Relative Uncertainty Learning for Facial Expression Recognition

Anonymous Author(s)
Affiliation
Address
email

Abstract

In facial expression recognition (FER), the uncertainties introduced by inherent noises like ambiguous facial expressions and inconsistent labels raise concerns about the credibility of recognition results. To quantify these uncertainties and achieve good performance under noisy data, we regard uncertainty as a relative concept and propose an innovative uncertainty learning method called Relative Uncertainty Learning (RUL). Rather than assuming Gaussian uncertainty distributions for all datasets, RUL builds an extra branch to learn uncertainty from the relative difficulty of real samples by feature mixing. Specifically, we use uncertainties as weights to mix facial features and design an add-up loss to encourage uncertainty learning. It is easy to implement and adds little or no extra computation overhead. Extensive experiments show that RUL outperforms state-of-the-art FER uncertainty learning methods in both real-world and synthetic noisy FER datasets. Besides, RUL also works well on other datasets such as MNIST and CIFAR.

1 Introduction

Although well-trained models can give powerful inferences directly, the inference results may be actually wrong and cause serious consequences [22]. Therefore, the concept of uncertainty is widely used in measuring how well the model trusts its inference results, which has become a research hotspot in reliable machine learning [3, 11, 10, 22, 26, 17]. It is expected that a reliable model will assign a high level of uncertainty to its erroneous predictions, so that human can intervene in decision-making to avoid lots of disasters. As described by Kiureghian *et al.* [25], the uncertainties in machine learning can be split into aleatoric uncertainty and epistemic uncertainty which are also called as data uncertainty and model uncertainty. In this paper, we focus on quantifying data uncertainty as model uncertainty can be eliminated by introducing more training data. Concretely, data uncertainty in facial expression recognition (FER) is mainly caused by ambiguous facial expressions and the subjectiveness of annotators, which blocks the improvement of recognition performance [45].

Recent studies propose a variety of methods to solve the uncertainty learning problem in face recognition field. PFE [36] introduces a new branch to learn the uncertainty of face recognition and recognizes faces by measuring the similarity between two Gaussian distributions. Chang *et al.* [4] further propose DUL to learn feature and uncertainty simultaneously in order to improve facial feature learning. When it comes to FER, SCN [45] uses a fully-connected layer to learn an importance weight for each image and suppress uncertainties according to the learned weights. However, the strong learning ability of neural networks will deteriorate uncertainty learning branch. One solution is adding regularization to force the model learn uncertainty, which needs to be tuned carefully, otherwise it would lead to underfit. For example, SCN uses margin loss to keep the gap between certain and uncertain images, but choosing a suitable gap value is time-consuming. DUL regularizes features to follow a standard normal distribution, but the real distribution of dataset is not always

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

following such a distribution. Therefore, it is still a challenge to learn uncertainty distributions of real datasets without deteriorating feature learning.

To address the problem, we perceive data uncertainty from a different point of view. We realize that data uncertainty – the difficulty to classify a sample correctly is actually a relative concept. More specifically, the difficulty of classification is a person's subjective feeling comes from comparison. One can only know whether a sample is easy or hard according to a reference. Inspired by this idea, we propose an innovative method called Relative Uncertainty Learning (RUL) to help deep learning models learn uncertainty for each sample. Concretely, we build a new branch to model the uncertainties of input images and utilize the uncertainties as weights to mix two features of different labels. Through an add-up loss function, the model is encouraged to recognize two expressions simultaneously from the mixed features, which enables the model to learn uncertainty comparatively along with minimizing the total loss. As discussed in Section 3.3, the uncertainty learning branch will assign large uncertainty values to uncertain facial expression images while small uncertainty values to certain images.

The main contributions of this work are as follows:

- We perceive uncertainty from a new perspective and propose an innovative uncertainty quantification method to learn uncertainty from the relative difficulty of two samples.
- We get state-of-the-art performance in both real-world and synthetic noisy FER datasets.
- RUL doesn't need any prior knowledge of the dataset uncertainty distribution and can be easily applied into different classification tasks with low computation cost.

2 Related Work

Uncertainty learning in face field Uncertainties can be mainly categorised into model uncertainty which accounts for uncertainty in the model parameters and data uncertainty which is the noise inherent in the dataset [22]. In face recognition field, uncertainty learning is important for learning discriminative feature embeddings of noisy face images. Because noisy face images are usually out of the cluster and have larger variances in the latent embedding space which might cause wrong recognition [36]. There have been several proposals to solve the uncertainty learning problem in face recognition tasks [14, 49, 23, 36, 4]. Some of them [14, 49, 23] have tried to analyze and learn face representations using model uncertainty. Gong *et al.* [14] estimate the capacity of a given face representation by explicitly accounting for the manifold structure, model uncertainty and data variability. Khan *et al.* [23] present a framework for class imbalance learning based on model uncertainty. Zafar *et al.* [49] cope with false positives through employing model uncertainty to improve the efficacy of face recognition systems. Thereinto the methods considering data uncertainty learning. Shi *et al.* [36] use PFE to learn a variance for each fixed feature and then measure the likelihood of each positive face pair of (x_i, x_j) sharing the same distribution of latent embedding p(z_i, z_j). Chang *et al.* [4] further propose to learn feature and uncertainty simultaneously by encoding the latent feature embedding as a Gaussian distribution. In FER task, Wang *et al.* [45] try to learn an attention weight for each image, then utilize it to weight cross-entropy loss in order to suppress the influence of noisy samples.

Noisy dataset training With the enhancement of deep learning network capabilities, the datasets for facial expression recognition is getting larger and larger. For the large-scale FER datasets [42, 52, 7, 1, 33, 9, 27], due to the ambiguous of expressions and the subjectiveness of human annotators, it is extremely hard to get high quality labels. Getting high performance training with label noisy datasets has been studied for a long time. The basic idea is to enable the model access the quality of the labels or estimate the noise distribution using a small set of clean dataset [5, 28, 38, 43]. Li *et al.* [28] design a distillation framework to use information from both a small clean dataset and label relations in knowledge graphs to prevent the model learning from noisy labels. Veit *et al.* [43] use small amount of clean annotations to reduce the noisy samples before fine-tuning the network using both the clean set and the full set. Other methods don't use a small clean dataset but they usually assume extra constrains on the noisy samples. Natarajan *et al.* [34] design a specific loss function for randomly flipped labels. Goldberger *et al.* [13] explicitly model the noises in datasets by a softmax layer which connects the correct labels to the noisy ones. Without collecting a clean set of dataset or

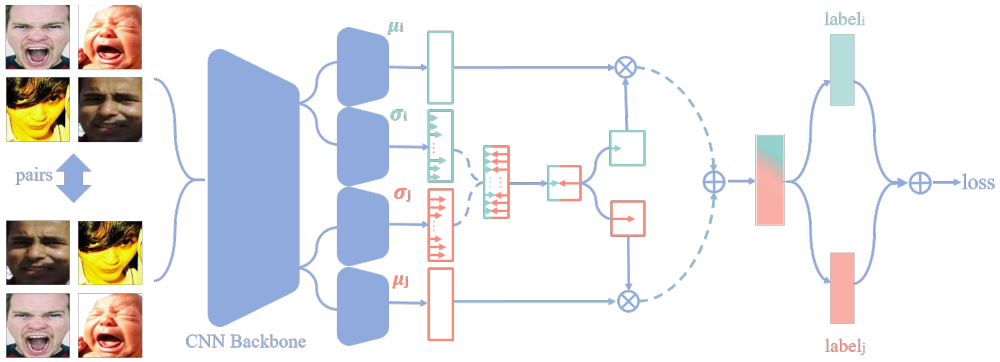

Figure 1: Overview of our Relative Uncertainty Learning method. We use two branches to learn facial features μ and uncertainty vectors σ simultaneously. Then we normalize uncertainty vectors with their counterparts sampled from shuffled indexes. Next we mix the facial features according to the mean uncertainty values. *Note that the easy sample will get a small uncertainty value because its facial feature is very obvious for recognition, while the ambiguous sample tends to get a large uncertainty value in order to make it easy for the model to recognize its corresponding label from the mixed features.* Finally, we use the add-up loss to encourage the model to recognize the two expressions from the mixed feature which implicitly force the model to learn reliable uncertainty values.

using prior knowledge of noise distribution, our method learns uncertainty through relativity of the given dataset and suppress the most uncertain samples to get good performance under label noise.

3 Proposed Method

In order to reduce the negative impact of the strong learning ability of modern neural networks and learn uncertainty values closer to the real dataset distribution, we propose a simple and effective method named Relative Uncertainty Learning (RUL). In this section, we first provide the motivation behind our method. Then we demonstrate the details of RUL. Finally, we explain why RUL is effective in uncertainty learning.

3.1 Motivation

The uncertainty learning methods mentioned before suffer from the strong learning ability of deep neural networks more or less. For example, the introduced variances in DUL are equivalent to some noises being added to feature points, so the model will learn very small variances for all feature points, which makes the estimation of uncertainty meaningless. SCN also learns large importance weights for noisy samples to obtain small cross-entropy loss values as these noisy images will be memorized by the networks after training for several epochs.

In order to prevent the strong learning ability of neural network from degrading the uncertainty learning branch, we perceive uncertainty from a relative point of view. We find that learning uncertainty through comparison forms a natural regularization because if the model assigns a very small uncertainty value for one image, then it means the other image is harder to recognize and that image will get a large uncertainty value which prevents the model learning small uncertainty values for all images. Learning uncertainty through relativity also conforms to human cognition – it is very difficult for a person to define which sample is certain and which sample is uncertain without using one image as reference. In the following section, we thoroughly illustrate the details of our proposed RUL.

3.2 Overview of Relative Uncertainty Learning

In this section, we illustrate the structure of our proposed RUL. The pipeline of RUL is shown in Figure 1.

Two Branches Model Given a batch of face images $\boldsymbol{X}$, assuming the conventional feature extraction module $f_\theta(\boldsymbol{X})$ can be decomposed as

$$f_\theta(\boldsymbol{X}) = f_\theta^{(l)}(f_\theta^{(l-1)}(\boldsymbol{X})), \tag{1}$$

we first extract general facial features $\boldsymbol{F}_g$ from the penultimate feature extraction layer denoted as $\boldsymbol{F}_g = f_\theta^{(l-1)}(\boldsymbol{X})$ [48]. Then, in order to learn facial features and uncertainty values simultaneously, we drop the layer indexed by l, i.e., $f_\theta^{(l)}$ and build two separate branches $f_{\theta_\mu}(f_\theta^{(l-1)}(\boldsymbol{X}))$ and $f_{\theta_\sigma}(f_\theta^{(l-1)}(\boldsymbol{X}))$ to produce facial features $\boldsymbol{F} = [\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, ..., \boldsymbol{\mu}_N] \in \mathbb{R}^{D\times N}$ and uncertainty vectors $\boldsymbol{U} = [\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2, ..., \boldsymbol{\sigma}_N] \in \mathbb{R}^{D\times N}$. D, N denote output dimension and batch size separately.

Relative Uncertainty Learning Inspired by the relativity of the uncertainty concept and the mixup method [51, 41, 44], we mix two different facial features according to their uncertainty values which enables FER model to learn uncertainty through the relativity of different samples. First, we mix a mini-batch with their shuffled ones which results in the facial images of the whole dataset being compared with each other during training process because of the randomness introduced by dataloader and the shuffle of mini-batches. Before mixing features, we first normalize the uncertainty vectors at element-wise to compare two features with each other following Equation (2)

$$\hat{\boldsymbol{\sigma}}_i, \hat{\boldsymbol{\sigma}}_j = \frac{\boldsymbol{\sigma}_i}{\boldsymbol{\sigma}_i + \boldsymbol{\sigma}_j}, \frac{\boldsymbol{\sigma}_j}{\boldsymbol{\sigma}_i + \boldsymbol{\sigma}_j}, \tag{2}$$

$\boldsymbol{\sigma}_i$ and $\boldsymbol{\sigma}_j$ are the uncertainty vectors for image i and its counterpart image j sampled from shuffled indexes, note that they have different labels. This operation not only forms a natural regularization as the model can't learn small uncertainty values for both features, but also benefits the training process as extreme uncertainty values might cause the training process unstable or even the loss can't converge. As mentioned by DUL [4], mean of the predicted uncertainty vectors $\hat{\boldsymbol{\sigma}}_i \in \mathbb{R}^D$ can be viewed as an approximated measurement of the estimated uncertainty for image i. So we use the mean of $\hat{\boldsymbol{\sigma}}_i$ as learned uncertainty value for image i denoted as $\hat{\sigma}_{i_{mean}} \in \mathbb{R}$. Then we get an uncertainty value for each input image denoted as $\boldsymbol{U} = [\hat{\sigma}_{1_{mean}}, \hat{\sigma}_{2_{mean}}, ..., \hat{\sigma}_{N_{mean}}] \in \mathbb{R}^{1\times N}$. We then mix features according to their uncertainty values following Equation (3)

$$\tilde{\boldsymbol{\mu}} = \hat{\sigma}_{i_{mean}}\boldsymbol{\mu}_i + \hat{\sigma}_{j_{mean}}\boldsymbol{\mu}_j, \tag{3}$$

$\boldsymbol{\mu}_i$ and $\boldsymbol{\mu}_j$ represent embedding features of the two images, $\hat{\sigma}_{i_{mean}}$ and $\hat{\sigma}_{j_{mean}}$ represent their corresponding uncertainty values. We introduce negligible computation overhead as we only mix facial features according to their learned uncertainty values in training phase. Furthermore, the mix feature part will be removed for deployment. Therefore, our network is end-to-end and adds no extra cost on inference.

Classification Loss We design an add-up loss to encourage the uncertainty learning branch to learn different uncertainty values for different facial images. Specifically, add-up loss requires the FER model to recognize two expressions of input images simultaneously from mixed features. Since $\tilde{\boldsymbol{\mu}}$ is the mixed feature, we feed $\tilde{\boldsymbol{\mu}}$ to the classifier:

$$L_{total} = -\frac{1}{N}\sum_{i,j}^{N}(\log\frac{e^{\boldsymbol{W}_{y_i}\tilde{\boldsymbol{\mu}}}}{\sum_c^C e^{\boldsymbol{W}_c\tilde{\boldsymbol{\mu}}}} + \log\frac{e^{\boldsymbol{W}_{y_j}\tilde{\boldsymbol{\mu}}}}{\sum_c^C e^{\boldsymbol{W}_c\tilde{\boldsymbol{\mu}}}}). \tag{4}$$

The loss function adds up the loss of recognizing $\tilde{\boldsymbol{\mu}}$ as i class and as j class. y_i, y_i means $label_i$, $label_j$, $\boldsymbol{W}_c$ is the c-th classifier and C means the total number of expression classes. The loss function forces the model to recognize two expressions equally from the mixed features. When mixing two facial features, there will be a relatively easy facial feature for expression recognition and the other is relatively hard. For simplicity, we note $\boldsymbol{\mu}_i$ as the easy facial feature. Our model will learn a small uncertainty value for $\boldsymbol{\mu}_i$ to mix with $\boldsymbol{\mu}_j$ as $\boldsymbol{\mu}_i$ is easy to recognize, the model can already get small classification loss with its label y_i even $\boldsymbol{\mu}_i$ only takes up a small amount of the mixed feature $\tilde{\boldsymbol{\mu}}$. While for the hard image j, it needs to take up a large amount of the mixed feature $\tilde{\boldsymbol{\mu}}$ to enable the model find useful feature to get a small classification loss with label y_j.

3.3 Why RUL is Effective

In this section, we illustrate why RUL is effective in uncertainty learning. For visualization simplicity, we plot the mixed images instead of mixed features in Figure 2. At the start of training, RUL might

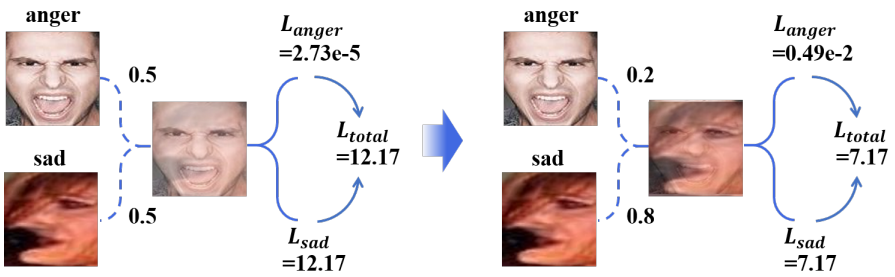

Figure 2: An experiment to show why RUL is effective in uncertainty learning. We mix different features with different weights and print the losses of recognizing the mixed feature to different labels. Visually, the face with anger expression is the certain one, and the face with sad expression is the uncertain one. In the process of feature mixup, RUL will assign a larger weight to the uncertain one to minimize the total loss.

learn similar uncertainty values for all images, which is shown in the left part of Figure 2. As the certain facial expression image with anger label in Figure 2 can be easily recognized from the mixed feature, the model can already get a small loss value with anger label while gets a large loss value with sad label. In order to get minimum total loss, the model needs to recognize the sad expression from the mixed feature more effectively. Thus, as training progress, the mix weight (uncertainty value) of uncertain sad image will increase to enable the model recognize the sad expression from the mixed feature. Thus, RUL will learn larger uncertainty values for uncertain images.

4 Experiments

In this section, we first describe two in-the-wild FER datasets and our implementation details. We then evaluate our proposed method on these datasets. We compare our method with state-of-the-art uncertainty learning methods both on accuracy, noisy label training and accuracy with rejection [40]. Furthermore, we provide qualitative and quantitative analysis to show what is the meaning of our learned uncertainty values. Last, we compare our method to other state-of-the-art FER methods and also demonstrate that RUL is applicable to datasets in different fields. We also carry out ablation study to find the effectiveness of mix part in RUL.

4.1 Datasets

To evaluate the effectiveness of our model, we select two widely used in-the-wild FER datasets. Compared with FER datasets collected in the laboratory, they have more noises like noisy labels and ambiguous expressions. So they are more complex, variable and challenging for expression recognition tasks. The details are as follows.

RAF-DB [27] is a crowdsourced facial expression dataset which contains 29672 facial images annotated with basic or compound expressions by 40 trained annotators. Annotators annotate basic emotion images with one label and compound emotion images with two labels. In this work, we use facial images with 6 basic expressions [8] (surprise, anger, sad, happy, fear, disgust) and neutral expression including 12271 images as training data and 3068 images as test data.

FER2013 [15] consists of 35,887 grayscale 48x48 pixel images in total, with 28,709 training samples, 3,589 public test samples and 3,589 private test samples. These facial images are classified into 7 emotion classes same as RAF-DB.

4.2 Implementation Details

Given a batch of images from RAF-DB, we first resize them to 224×224 pixels and use ResNet-18 [19] which is pretrained on Ms-Celeb-1M [18] as backbone. The two branches have the same structure BatchNorm2d [20]-Dropout [37]-Flatten-FC-BatchNorm1d. We set dropout rate as 0.4. Note that the second branch outputs $\log \boldsymbol{\sigma}$ instead of $\boldsymbol{\sigma}$ in order to stabilize the training following [4]. The model is trained in an end-to-end manner with a single GTX 1080ti GPU for 70 epochs with

batch size of 64. We also utilize an Adam optimizer [24] with weight decay of 0.0001. The learning rate is initialized as 0.0006 except the last fully connected layer for classification which is 0.001. We use ExponentialLR [30] learning rate scheduler with gamma of 0.9 to decrease the learning rate after each epoch. As for FER2013, we use the same setting as [47].

4.3 Evaluation on Label Noises

Both ambiguous facial expressions and inconsistent annotations will cause label noises in FER datasets which are one of the main sources of FER dataset noises. We carry out extensive experiments to demonstrate the improvement of RUL compared with state-of-the-art noise-tolerant FER methods. Following SCN [45], we randomly choose 10%, 20%, 30% of training data and flip their labels to other expression categories randomly. To make a fair comparison with SCN, we also considers the most uncertain samples in training process as samples with label noises and if the maximum prediction probability is higher than the one of given label with a threshold (set to 0.2 in all cases), we choose to rely on the FER model and change the label to the index of the maximum prediction probability.

We run all experiments three times and compute mean and standard variance of the results. As shown in Table 1, RUL outperforms SCN and DUL under all circumstances. RUL improves test accuracy by 1.14%, 2.39%, 3.16% with noise ratio of 10%, 20% and 30% on RAF-DB and 0.86%, 2.31%, 3.64% on FER2013 Private testset compared with DUL. Experiment results show that the benefit from RUL becomes more obvious as the noise ratio increases up which means RUL are more robust to label noises. The results also demonstrate that RUL can learn meaningful uncertainty values as we use uncertainty values to guide the model find the latent truth of training samples and suppress label noises.

Table 1: Test accuracy (%) on RAF-DB and FER2013 with synthetic noisy labels.

Method	Noisy (%)	RAF-DB	FER2013 PublicTest	FER2013 PrivateTest
Baseline	10	80.43±0.72	67.27±0.25	69.25±0.18
SCN	10	81.92±0.69	67.67±0.05	69.28±0.05
DUL	10	85.08±0.21	67.68±0.13	69.43±0.27
RUL (Ours)	10	**86.22±0.29**	**68.45±0.14**	**70.29±0.28**
Baseline	20	78.01±0.29	64.16±0.13	64.87±0.32
SCN	20	80.02±0.32	65.75±0.62	66.30±0.49
DUL	20	81.95±0.32	64.24±0.06	65.55±0.31
RUL (Ours)	20	**84.34±0.29**	**66.64±0.21**	**67.86±0.35**
Baseline	30	75.12±0.78	61.26±0.36	62.52±0.56
SCN	30	77.46±0.64	61.50±0.39	62.61±0.82
DUL	30	78.90±0.80	59.78±0.31	60.98±0.34
RUL (Ours)	30	**82.06±0.44**	**63.16±0.07**	**64.62±0.39**

4.4 Test with Rejection.

In order to illustrate the quality of our learned uncertainty values more intuitively, we utilize an evaluation metric similar to [40, 2, 16, 39] named accuracy versus reject curve. The metric shows a test accuracy over the fraction of unconsidered facial expression images. Specifically, based on the predicted uncertainty values, these unconsidered images are the most uncertain images to recognize and the test accuracy is calculated on the remaining images. Accuracy versus reject curve indicates good uncertainty estimation when the test accuracy increases consistently when increasing the ratio of unconsidered images. We also compare with out-of-distribution detection method [6] noted as CONF. CONF uses labels as hints to help indicate when the network tends to give wrong predictions. The results on different datasets are shown in Figure 3 and Figure 4. RUL gets the best performance in all cases which illustrates the uncertainty values learned by RUL are more related with recognition confidence than other methods. Another interesting find is that CONF performs pretty well on RAF-DB while degrades on FER2013. It may be explained by the following reason. We use pretrained model on RAF-DB while training from scratch on FER2013. CONF needs strong backbone to make good use of the information from labels while our method can perform well in both ways. We also show the precise test accuracies in Table 2.

Table 2: Test accuracy (%) versus rejection rate results of different uncertainty learning methods on RAF-DB and FER2013. The best performance among each column is shown in bold form.

Method	RAF-DB				FER2013 Private Testset			
	0 rejection	10% rejection	20% rejection	30% rejection	0 rejection	10% rejection	20% rejection	30% rejection
SCN	87.35	86.85	86.63	87.28	72.67	72.82	74.29	75.80
CONF	88.30	91.78	93.97	95.90	72.69	73.00	73.11	72.97
DUL	88.04	90.11	92.58	94.50	73.67	76.01	79.10	82.25
RUL (Ours)	**88.98**	**92.72**	**95.40**	**97.35**	**73.75**	**77.24**	**80.98**	**84.75**

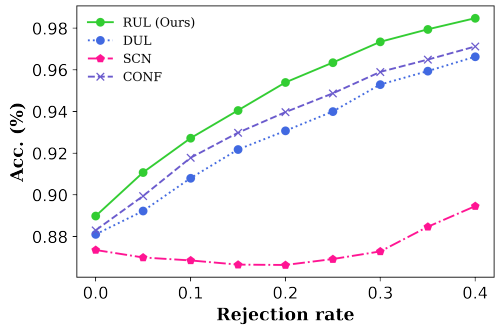

Figure 3: Accuracy versus rejection rate curves on RAF-DB.

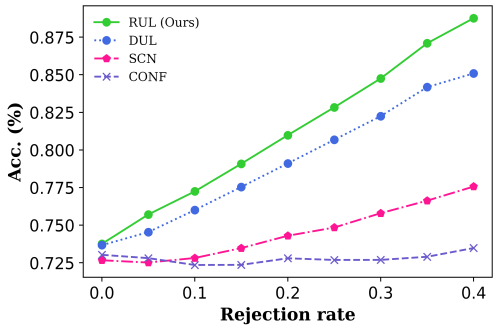

Figure 4: Accuracy versus rejection rate curves on FER2013 Private Testset.

4.5 Understand learned uncertainty values.

In this section, we provide qualitative and quantitative analysis to show what is the meaning of our learned uncertainty values.

Visualization of uncertainty values. We display some expression images in testset with their estimated uncertainty values in Figure 5. The first row shows that images with large uncertainty values are more likely to contain ambiguous expressions and they are easy to be misclassified (The wrong predicted labels are marked at the bottom of the first row images in red color). While the second row shows that images with small uncertainty values tend to contain obvious facial features and they are easy to be rightly classified. This illustrates that our learned uncertainty values can represent the difficulty of expression images as well as the model's confidence of its predictions. That property brings us benefits which are important for the development of safety AI systems. If the model knows when it is prone to be wrong (with high uncertainty value) then it can turn to human for help, which might avoid disasters in fields with high accuracy requirements.

Figure 5: The learned uncertainty values for different images. Images with large and small uncertainty values are shown in first and second rows respectively. Uncertainty values are marked in the upper left corner and predicted labels are marked at bottom. *Note that RUL learns large uncertainty values for images with ambiguous expressions while small uncertainty values for images with obvious expressions.*

Distribution of uncertainty values. RUL can learn different uncertainty distributions based on different datasets instead of constraining uncertainty distributions of all datasets to the same standard normal Gaussian distribution like DUL. We show the learned uncertainty distributions of different datasets in Figure 6 and Figure 7. In order to make comparison, we normalize RUL learned uncertainty values to [0,1]. Note that larger uncertainty values mean the images are more uncertain. We cut off at 400 on the y-axis in Figure 6 as SCN learns too many uncertainty values close to 0 which indicates bad performance. The results demonstrate that RUL learns large uncertainty values for more samples while the other two methods learn relative small uncertainty values (smaller than 0.4 on RAF-DB and smaller than 0.6 on FER2013) for all samples, which means RUL can get rid of the negative influence from strong learning ability of networks mentioned in Section 3.1. Furthermore, we also randomly sample several uncertain images and find that only RUL learns large uncertainty values for them.

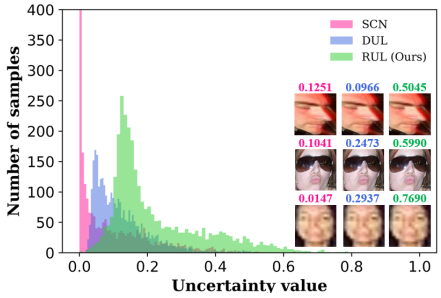

Figure 6: The learned uncertainty distribution of RAF-DB. *Uncertainty values are marked at the top of the image, different colors represent different methods.*

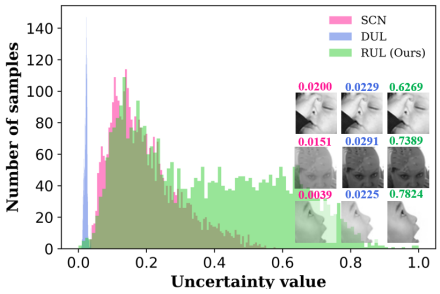

Figure 7: The learned uncertainty distribution of FER2013. *Uncertainty values are marked at the top of the image, different colors represent different methods.*

4.6 Comparison with state-of-the-art FER methods.

Although we aim to deal with uncertainty learning problems in FER, our RUL method still gets better test accuracy comparing to a series of state-of-the-art FER methods according to the results in Table 3 and Table 4.

Note that though RUL uses no extra data for co-training, it still achieves better performance. We speculate that this is because comparison of different FER images in the training process implicitly introduces data augmentation effect. Meanwhile, as uncertain images will get larger weights to mix with certain images, FER model will pay more attention to learning the uncertain images in the training process which improves the model's ability to recognize difficult facial expressions. We also plot the confusion matrix of different methods on RAF-DB in Figure 8, 9, 10 to verify our hypothesis. As mentioned by [23], FER model tends to make more mistakes in classes with less samples such as disgust and fear. However, the confusion matrices show that RUL increases test accuracy mainly in these two classes which means RUL can better recognize difficult facial expressions.

We also replace Gaussian variances in DUL with RUL learned uncertainty values to guide feature learning. Though RUL doesn't conform uncertainty distribution to a Gaussian distribution, it still improves DUL performance from 88.17% to 88.75%, showing RUL's superiority in uncertainty learning.

4.7 Other Experiments.

Experiments on MNIST and CIFAR. To show RUL is effective not only on FER tasks, we also carry out experiments in MNIST and CIAFR-10. Table 6 shows that RUL could still improve the performance of deep learning model though the accuracy is already saturate in MNIST and CIFAR-10. The test accuracy also increases along with the rejection of most uncertain test samples which illustrates that RUL can still learn meaningful uncertainty values in datasets besides FER tasks. We provide more experiments in appendix.

Table 3: Comparison on RAF-DB

Method	Acc.(%)
DLP-CNN [27]	84.22
gaCNN [29]	85.07
IPA2LT [50]	86.77
RAN [46]	86.90
SCN [45]	87.03
RUL	**88.98**

Table 4: Comparison on FER2013

Method	Acc.(%)
GoogleNet [12]	65.20
Conv+Inception [32]	66.40
Bag of Words [21]	67.40
Deep-Emotion [31]	70.02
VGG [35]	72.70
RUL	**73.75**

Table 5: Comparison of different structures.

Method	Acc.(%)
mix output	88.36
loss attention	88.75
RUL	**88.98**
Method	**10% rejection**
mix output	92.17
loss attention	89.32
RUL	**92.72**

Table 6: Test accuracy versus rejection rate results of different uncertainty learning methods on MNIST and CIFAR-10.

Method	MNIST				CIFAR-10			
	0 rejection	10% rejection	20% rejection	30% rejection	0 rejection	10% rejection	20% rejection	30% rejection
CONF	99.05	99.92	99.98	99.99	93.47	96.84	98.90	**99.56**
SCN	99.28	99.49	99.44	99.40	95.37	94.87	94.24	94.14
DUL	**99.47**	99.92	99.95	99.96	95.24	98.16	99.22	99.49
RUL (Ours)	99.45	**100.00**	**100.00**	**100.00**	**95.58**	**98.60**	**99.34**	99.44

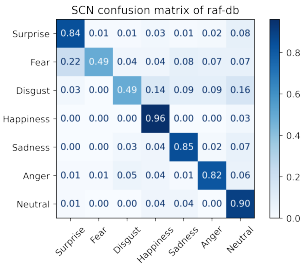

Figure 8: SCN confusion matrix on RAF-DB.

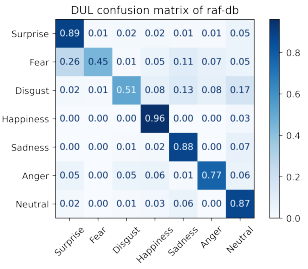

Figure 9: DUL confusion matrix on RAF-DB.

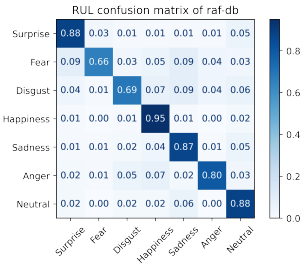

Figure 10: RUL confusion matrix on RAF-DB.

Ablation study. In order to show the mixed feature part is useful for uncertainty learning, we implement a mix output method and a loss attention method to make comparison. Specifically, we don't mix facial features in mix output method. Instead, we classify two images separately and mix the logits according to uncertainty values. Results are shown at Table 5. RUL performs better than mix output in both test accuracy and accuracy with 10% rejection, reflecting that mixed features may contain more relativity information than mixed logits. Loss attention method means we skip the mix part and use the uncertainty values to directly weight the loss values. Table 5 shows RUL outperforms loss attention method which illustrates that RUL is effective not only because uncertainty values affect loss values but also due to the interaction of different facial features during training process.

5 Conclusion

In this paper, we propose a novel and effective uncertainty learning method called Relative Uncertainty Learning for uncertainty quantification and noisy label training. Unlike traditional uncertainty learning methods which use uncertainty values to weight the loss and then add regularization to force the network learn uncertainty, we view uncertainty as a relative concept and learn uncertainty through the relativity of different samples. Extensive experiments have shown that RUL can learn meaningful uncertainty values which reflect the difficulty of samples being rightly recognized. RUL can also help FER model achieve state-of-the-art performance in both real-world and synthetic noisy FER datasets. We further demonstrate that RUL is also useful in other tasks such as MNIST and CIFAR-10.

References

[1] E. Barsoum, C. Zhang, C. C. Ferrer, and Z. Zhang. Training deep networks for facial expression recognition with crowd-sourced label distribution. In *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, pages 279–283, 2016.

[2] L. Best-Rowden and A. K. Jain. Learning face image quality from human assessments. *IEEE Transactions on Information Forensics and Security*, 13(12):3064–3077, 2018.

[3] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra. Weight uncertainty in neural network. In *International Conference on Machine Learning*, pages 1613–1622. PMLR, 2015.

[4] J. Chang, Z. Lan, C. Cheng, and Y. Wei. Data uncertainty learning in face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5710–5719, 2020.

[5] M. Dehghani, A. Severyn, S. Rothe, and J. Kamps. Avoiding your teacher's mistakes: Training neural networks with controlled weak supervision. *arXiv preprint arXiv:1711.00313*, 2017.

[6] T. DeVries and G. W. Taylor. Learning confidence for out-of-distribution detection in neural networks. *arXiv preprint arXiv:1802.04865*, 2018.

[7] A. Dhall, R. Goecke, S. Lucey, and T. Gedeon. Static facial expression analysis in tough conditions: Data, evaluation protocol and benchmark. In *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 2106–2112. IEEE, 2011.

[8] P. Ekman. An argument for basic emotions. *Cognition & emotion*, 6(3-4):169–200, 1992.

[9] C. Fabian Benitez-Quiroz, R. Srinivasan, and A. M. Martinez. Emotionet: An accurate, real-time algorithm for the automatic annotation of a million facial expressions in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5562–5570, 2016.

[10] Y. Gal. *Uncertainty in Deep Learning*. PhD thesis, University of Cambridge, 2016.

[11] Y. Gal and Z. Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR, 2016.

[12] P. Giannopoulos, I. Perikos, and I. Hatzilygeroudis. Deep learning approaches for facial emotion recognition: A case study on fer-2013. In *Advances in hybridization of intelligent methods*, pages 1–16. Springer, 2018.

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

[14] S. Gong, V. N. Boddeti, and A. K. Jain. On the capacity of face representation. *arXiv preprint arXiv:1709.10433*, 2017.

[15] I. J. Goodfellow, D. Erhan, P. L. Carrier, A. Courville, M. Mirza, B. Hamner, W. Cukierski, Y. Tang, D. Thaler, D.-H. Lee, et al. Challenges in representation learning: A report on three machine learning contests. In *International conference on neural information processing*, pages 117–124. Springer, 2013.

[16] P. Grother, M. Ngan, and K. Hanaoka. Face recognition vendor test-face recognition quality assessment concept and goals. In *NIST*, 2019.

[17] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger. On calibration of modern neural networks. In *International Conference on Machine Learning*, pages 1321–1330. PMLR, 2017.

[18] Y. Guo, L. Zhang, Y. Hu, X. He, and J. Gao. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In *European conference on computer vision*, pages 87–102. Springer, 2016.

[19] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[20] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

[21] R. T. Ionescu, M. Popescu, and C. Grozea. Local learning to improve bag of visual words model for facial expression recognition. In *Workshop on challenges in representation learning, ICML*, 2013.

[22] A. Kendall and Y. Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 5580–5590, Red Hook, NY, USA, 2017. Curran Associates Inc.

[23] S. Khan, M. Hayat, S. W. Zamir, J. Shen, and L. Shao. Striking the right balance with uncertainty. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 103–112, 2019.

[24] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 12 2014.

[25] A. D. Kiureghian and O. Ditlevsen. Aleatory or epistemic? does it matter? *Structural Safety*, 31(2):105–112, 2009. Risk Acceptance and Risk Communication.

[26] B. Lakshminarayanan, A. Pritzel, and C. Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6405–6416, 2017.

[27] S. Li, W. Deng, and J. Du. Reliable crowdsourcing and deep locality-preserving learning for expression recognition in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2852–2861, 2017.

[28] Y. Li, J. Yang, Y. Song, L. Cao, J. Luo, and L.-J. Li. Learning from noisy labels with distillation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1910–1918, 2017.

[29] Y. Li, J. Zeng, S. Shan, and X. Chen. Occlusion aware facial expression recognition using cnn with attention mechanism. *IEEE Transactions on Image Processing*, 28(5):2439–2450, 2018.

[30] Z. Li and S. Arora. An exponential learning rate schedule for deep learning. *arXiv preprint arXiv:1910.07454*, 2019.

[31] S. Minaee, M. Minaei, and A. Abdolrashidi. Deep-emotion: Facial expression recognition using attentional convolutional network. *Sensors*, 21(9):3046, 2021.

[32] A. Mollahosseini, D. Chan, and M. H. Mahoor. Going deeper in facial expression recognition using deep neural networks. In *2016 IEEE Winter conference on applications of computer vision (WACV)*, pages 1–10. IEEE, 2016.

[33] A. Mollahosseini, B. Hasani, and M. H. Mahoor. Affectnet: A database for facial expression, valence, and arousal computing in the wild. *IEEE Transactions on Affective Computing*, 10(1):18–31, 2017.

[34] N. Natarajan, I. S. Dhillon, P. Ravikumar, and A. Tewari. Learning with noisy labels. In *NIPS*, volume 26, pages 1196–1204, 2013.

[35] C. Pramerdorfer and M. Kampel. Facial expression recognition using convolutional neural networks: State of the art. *CoRR*, abs/1612.02903, 2016.

[36] Y. Shi and A. K. Jain. Probabilistic face embeddings. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6902–6911, 2019.

[37] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[38] S. Sukhbaatar and R. Fergus. Learning from noisy labels with deep neural networks. *arXiv preprint arXiv:1406.2080*, 2(3):4, 2014.

[39] E. Tabassi and P. Grother. Biometric sample quality., 2015.

[40] P. Terhorst, J. N. Kolf, N. Damer, F. Kirchbuchner, and A. Kuijper. Ser-fiq: Unsupervised estimation of face image quality based on stochastic embedding robustness. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5651–5660, 2020.

[41] S. Thulasidasan, G. Chennupati, J. A. Bilmes, T. Bhattacharya, and S. Michalak. On mixup training: Improved calibration and predictive uncertainty for deep neural networks. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[42] M. Valstar and M. Pantic. Induced disgust, happiness and surprise: An addition to the mmi facial expression database. *Proc. Int'l Conf. Language Resources and Evaluation, Workshop EMOTION*, pages 65–70, 01 2010.

[43] A. Veit, N. Alldrin, G. Chechik, I. Krasin, A. Gupta, and S. Belongie. Learning from noisy large-scale datasets with minimal supervision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 839–847, 2017.

[44] V. Verma, A. Lamb, C. Beckham, A. Najafi, I. Mitliagkas, D. Lopez-Paz, and Y. Bengio. Manifold mixup: Better representations by interpolating hidden states. In *International Conference on Machine Learning*, pages 6438–6447. PMLR, 2019.

[45] K. Wang, X. Peng, J. Yang, S. Lu, and Y. Qiao. Suppressing uncertainties for large-scale facial expression recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6897–6906, 2020.

[46] K. Wang, X. Peng, J. Yang, D. Meng, and Y. Qiao. Region attention networks for pose and occlusion robust facial expression recognition. *IEEE Transactions on Image Processing*, 29:4057–4069, 2020.

[47] J. Wu. Facial-expression-recognition.pytorch github. `https://github.com/WuJie1010/Facial-Expression-Recognition.Pytorch`. Accessed: 2020-08-07.

[48] T. Yu, D. Li, Y. Yang, T. M. Hospedales, and T. Xiang. Robust person re-identification by modelling feature uncertainty. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 552–561, 2019.

[49] U. Zafar, M. Ghafoor, T. Zia, G. Ahmed, A. Latif, K. R. Malik, and A. M. Sharif. Face recognition with bayesian convolutional networks for robust surveillance systems. *EURASIP Journal on Image and Video Processing*, 2019(1):1–10, 2019.

[50] J. Zeng, S. Shan, and X. Chen. Facial expression recognition with inconsistently annotated datasets. In *Proceedings of the European conference on computer vision (ECCV)*, pages 222–237, 2018.

[51] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018.

[52] G. Zhao, X. Huang, M. Taini, S. Z. Li, and M. PietikäInen. Facial expression recognition from near-infrared videos. *Image and Vision Computing*, 29(9):607–619, 2011.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [No]
   (c) Did you discuss any potential negative societal impacts of your work? [No]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] The code will be released soon.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section 4.2.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See Section 4.3.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Section 4.2.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See Section 4.1.
   (b) Did you mention the license of the assets? [No]
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]