Fact sheet: ECCV 2020 ChaLearn Looking at People 1st Fair Face Recognition Challenge

This is the fact sheet's template for the ECCV 2020 ChaLearn Fair Face Recognition Challenge[1]. Please fill out the following sections carefully in a scientific writing style. Then, send the compressed project (in `.zip` format), i.e., the generated PDF, `.tex`, `.bib` and any additional files to `juliojj@gmail.com`, and put in the Subject of the email "ECCVW 2020 FairFaceRec Challenge / Fact Sheets", following the schedule and instructions provided in the Challenge webpage[1] (post-challenge/fact sheets).

I. Team details

- Team leader name: Jun Yu
- Username on Codalab:ustc-nelslip
- Team leader affiliation:University of Science and Technology of China
- Team leader address: Department of Automation, University of Science and Technology of China
- Team leader phone number: +86-13856070316
- Team leader email: harryjun@ustc.edu.cn
- Name of other team members (and affiliation): Xinlong Hao (University of Science and Technology of China), Haonian Xie (University of Science and Technology of China), Ye Yu (Hefei University of Technology)
- Team website URL (if any):

II. Contribution details

A. Title of the contribution

Recent research has shown many face recognition systems have lower performance for certain demographic groups than others, such face recognition systems are said to be biased in terms of demographics. In this paper, we do some works to reduce the bias in automated face recognition. Firstly, we test different face detection algorithms and find an effective face cropped method. Secondly, a data re-sampling method is used to balance the data distribution by under-sampling the majority class, we also find that using external data to balance data is also helpful. Then, after analyzing train data, we use many training data enhancement methods to improve the performance of our model. Finally,we get eight models with different backbones and head loss, and get our final prediction by combining them. Our team ustc-nelslip get 1st in the development stage and get second in the test stage.

B. Introduction and Motivation

As a part of face recognition preprocessing, face detection is very important. Given a image with a loosely cropped face roughly in the center of the image, we crop image in the center to get a face directly, but it turned out to be a poor performance, because the proportion of the face in the image is different. Then we test dlib[2],retinaface[3] and dsfd[4], the performance of dsfd is best.And face alignment is not useful, because there are many hard face, face alignment would not help for getting better face.

Then we use a data re-sampling method to balance the data distribution by under-sampling the majority class[5], the class is divided based on gender and skin color. Compare to the method without data balance, our method get better performance in bias and the accuracy has hardly changed. Moreover, we find using external data to balance data is also helpful, it can be helpful to reduce the bias.

After analyzing the train data, we find many images with low qulitaty and there is a very big difference in light. So we use many training data enhancement methods, such as Color Jitter and RandomCrop, these measures help to increase the diversity of samples and improve result accuracy.

Finally, our method uses two different backbones: Ir-resnet50, Ir-resnet152 and two different headers: arcface[6], cosface[7], we also change the classified loss to focal loss[8].In the end, we get eight models with diferent backbones and head loss, and get our final predictions by linear combination of prediction results of single models.

C. Representative image / workflow diagram of the method

The workflow diagram of our method is as shown in Fig1.

D. Detailed method description

Data processing: We use dsfd to detect face in the image, hyper-parameters is as following: confidence_threshold=0.7, nms_iou_threshold=0.3, max_resolution=1080. And if we get many face from one image, We choose the face closest to the center of the image, if no images are detected, we just crop the image in the centure by the rate 7/12. Finally, we resize the face to 112x112.

Data balance: First, We count the number of different class persons, there are four classes base on gender and skin, and the number is 1490:1448:844:514. We just use a data re-sampling method to balance the data distribution by under-sampling the majority class.

Data_enhancement: Research has shown there are many faces are in low quality, they are difficult to identify, so we make some images blur randomly, this method can increase the recognition rate of low quality images. Besides, we also use Color Jitter and RandomCrop to increase image diversity. Hyper-parameters are as following: ColorJitter(brightness=(0.6,1.4),contrast=(0.6,1.4),saturation=(0.6,1.4))

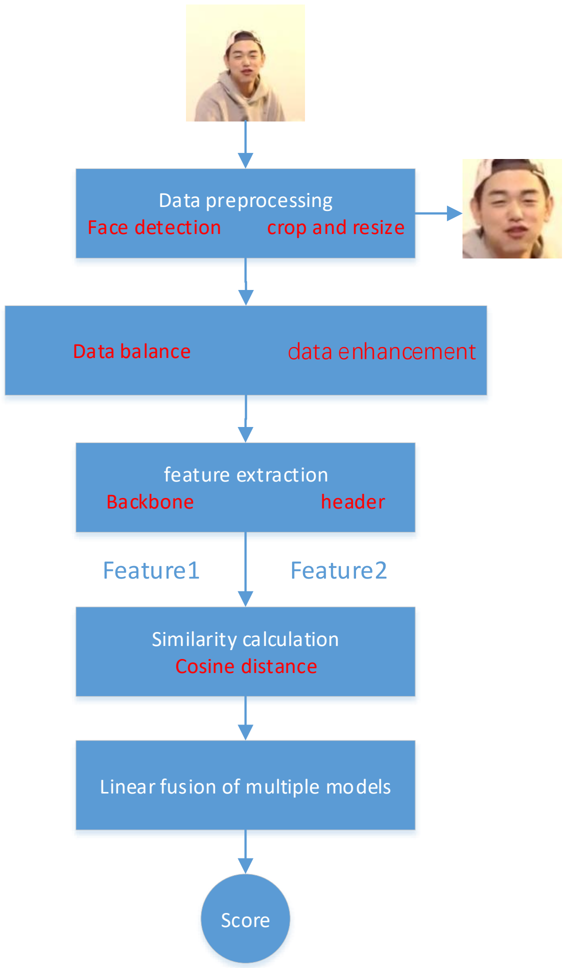

Fig. 1. Workflow diagram of our method

Backbone and header: We use Ir-resnet50, Ir-resnet152 backbone, and we use arcface and cosface header, besides, we use focal loss as Classified loss.

Model fusion: We get different models with different data enhancement methods, backbones and headers, and we get eight prediction scores independently. Finally, we get our final prediction by linear combination of prediction results of single models.

Other parameter: 4*gtx 2080 gpu are used, our BATCH_SIZE=256, we use SGD optimizer, initial_lr = 0.01, WEIGHT_DECAY = 5e-4, MOMENTUM = 0.9.NUM_EPOCH=60, STAGES = [25,48,54].

E. Challenge results and final remarks

Fill Table I with your obtained results, shown in the leaderboard of the challenge[1]. Note, if you joined the challenge in the test phase, keep the "development" row blank.

[1] https://competitions.codalab.org/competitions/24123

TABLE I

LEADERBOARD: RESULTS OBTAINED BY THE PROPOSED APPROACH.

Phase	Rank	Bias positive pairs	Bias negative pairs	Accuracy
Development	1	0.002956	0.000142	0.999287
Test	2	0.000172	0.000175	0.999569

III. ADDITIONAL METHOD DETAILS

Please reply if your challenge entry considered (or not) the following strategies and provide a brief explanation.

- **Did you use pre-trained models?** (✓) Yes, ( ) No
  We just use two different pre-trained models, one is ir_resnet50 model, the other is ir_resnet152 model, you can find them easily in the link: https://github.com/ZhaoJ9014/face.evoLVe.PyTorch
- **Did you use external data?** (✓) Yes, ( ) No
  We use MS-Celeb-1M and DeepGlint to get better pre-trained models and we also use them to do some works about data balance.
- **Did you use other regularization strategies/terms?** (✓) Yes, ( ) No
  But we just use dropout and Early stopping.
- **Did you use handcrafted features?** ( ) Yes, (✓) No
- **Did you use any face detection, alignment or segmentation strategy?** (✓) Yes, ( ) No
  We try many face detection method,include Dlib, Retinaface and DSFD ,and we find DSFD would help us get best score.
- **Did you use ensemble models?** (✓) Yes, ( ) No
  We try many train strategies, so we get many models with diferent score and performance, we choose eight models from them, and we get our ensemble model by combining them.
- **Did you use different models for different protected groups?** ( ) Yes, (✓) No
- **Did you explicitly classify the legitimate attributes?** ( ) Yes, (✓) No
- **Did you explicitly classify other attributes (e.g. image quality)?** ( ) Yes, (✓) No
- **Did you use any pre-processing bias mitigation technique (e.g. rebalancing training data)?** (✓) Yes, ( ) No
  we try two pre-processing tenchniques. First,we make a vector of weights for each image in the dataset, based on class frequency, the returned vector of weights can be used to create a WeightedRandomSampler for a DataLoader to have class balancing when sampling for a training batch. PS: the class is just about gender and skin color. Second, we calculated the number of persons of different colors and genders. Then, we use MS-Celeb-1M and DeepGlint to balance data by adding face images in train data.
- **Did you use any in-processing bias mitigation**

technique (e.g. bias aware loss function)?
( ) Yes, (✓) No

- **Did you use any post-processing bias mitigation technique?** ( ) Yes, (✓) No
  But we do some work about model fusion, the method can help us reduce the positive pairs bias value and negative pairs bias value at the same time.

IV. Code repository

Link to a code repository with complete and detailed instructions so that the results obtained on Codalab can be reproduced locally. This includes a list of requirements, pre-trained models, and so on. Note, training code with instructions is also required. This is recommended for all participants and mandatory for winners to claim their prize. **Organizers strongly encourage the use of docker to facilitate reproducibility**.

Code repository: our code can be easily download from the url `https://drive.google.com/file/d/1MxBHarqjvWNsl8UAS5mf0oJQetRWBQaT/view?usp=sharing`

References

[1] ChaLearnLAP. ECCV 2020 ChaLearn Fair Face Recognition Challenge. [Online]. Available: http://chalearnlap.cvc.uab.es/challenge/38/description/

[2] King D E. Dlib-ml: A machine learning toolkit[J]. The Journal of Machine Learning Research, 2009, 10: 1755-1758.

[3] Deng J, Guo J, Zhou Y, et al. Retinaface: Single-stage dense face localisation in the wild[J]. arXiv preprint arXiv:1905.00641, 2019.

[4] Li J, Wang Y, Wang C, et al. DSFD: dual shot face detector[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 5060-5069.

[5] Drummond, C., Holte, R.C., et al.: C4. 5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling. In: Workshop on Learning from Imbalanced Datasets II. Citeseer (2003)

[6] Deng J, Guo J, Xue N, et al. Arcface: Additive angular margin loss for deep face recognition[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 4690-4699.

[7] Wang H, Wang Y, Zhou Z, et al. Cosface: Large margin cosine loss for deep face recognition[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 5265-5274.

[8] Lin T Y, Goyal P, Girshick R, et al. Focal loss for dense object detection[C]//Proceedings of the IEEE international conference on computer vision. 2017: 2980-2988.