution mining and pairwise uncertainty estimation. On one hand, the mined latent distribution describes the ambiguous instance in a fine-grained way to guide the model. On the other hand, pairwise relationships between samples are fully exploited to estimate the ambiguity degree. Our framework imposes no extra burden on inference, and can be flexibly integrated with the existing network architectures. Experiments on popular benchmarks and synthetic ambiguous datasets show the effectiveness of DMUE.

Acknowledgements

This work is supported by the National Key R&D Program of China under Grant No. 2020AAA0103800 and by the National Natural Science Foundation of China under Grant U1936202 and 62071216. This work is mainly done at JD AI Research.

References

[1] Dinesh Acharya, Zhiwu Huang, Danda Pani Paudel, and Luc Van Gool. Covariance pooling for facial expression recognition. In *CVPRW*, 2018. 8

[2] Samuel Albanie, Arsha Nagrani, Andrea Vedaldi, and Andrew Zisserman. Emotion recognition in speech using cross-modal transfer in the wild. In *ACM MM*, 2018. 2, 8

[3] Görkem Algan and Ilkay Ulusoy. Image classification with deep learning in the presence of noisy labels: A survey. *KBS*, 2021. 2

[4] Emad Barsoum, Cha Zhang, Cristian Canton-Ferrer, and Zhengyou Zhang. Training deep networks for facial expression recognition with crowd-sourced label distribution. In *ICMI*, 2016. 5, 8

[5] Juliano J. Bazzo and Marcus V. Lamar. Recognizing facial actions using gabor wavelets with neutral face average difference. In *FG*, 2004. 2

[6] Carlos Fabian Benitez-Quiroz, Ramprakash Srinivasan, and Aleix M. Martínez. Emotionet: An accurate, real-time algorithm for the automatic annotation of a million facial expressions in the wild. In *CVPR*, 2016. 1

[7] David Berthelot, Nicholas Carlini, Ian J. Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *NeurIPS*, 2019. 2, 3

[8] Jie Cai, Zibo Meng, Ahmed-Shehab Khan, Zhiyuan Li, James O'Reilly, and Yan Tong. Island loss for learning discriminative features in facial expression recognition. In *FG*, 2018. 8

[9] Jie Chang, Zhonghao Lan, Changmao Cheng, and Yichen Wei. Data uncertainty learning in face recognition. In *CVPR*, 2020. 2

[10] Defang Chen, Jian-Ping Mei, Can Wang, Yan Feng, and Chun Chen. Online knowledge distillation with diverse peers. In *AAAI*, 2020. 3

[11] Shikai Chen, Jianfeng Wang, Yuedong Chen, Zhongchao Shi, Xin Geng, and Yong Rui. Label distribution learning on auxiliary label space graphs for facial expression recognition. In *CVPR*, 2020. 1, 2, 8

[12] Abhinav Dhall, Roland Goecke, Simon Lucey, and Tom Gedeon. Static facial expression analysis in tough conditions: Data, evaluation protocol and benchmark. In *ICCV*, 2011. 5

[13] Jacob Goldberger and Ehud Ben-Reuven. Training deep neural-networks using a noise adaptation layer. In *ICLR*, 2017. 2

[14] I. J. Goodfellow, D. Erhan, P. L. Carrier, A. Courville, M. Mirza, B. Hamner, W. Cukierski, Y. Tang, D. Thaler, and et al. Dong-Hyun Lee. Challenges in representation learning: A report on three machine learning contests. In *ICONIP*, 2013. 5

[15] Jiangfan Han, Ping Luo, and Xiaogang Wang. Deep self-learning from noisy labels. In *ICCV*, 2019. 2

[16] Wei Hu, Yangyu Huang, Fan Zhang, and Ruirui Li. Noise-tolerant paradigm for training face recognition cnns. In *CVPR*, 2019. 5

[17] Yibo Hu, Xiang Wu, and Ran He. TF-NAS: rethinking three search freedoms of latency-constrained differentiable neural architecture search. In *ECCV*, 2020. 2

[18] Christina Huang. Combining convolutional neural networks for emotion recognition. *2017 IEEE MIT URT*, 2017. 5, 8

[19] Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *ICML*, 2018. 2

[20] Corentin Kervadec, Valentin Vielzeuf, Stéphane Pateux, Alexis Lechervy, and Frédéric Jurie. CAKE: a compact and accurate k-dimensional representation of emotion. In *BMVC*, 2018. 6, 8

[21] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *ICLR*, 2017. 5

[22] Xu Lan, Xiatian Zhu, and Shaogang Gong. Knowledge distillation by on-the-fly native ensemble. In *NeurIPS*, 2018. 3

[23] Junnan Li, Richard Socher, and Steven C. H. Hoi. Dividemix: Learning with noisy labels as semi-supervised learning. In *ICLR*, 2020. 2, 3

[24] Shan Li, Weihong Deng, and Junping Du. Reliable crowdsourcing and deep locality-preserving learning for expression recognition in the wild. In *CVPR*, 2017. 1, 2, 5

[25] Yong Li, Jiabei Zeng, Shiguang Shan, and Xilin Chen. Occlusion aware facial expression recognition using CNN with attention mechanism. *TIP*, 2019. 2, 8

[26] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. Sphereface: Deep hypersphere embedding for face recognition. In *CVPR*, 2017. 5

[27] Debin Meng, Xiaojiang Peng, Kai Wang, and Yu Qiao. Frame attention networks for facial expression recognition in videos. In *ICIP*, 2019. 8

[28] Ali Mollahosseini, Behzad Hassani, and Mohammad H. Mahoor. Affectnet: A database for facial expression, valence, and arousal computing in the wild. *TAC*, 2019. 1, 2, 5, 6, 8

[29] Pauline C. Ng and Steven Henikoff. SIFT: predicting amino acid changes that affect protein function. *Nucleic Acids Res.*, 2003. 2

[30] Curtis G. Northcutt, Lu Jiang, and Isaac L. Chuang. Confident learning: Estimating uncertainty in dataset labels. *arXiv*, 1911.00068, 2019. 2

[31] Caifeng Shan, Shaogang Gong, and Peter W. McOwan. Facial expression recognition based on local binary patterns: A comprehensive study. *IVC.*, 2009. 2

[32] Yichun Shi and Anil K. Jain. Probabilistic face embeddings. In *ICCV*, 2019. 2

[33] Jun Shu, Qi Xie, Lixuan Yi, Qian Zhao, Sanping Zhou, Zongben Xu, and Deyu Meng. Meta-weight-net: Learning an explicit mapping for sample weighting. In *NeurIPS*, 2019. 2

[34] Yansong Tang, Zanlin Ni, Jiahuan Zhou, Danyang Zhang, Jiwen Lu, Ying Wu, and Jie Zhou. Uncertainty-aware score distribution learning for action quality assessment. In *CVPR*, 2020. 2

[35] Frederick Tung and Greg Mori. Similarity-preserving knowledge distillation. In *ICCV*, 2019. 4

[36] Valentin Vielzeuf, Alexis Lechervy, Stéphane Pateux, and Frédéric Jurie. Towards a general model of knowledge for facial analysis by multi-source transfer learning. In *WACV*, 2019. 2

[37] Can Wang, Shangfei Wang, and Guang Liang. Identity- and pose-robust facial expression recognition through adversarial feature learning. In *ACM MM*, 2019. 2

[38] Kai Wang, Xiaojiang Peng, Jianfei Yang, Shijian Lu, and Yu Qiao. Suppressing uncertainties for large-scale facial expression recognition. In *CVPR*, 2020. 1, 2, 5, 6, 8

[39] Kai Wang, Xiaojiang Peng, Jianfei Yang, Debin Meng, and Yu Qiao. Region attention networks for pose and occlusion robust facial expression recognition. *TIP*, 2020. 5, 8

[40] Xinyao Wang, Liefeng Bo, and Fuxin Li. Adaptive wing loss for robust face alignment via heatmap regression. In *ICCV*, 2019. 5

[41] Yisen Wang, Xingjun Ma, Zaiyi Chen, Yuan Luo, Jinfeng Yi, and James Bailey. Symmetric cross entropy for robust learning with noisy labels. In *ICCV*, 2019. 2

[42] Guile Wu and Shaogang Gong. Peer collaborative learning for online knowledge distillation. *arXiv*, 2006.04147, 2020. 3

[43] Xiang Wu, Ran He, Yibo Hu, and Zhenan Sun. Learning an evolutionary embedding via massive knowledge distillation. *IJCV*, 2020. 2

[44] N. Xu, Y. P. Liu, and X. Geng. Label enhancement for label distribution learning. *TKDE*, 2021. 2

[45] Ning Xu, Jun Shu, Yun-Peng Liu, and Xin Geng. Variational label enhancement. In *ICML*, 2020. 2

[46] Huiyuan Yang, Umur A. Ciftci, and Lijun Yin. Facial expression recognition by de-expression residue learning. In *CVPR*, 2018. 2

[47] Kun Yi and Jianxin Wu. Probabilistic end-to-end noise correction for learning with noisy labels. In *CVPR*, 2019. 2

[48] Zhiding Yu and Cha Zhang. Image based static facial expression recognition with multiple deep network learning. In *ACM ICMI*, 2015. 8

[49] Jiabei Zeng, Shiguang Shan, and Xilin Chen. Facial expression recognition with inconsistently annotated datasets. In *ECCV*, 2018. 1, 2, 8