CS229 Project Final Report (2020 Fall)
Category: Computer Vision
Tackling Long Tail Problem for Facial Expression Recognition In the Wild

Xinqi Fan, Bin (Claire) Zhang, Megan (Liwen) Zhang
{xinqifan, zhangbin, lzhang8}@stanford.edu

Abstract

Facial expression recognition plays an important role in human-robot interaction, education, health treatment, etc. In order to facilitate real-world application of expression recognition, it is crucial to tackle the problem in-the-wild, which suffers from the long tail problem. In this work, we propose a novel paradigm named ***disentangled and balanced learning****. The proposed method first learns disentangled features through self-supervised contrastive learning, then separately learns a class-balanced classifier in a supervised manner. Comprehensive experiments and analysis were conducted on RAF-DB. We drew comparisons with classical re-weighting and re-sampling methods, recently-proposed models for long tail problems and other deep learning methods on the same dataset. Results show that the proposed method has satisfactory accuracy, outperforming most of the listed methods in terms of average and overall accuracy. In addition, the proposed method is transferable to other applications for tackling the long tail problem.*

1. Introduction

Facial expressions, an important form of nonverbal communication, allow humans to convey information about their feelings [7]. Studies have been conducted to teach machines to recognize facial expressions. Facial expression recognition (FER) has broad applications in self-driving cars, smart hospitals, etc. Human facial expressions can be divided into seven categories, including one neutral emotion and six basic emotions - angry, disgusted, fearful, happy, sad, and surprised [12].

Several achievements have been made by using machine learning and deep learning-based methods for FER, including local binary patterns [18], suppressing uncertainty [21], and occlusion aware [14]. One significant issue, however, is the long tail (LT) problem in FER in-the-wild, referring to the skewed distribution of datasets. Even though visual recognition algorithms nowadays have high classification performance in most cases, the relatively low number of training examples of some categories can still significantly hurt the classification accuracy of the minority class [9]. The purpose of this project is to develop an algorithm that lowers the long tail problem's influence on FER classification performance.

In this study, we propose a novel paradigm called disentangled and balanced learning (DBL) for tackling the long tail problem for FER. The proposed method first learns disentangled features through self-supervised contrastive learning (CL), and then separately learns a class-balanced classifier through re-sampled data in a supervised manner. Through contrastive learning, the network is able to learn disentangled features by minimizing the inner-class distance and maximizing the inter-class distance. With the obtained good features, class-balanced data is provided to reduce the bias to majority classes of the classifier. Comprehensive experiments showed that proposed method can outperform classical re-weighting and re-sampling models by at least 2%. The distinction in performance is more significant in terms of average accuracy.

The rest of the report is organized in the following way. In Section 2, we briefly discuss the related work for the FER and LT problem. Then, we explain the proposed method in Section 3. Detailed experiments and result analysis are given in Section 4. Finally, we conclude the paper and discuss potential ways that can be used to further test and enhance this model.

2. Related Work

2.1. Facial Expression Recognition

In the past decades, many hand-crafted based methods have been developed for FER. Feature extractors, such as local binary pattern (LBP) [22], speeded up robust features (SURF) [2], and Gabor filters [19] followed by a SVM classifier, were developed for and applied to FER. Recently, convolution neutral network (CNN) based feature extraction methods gained much popularity. Barsoum *et al*. [1] proposed to train CNN with noisy labels for FER. Zhang *et*

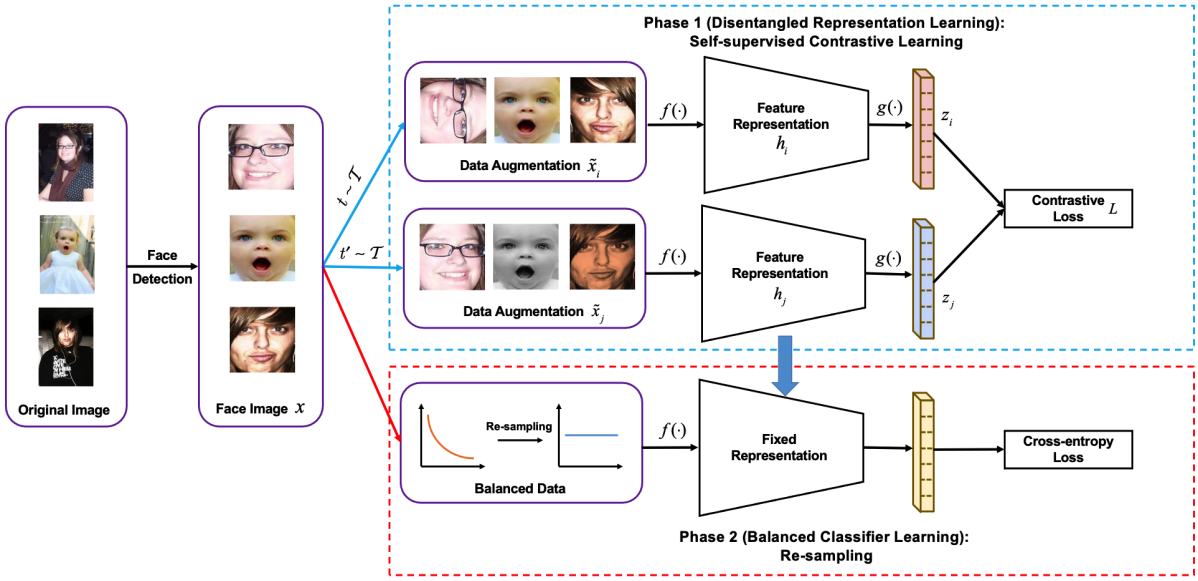


Figure 1: Schematic of Proposed Network Pipeline. Phase 1 focuses on disentangled representation learning through self-supervised contrastive learning; Phase 2 focuses on balanced classifier learning through re-sampled data.

al. [25] proposed a joint pose and expression modeling in an end-to-end manner. Fernandez *et al*. [16] proposed a CNN enhanced by attention mechanism to focus on facial expression related parts.

2.2. The Long Tail Problem

Due to the unbalanced nature of real-world data, the LT problem is unavoidable in machine learning. Conventional methods [3, 8] proposed to solve this problem, including data re-sampling [5] and re-weighting [20], were shown to hurt the representative ability by distorting data distributions in exchange for better classifier-learning [27]. Other attempts worth investigating include quintuplet loss [10], rangeloss [26], and center invariant loss [24].

Most recently, studies have shown the noticeable effectiveness in decoupling representation and classifier learning [11] and bilateral branch network [27]. The commonality between these two approaches is that they both first learn a good feature and then separately learn a classifier from the re-balanced data.

3. Dataset and Features

Two of the most widely-used FER in-the-wild databases, Real-world Affective Faces Database (RAF-DB) [13] and Expression in-the-Wild (ExpW) [17] both suffer from long tail problem. For example, RAF-DB has around 2000 images of happy, but only 200 images of fearful. For this project, we ran experiments on RAF-DB, which has separate training and test sets. The image resolution is 100×100. There are 15338 images in the training set and 3069 in test set. We further split the original training set into training and validation sets by the ratio of $9:1$. Some examples of images from the RAF-DB dataset are shown in Fig. 2.

Data augmentation is indispensable to the construction of an effective contrastive learning model. Thus, we adopted several different approaches for data augmentation, such as cropping, rotation, and colorization. We did not manually perform any feature extraction, as it is inherently conducted in the hidden layers of our deep learning model.

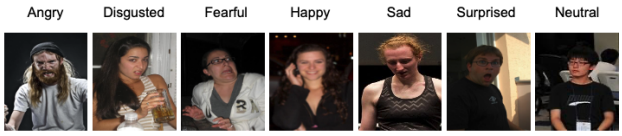


Figure 2: Examples of Datasets - RAF-DB.

4. Methodology

In order to solve the long tail problem, we propose a new paradigm DBL illustrated in Fig. 1. DBL first learns disentangled features through self-supervised CL (Phase 1), and then learns a classifier through re-sampled data in a supervised manner (Phase 2). Details of the two phases are given in the following subsections individually.

4.1. Disentangled Feature Learning

Self-supervised learning (SSL), a form of unsupervised learning, is implemented to learn features without any given

labels. More specifically, CL, a subclass of SSL, is chosen to provide disentangled feature representation by minimizing intra-class distance and maximizing inter-class distance.

We adopt and modify the simple framework for CL of visual representation (SimCLR) framework [6] in Fig. 1 with a blue bounding box, which maximizes the agreement between differently augmented perspectives of the same image. The framework starts with a data augmentation module where two separate data augmentation operators from the same family of augmentation family, $t \sim \mathcal{T}$ and $t' \sim \mathcal{T}$, are applied to the same sample x to give the two views $\tilde{x_i}$ and $\tilde{x_j}$. The augmentation implemented include random crop, random flip, and random color distortion. It then uses a deep neural network $f(\cdot)$ parameterized by θ_{feature} to extract representations h_i and h_j to feed into a small neural network projection head $g(\cdot)$. $g(\cdot)$ maps the representations into latent space, where the contrastive loss for a positive pair of examples (i, j) is applied on z_i and z_j as

$$l_{i,j} = -\log \frac{\exp(\text{sim}(z_i, z_j)/\tau)}{\sum_{k=1}^{2N} \mathbb{1}\{k \neq i\} \exp(\text{sim}(z_i, z_k)/\tau)}, \quad (1)$$

where N is the number of samples in the minibatch and τ is a temperature parameter. The similarity metric is implemented as

$$\text{sim}(z_i, z_j) = \frac{z_i^T z_j}{||z_i|| ||z_j||}. \quad (2)$$

Total loss is minimized by updating f and g and implemented as

$$\mathcal{L} = \frac{1}{2N} \sum_{k=1}^{N} [l(2k-1, 2k) + l(2k, 2k-1)]. \quad (3)$$

4.2. Balanced Classifier Learning

After learning the disentangled features through CL, we then provide class-balanced data to train the classifier with frozen features.

We adopt re-sampling strategy to obtain class-balanced data. Re-sampling involves repeatedly drawing certain training examples form training data by achieving a uniform distribution over classes as [4]

$$p_j = 1/C, \quad (4)$$

which stands for the probability of sampling a point from class j, out of a total of C classes. To approximate the uniform distribution, we implement a weight w to sample the input data by taking the nomarlized reciprocal of the number of samples for each class j as

$$w_j = \frac{\alpha/n_j}{\max(\{1/n_j\}_{j=1}^{C})}, \quad (5)$$

where n_j is the number of samples for class j and α is the normalization constant s.t. $\sum_{j=1}^{C} w_j = 1$. In order to perform supervised classifier learning, we need to attach a new classifier at the end of the features. The attached classifier is realized by a multi-layer perceptron (MLP) with C outputs parameterized by θ_{cls} as

$$f_{cls}(\theta_{cls}) := \text{MLP}(\theta_{cls}), \quad (6)$$

where cls stands for classifier. Then, we re-train the network for a fewer number of epochs by only updating the classifier θ_{cls} (i.e. $\theta_{feature}$ remains unchanged). This approach is addressed as classifier re-training (cRT) in [11]. In this case, we use multi-class cross-entropy loss as

$$\mathcal{J} = -\frac{1}{n} \sum_{i=1}^{n} \log \frac{e^{s_{y_i}}}{\sum_{j=1}^{C} e^{s_j}}, \quad (7)$$

where s_j is the normalized prediction after softmax, s_{y_i} is the probability of the ground truth class of example i.

5. Experimental Result and Discussion

5.1. Experiment Setup and Evaluation Metrics

We implemented the proposed method according to Algorithm 1 using PyTorch and PyTorch Lightning deep learning framework. We adopted NVIDIA 2070s, 2080Ti, Tesla T4 and P100 Graphical Computer Units (GPUs) on local computers, Google Colab and Google Cloud Platforms. We choose residual network with 18 layers (ResNet-18) as the backbone network.

In terms of evaluation metrics, we used overall accuracy, average accuracy against each class, and confusion matrix to assess performance of the proposed model.

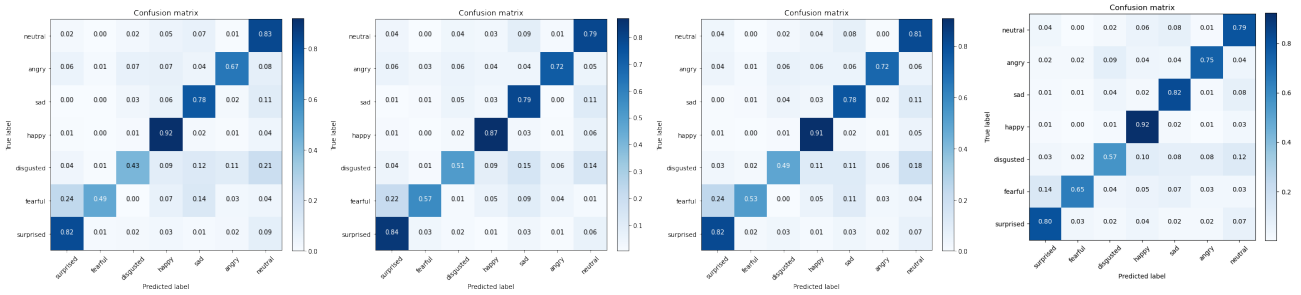
5.2. Comparison with Other Methods

To begin with, we ran experiments on the basic model, some classical methods widely adopted when tackling LT

Algorithm 1 Disentangled and Balanced Learning

Require: Model θ, Data $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{n}$

Disentangled Feature Learning ▷ Contrastive Learning
for $t = 1$ to T_0 **do**
  $\mathcal{B} \leftarrow \text{SampleMiniBatch}(\mathcal{D}, m)$ ▷ Minibatch Size m
  $\tilde{x} \leftarrow \mathcal{T}(x)$ ▷ Data Augmentation
  $\mathcal{L} \leftarrow$ update loss using (3)
  $\theta \leftarrow \theta - \alpha \nabla_\theta \mathcal{L}(\theta)$
Balanced Classifier Learning ▷ Classifier Re-training
for T_0 to T **do**
  $\mathcal{B} \leftarrow \text{SampleMiniBatch}(\mathcal{D}, m, w)$ ▷ Sample Weight w
  $\theta_{\text{feature}} \leftarrow \theta(\text{feature})$ ▷ Freeze Feature
  $\mathcal{J} \leftarrow$ update loss using (7)
  $\theta_{\text{cls}} \leftarrow \theta_{\text{cls}} - \alpha \nabla_{\theta_{\text{cls}}} \mathcal{J}(\theta_{\text{cls}})$



(a) Basic Model (b) Re-weighting Model (c) Re-sampling Model (d) Proposed Method

Figure 3: Confusion Matrix of Models on RAF-DB

problems (re-weighting and re-sampling), and the decoupling method [11].

First, we compare re-weighting, re-sampling, and decoupling methods with the basic model. Re-weighting and re-sampling both result in higher average accuracy, by 2.34% and 1.8% respectively, while compromising the overall accuracy. The reason is that we assign higher weights on minority classes during training stage, hence improving the model's performance on these categories and resulting in better average accuracy. However, as we are laying less emphasis on the majority classes, re-sampling and re-weighting do not lead to higher overall accuracy. For the decoupled model, we were unable to reproduce satisfactory experimental results as claimed in [11]. We observed 0.91% of increase in average accuracy. This may be caused by hyper-parameter tuning and/or errors in our decoupling model.

Next, we compare our proposed model with other methods listed in Table 1. We can observe significant improvement in average accuracy. Specifically, there is a 5.52%, 3.18%, 3.72%, and 4.51% increase in average accuracy when compared with basic, re-weighting, re-sampling, and decoupling model, respectively. By reading the confusion matrices in Fig. 3, we can see that DBL reaches higher accuracy on minority classes (e.g. disgusted and fearful). It also outperforms other loss function design (i.e. Center Loss [23]) and deep learning-based models (i.e. base DCNN[13], DLP-CNN[13], Lian *et al*. [15]) in Table 1 on average accuracy. In terms of overall accuracy, the proposed method also has better performance than the classical models. The improvement is a result of the disentangled feature learning mechanism in the design of our DBL model.

Table 1: Comparison of Different Models on RAF-DB

Model	Average Accuracy	Overall Accuracy
Basic	70.35	81.68
Re-weighting	72.69	80.28
Re-sampling	72.15	81.64
Decoupling [11]	71.36	80.73
Center Loss [23]	72.87	83.68
base DCNN [13]	72.42	82.86
DLP-CNN [13]	74.20	84.13
Lian *et al*. [15]	-	82.69
Proposed Method	**75.87**	83.21

6. Conclusion and Future Work

In this work, the proposed novel paradigm **disentangled and balanced learning** first extracts disentangled features through self-supervised contrastive learning, and then learns a class-balanced classifier through re-sampled data separately. The results on widely-used datasets on FER show that the proposed method improved the average accuracy and overall accuracy for around 5% and 2%, and outperforms all the other listed methods in terms of average accuracy. We believe this result suggests the importance of learning the correct features from the original dataset, including its long-tailed nature. DBL could be further tested on its potential by increasing the CL training batch size and the number of epochs. Due to some computing constraints, we were only able to implement a batch size and number of epochs that are about or more than half-fold smaller than the smallest value suggested in the SimCLR paper. We could also tune the hyper-parameters, especially the learning rate, to further optimize DBL.

In addition, with its proved ability to tackle the long-tailed FER problem, DBL could be transferred to other needs as essentially data for any application have some level of unevenness.

7. Contribution

Xinqi: Ran experiments with SimCLR and SimCLR classifier on local computers, wrote report sections regarding FER, proposed method and its experiment.

Claire: Ran experiments with SimCLR and its classifier on Google Cloud Platform, and wrote report sections regarding the long tail problem, CL/SimCLR and conclusion.

Megan: Ran experiments with basic/classical/decoupled models on RAF-DB, wrote sections on dataset and experimental results in the final report.

References

[1] Emad Barsoum, Cha Zhang, Cristian Canton Ferrer, and Zhengyou Zhang. Training deep networks for facial expression recognition with crowd-sourced label distribution. In *Proceedings of the ACM International Conference on Multimodal Interaction*, pages 279–283, 2016.

[2] Martin Bäuml and Rainer Stiefelhagen. Evaluation of local features for person re-identification in image sequences. In *IEEE International Conference on Advanced Video and Signal based Surveillance*, pages 291–296. IEEE, 2011.

[3] Paula Branco, Luís Torgo, and Rita P Ribeiro. A survey of predictive modeling on imbalanced domains. *ACM Computing Surveys*, 49(2):1–50, 2016.

[4] Mateusz Buda, Atsuto Maki, and Maciej A. Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *CoRR*, abs/1710.05381, 2017.

[5] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority oversampling technique. *Journal of Artificial Intelligence Research*, 16:321–357, 2002.

[6] Ting Chen, Simon Kornblith, et al. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020.

[7] P Ekman and H Oster. Facial expressions of emotion. *Annual Review of Psychology*, 30(1):527–554, 1979.

[8] Guo Haixiang, Li Yijing, Jennifer Shang, Gu Mingyun, Huang Yuanyue, and Gong Bing. Learning from class-imbalanced data: Review of methods and applications. *Expert Systems with Applications*, 73:220–239, 2017.

[9] Grant Van Horn and Pietro Perona. The devil is in the tails: Fine-grained classification in the wild. *CoRR*, abs/1709.01450, 2017.

[10] Chen Huang, Yining Li, Chen Change Loy, and Xiaoou Tang. Learning deep representation for imbalanced classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5375–5384, 2016.

[11] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, et al. Decoupling representation and classifier for long-tailed recognition. *International Conference on Learning Representations*, 2019.

[12] Shan Li and Weihong Deng. Deep facial expression recognition: A survey. *IEEE Transactions on Affective Computing*, 2020.

[13] Shan Li, Weihong Deng, et al. Reliable crowdsourcing and deep locality-preserving learning for expression recognition in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2852–2861, 2017.

[14] Yong Li, Jiabei Zeng, Shiguang Shan, and Xilin Chen. Occlusion aware facial expression recognition using cnn with attention mechanism. *IEEE Transactions on Image Processing*, 28(5):2439–2450, 2018.

[15] Zheng Lian, Ya Li, Jian-Hua Tao, Jian Huang, and Ming-Yue Niu. Expression analysis based on face regions in read-world conditions. *International Journal of Automation and Computing*, 17(1):96–107, 2020.

[16] Pedro D Marrero Fernandez, Fidel A Guerrero Pena, Tsang Ren, and Alexandre Cunha. Feratt: Facial expression recognition with attention net. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[17] Ali Mollahosseini, Behzad Hasani, and Mohammad 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.

[18] Caifeng Shan, Shaogang Gong, and Peter W McOwan. Robust facial expression recognition using local binary patterns. In *IEEE International Conference on Image Processing 2005*, volume 2, pages II–370. IEEE, 2005.

[19] Caifeng Shan, Shaogang Gong, and Peter W McOwan. Facial expression recognition based on local binary patterns: A comprehensive study. *Image and Vision Computing*, 27(6):803–816, 2009.

[20] Kai Ming Ting. A comparative study of cost-sensitive boosting algorithms. In *Proceedings of the International Conference on Machine Learning*, pages 983–990, 2000.

[21] Kai Wang, Xiaojiang Peng, Jianfei Yang, Shijian Lu, and Yu Qiao. Suppressing uncertainties for large-scale facial expression recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 6897–6906, 2020.

[22] Wencheng Wang, Faliang Chang, Jianguo Zhao, and Zhenxue Chen. Automatic facial expression recognition using local binary pattern. In *World Congress on Intelligent Control and Automation*, pages 6375–6378. IEEE, 2010.

[23] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *European Conference on Computer Vision*, pages 499–515. Springer, 2016.

[24] Yue Wu, Hongfu Liu, Jun Li, et al. Deep face recognition with center invariant loss. In *Proceedings of the on Thematic Workshops of ACM Multimedia*, pages 408–414, 2017.

[25] Feifei Zhang, Tianzhu Zhang, Qirong Mao, and Changsheng Xu. Joint pose and expression modeling for facial expression recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3359–3368, 2018.

[26] Xiao Zhang, Zhiyuan Fang, Yandong Wen, Zhifeng Li, and Yu Qiao. Range loss for deep face recognition with long-tailed training data. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5409–5418, 2017.

[27] Boyan Zhou, Quan Cui, Xiu-Shen Wei, et al. Bbn: Bilateral-branch network with cumulative learning for long-tailed visual recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 9719–9728, 2020.