FER: A Performance Comparison of Deep Neural Network Architectures

Loop Q Prize 2021 Competition Report

Chenxiang Zhang
Department of Computer Science
University of Pisa
c.zhang4@studenti.unipi.it

Abstract

Facial expression recognition (FER) is an important task to improve cross-domain human-computer interaction systems. Given the dataset provided by the competition organizer, we modified and applied the current state-of-the-art convolutional neural networks (CNNs) to the FER task. Our best single model obtain a 69.1% in the validation accuracy, while our ensemble model achieves 71.1%. Additionally, we investigated the quality of the models and the limits of the given dataset.

1 Introduction

Facial expression recognition (FER) is an universal and explicit way of interaction among humans from different cultural and social backgrounds. Being able to automatically classify a face expression can improve different cross-domain human-computer interaction systems such as the medical field, digital advertisement and customer feedback assessment. In computer vision, various FER systems have been used to represent and classify facial expressions.

Since 2013, the study of FER has transitioned from lab-controlled to in-the-wild conditions thanks to the building of dataset such as FER2013 [1], and the use of deep neural networks. Lab-controlled environments are easier to solve thanks to their limited constraints, such as specific face poses and consistent environment. The current researches focus on the classification of in-the-wild face images, which are data collected from real-world scenarios. As opposed to the controlled environment, in-the-wild images present numerous and different variations: pose, illumination, occlusions, gender, ages, and ethnicity. These diversities contribute in making FER a challenging task. In recent years, the computer vision field has been dominated by deep convolutional neural networks (CNNs). These architectures are able to automatically learn useful features given enough data. CNNs have been applied for the in-the-wild FER and achieved great results.

In this work, we explored and applied the current state-of-the-art deep neural networks for the competition Loop Q Prize 2021. Given the dataset FERLoopAI provided by the organizer, we conducted various experiments with different model architectures, optimizers, schedulers and hyper-parameters in order to achieve a great performance on the given dataset. Furthermore, we explored the characteristics of the dataset, discussed its limits and the possible solutions to achieve a better system.

2 Experiments

2.1 Datasets

The primary dataset **FERLoopAI** is provided by the organizer of the competition. This dataset has 30503 grey-scale images of dimension 48x48. The classes and their distribution in absolute numbers are: *Angry (4198), Disgust (466), Fear (4320), Happy (7682), Sad (5177), Surprise (3401), Neutral (5259)*. As we can see, the dataset presents a strong class imbalance bias. Furthermore, we also noticed that some images do not contain a face. We preprocessed the dataset by computing the standard deviation for each image and removed those that are below a certain threshold (0.1). In total 65 images were removed from the dataset. The structure of FERLoopAI is the same as the dataset FER2013 [1], they share the same input size, classes and they both present a strong class imbalance. FER2013 is composed by 35887 grey-scale images of dimension 48x48 and it is already divided in 28709, 3859 and 3589 respectively for training, validation and testing. We first used FER2013 as a baseline to validate the results of the architectures and then train the best models on the dataset FERLoopAI.

Figure 1: Example images from the FERLoopAI dataset with their respective class.

2.2 Architectures

The experiments involved different types and generations of deep neural architectures. We mostly trained small networks from scratch using first the FER2013 to validate the results based on the validation accuracy and then on the FERLoopAI dataset. While bigger networks are finetuned from a pretrained model. All the architectures employ batch normalization [2] for a faster and more accurate training.

- **VGGNet** [3]. A simple network that achieved one of the best result on the FER2013 dataset. It is composed by 4 blocks, where each block is composed by three layers convolution-convolution-maxpooling. At the end of the blocks, there is a fully connected hidden layer (1024) followed by the output (7). We modified the network used in [3] by adding a dropout layer to the end of each block to the hidden fully connected layer. We trained the modified network from scratch.
- **ResNet18** [4]. The first deep neural network dealing with the *degradation problem*: when the network depth increases, the accuracy gets saturated, (not because of overfitting) which hampers the increase in depth of neural architectures. ResNet solves the problem by introducing a residual connection which produces a loss function that is easier to train [5]. We modified the original ResNet similarly as in [6]: the network is without the initial convolution-pooling block, it is more narrow with 256 feature maps on the last residual block and uses a dropout layer (0.2) after each residual block. We trained the modified network from scratch.
- **DenseNet121** [7]. This network uses the computed feature maps as input for all the subsequent layers, connecting them using the idea of residual connection. DenseNet strengthens feature propagation and encourage feature reuse. We modified the DenseNet by removing the initial pooling layer, decreasing the growth-rate to 24 and using the dropout layer (0.2) after each block. We trained the modified network from scratch.

- **ViT** [8]. A Transformer based deep neural network. The architecture is the original vanilla Trasformer with small tweaks to feed the images to the model. It achieves results comparable to the best models on the dataset ImageNet, demonstrating the efficacy of attention-based model also for computer vision. For our experiment, we finetuned a pretrained version *vit_base_patch16_224* [1] on ImageNet.
- **MobileNetV3** [9]. Latest version of MobileNet architecture. It is an efficient model targeting low computation architectures. The architecture of MobileNetV3 is found by using network architecture search (NAS), which is a method that automatically design a neural network using an optimization algorithms. We trained the network from scratch.
- **EfficientNet-B3** [10]. A new family of small convolutional networks that have a faster training speed and high accuracy. The architecture is found by using network architecture search (NAS) by jointly optimize training speed and parameter efficiency. We trained the network from scratch.

2.3 Training

We used the same setup to train all the listed models. Starting from the data, the FERLoopAI dataset was divided into two splits of size 80% and 20% for training and validation. The splits were fixed, providing all the models the same input for training and validation. This allowed a fair comparison between the models. Furthermore the splits were stratified, preserving the same class distribution.

The models were trained by optimizing the cross-entropy error using the SGD optimizer with the Nesterov momentum set at 0.9. For the other hyperparameters we used a similar setup as in [6]. The learning rate, batch size, and weight decay were set respectively at 0.01, 128, and 0.0001. The ReduceLROnPlateau scheduler was used to automatically monitor and reduce the learning rate by 0.5 if the validation accuracy does not improve for the latest 10 epochs. Each model was trained for up to 100 epochs. For each epoch during the training, we randomly augment the data by applying in sequence the following transformation to the training set: translation ($\pm 20\%$), scaling ($\pm 20\%$), rotation ($\pm 10\%$) and horizontal flip. Each of these transformation was randomly applied with a probability of 50%. After completed the training, we load the best model for each architecture based on the validation accuracy, and improved the models more by finetuning. We finetuned the models with an additional 20 epochs using the same optimizer with a fixed learning rate of 0.001.

3 Results and Discussion

3.1 Quantitative Results

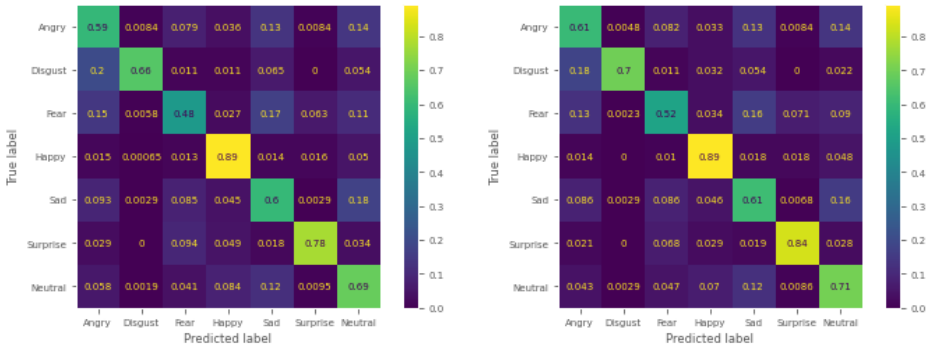
During the initial experiments while validating the models on the dataset FER2013, we noticed two interesting results: 1) Models trained from scratch perform better than the pretrained models on ImageNet. 2) Modified (narrower and more regularized) models performed better than the original architecture. For the first result, we think the reason may be because the features learned through ImageNet were not significantly helpful for the FER task. For the second result, we believe that given a fixed and small dataset we need also to modify the network architecture in order to make it more “suitable" for the dataset. The improvement in accuracy was significant for both ResNet18 and DenseNet121. ResNet18 obtained an increase of 5%, starting from 67% with the non modified version to 72% with the modified one. While DenseNet121 gained 2.5%, starting from 68% to 70.5%. The only final model that was trained with finetuning from ImageNet is ViT. Due to the scarcity of data in FERLoopAI, it was not possible to train it from scratch.

[1] https://github.com/rwightman/pytorch-image-models

Architecture	Parameters	FER2013 Valid Acc.	FERLoopAI Valid Acc.
Human (Base)	-	65 ± 5	-
VGGNet*	4.7 M	72.6	69.1
ResNet18*	5.1 M	72.2	67.7
DenseNet121*	3.9 M	70.6	67.2
ViT	85.2 M	64.2	63.8
MobileNetV3	5.4 M	-	61.8
EfficientNet-B3	10.7 M	-	66.6
Ensemble5	115 M	-	**71.1**

Table 1: Compare all the best classifier obtained by the model selection. * indicates the modified (narrower and more regularized) version of the original models. FER2013 models have not been finetuned.

By analyzing the results from Table 1, we see that almost all the models surpassed the human baseline reported from [1] on the dataset FER2013. The best performing single network on both datasets is the modified version of VGGNet with an accuracy of 69.1% on FERLoopAI. To further increase the accuracy, we ensembled the best 5 models (excluding MobileNetV3) and averaged their output achieving an even higher accuracy of 71.1%.



(a) Single (b) Ensemble

Figure 2: Confusion matrix of (a) Single model VGG4.0 (b) Ensemble5

The confusion matrix in Figure 2 shows that if a class has more samples, it can help to achieve a better accuracy. In our case the most popular class *Happy* also holds the highest accuracy. This phenomenon may has origin from the class imbalance bias in the dataset. During our experiments, we tried to deal with the problem by training a single model VGGNet and weighting the computed loss with the class weight. The results were similar with the previous versions, therefore we decided to not further explore this direction.

3.2 Qualitative Results

In order to analyze the inside of a deep convolutional neural network, we applied the saliency map technique as used in [3]. By analyzing the propagated gradients on the input pixels, we extracted a visualization of the features learned by the model and assessed its quality. In particular, we checked which areas of the face is highlighted by the model, allowing us to understand better its output decision. From the Figure 4 we can see that the VGGNet is correctly laying its attention on many relevant part of the face: cheeks, nose, mouth, and forehead. Although in some cases, it incorrectly focused on the irrelevant parts such as the hair.

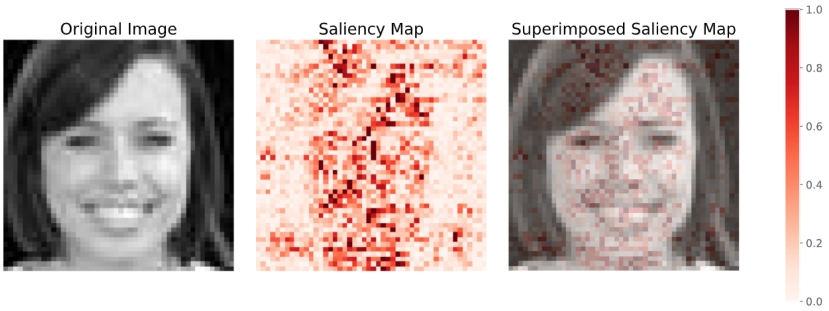


Figure 3: Saliency map of the single model VGGNet.

3.3 Dataset Consideration

The provided dataset FERLoopAI (also FER2013) presents different problems. The dataset is biased to the classes with more samples due to the class imbalance. There are noisy images that do not contain a face or that are not labelled correctly. Even when the labels are correct, some image expressions are difficult to distinguish even for humans. As we can see from the confusion matrix in Figure 2, the class *Disgust* is often misinterpreted as *Angry*, *Fear* with *Sad* and *Neutral* and *Sad* with *Neutral*. This problem is due to the limited nature of the dataset FERLoopAI by imposing a single class output. A better metric to benchmark the FER task is proposed in FER2013+ [11]. This dataset is an extension of the FER2013. Each image is labeled by 10 different taggers, producing a multi-label output which is helpful to classify ambiguous faces. Another approach is to train the machine learning models to predict two continuous output, valence and arousal. These values can then be used to determine the emotion [12].

Figure 4: Predictions of the model VGGNet. The tag above is the label, the tag below is the prediction and its confidence. As we can see, the fourth image is one of the subtle expressions.

As with other machine learning applications, the accuracy bottleneck often resides in the data. We suggest that in order to further improve the system, we should collect and prepare a bigger and better quality dataset. This can be done by processing and combining the public available datasets such as AffectNet [12].

4 Conclusion

In this work, we conducted various experiments with different deep neural architectures for the competition Loop Q Prize 2021. The given dataset FERLoopAI is provided by the organizer. First, we provided a quantitative comparison among the different models based on the validation accuracy. The resulting best single network VGGNet achieved an accuracy of 69.1%. Then, we ensembled the best 5 models and increased the accuracy to 71.1%. Furthermore, we analyzed the quality of VGGNet using a saliency map and found out that the learned features are indeed correct and useful for the task. Lastly, we highlighted the problems of the dataset and proposed two different directions for building a better system in future works: use a better suited evaluation system and use more data by combining the public datasets.

References

[1] Ian J. Goodfellow, Dumitru Erhan, Pierre Luc Carrier, Aaron C. Courville, Mehdi Mirza, Benjamin Hamner, William Cukierski, Yichuan Tang, David Thaler, Dong-Hyun Lee, Yingbo Zhou, Chetan Ramaiah, Fangxiang Feng, Ruifan Li, Xiaojie Wang, Dimitris Athanasakis, John Shawe-Taylor, Maxim Milakov, John Park, Radu Tudor Ionescu, Marius Popescu, Cristian Grozea, James Bergstra, Jingjing Xie, Lukasz Romaszko, Bing Xu, Chuang Zhang, and Yoshua Bengio. Challenges in representation learning: A report on three machine learning contests. In Minho Lee, Akira Hirose, Zeng-Guang Hou, and Rhee Man Kil, editors, *Neural Information Processing - 20th International Conference, ICONIP 2013, Daegu, Korea, November 3-7, 2013. Proceedings, Part III*, volume 8228 of *Lecture Notes in Computer Science*, pages 117–124. Springer, 2013.

[2] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Francis R. Bach and David M. Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 448–456. JMLR.org, 2015.

[3] Yousif Khaireddin and Zhuofa Chen. Facial emotion recognition: State of the art performance on FER2013. *CoRR*, abs/2105.03588, 2021.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society, 2016.

[5] Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 6391–6401, 2018.

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

[7] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 2261–2269. IEEE Computer Society, 2017.

[8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *CoRR*, abs/2010.11929, 2020.

[9] Andrew Howard, Ruoming Pang, Hartwig Adam, Quoc V. Le, Mark Sandler, Bo Chen, Weijun Wang, Liang-Chieh Chen, Mingxing Tan, Grace Chu, Vijay Vasudevan, and Yukun Zhu. Searching for mobilenetv3. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 1314–1324. IEEE, 2019.

[10] Mingxing Tan and Quoc V. Le. Efficientnetv2: Smaller models and faster training. *CoRR*, abs/2104.00298, 2021.

[11] 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 18th ACM International Conference on Multimodal Interaction, ICMI 2016, Tokyo, Japan, November 12-16, 2016*, pages 279–283. ACM, 2016.

[12] Ali Mollahosseini, Behzad Hassani, and Mohammad H. Mahoor. Affectnet: A database for facial expression, valence, and arousal computing in the wild. *IEEE Trans. Affect. Comput.*, 10(1):18–31, 2019.