Facial Expression Recognition via Non-Local ResNet50

Weitian Xue
Duke Kunshan University
Kunshan, China
weitian.xue@dukekunshan.edu.cn

***Abstract*—**

Facial expression recognition, as one of the most popular fields in computer vision, plays an important role in human-computer interaction. In the previous studies, convolutional neural networks were applied frequently for image classification works due to its high efficiency and less training parameters. However, focusing repeatedly on local information, CNN still has its limitations. In order to improve its performance by paying more attention to the potential relations over the whole image, in this content, we tried to add the non-local neural network to ResNetV2-50 to test whether it could gain more accuracy in facial expression recognition tasks. After training based on FER-2013 dataset and KDEF dataset, we compared the results of original ResNetV2-50 model and the non-local version. It turned out that adding the non-local block to higher layers could improve the accuracy for around 1%, but with a non-local block on lower layers would get worse performance. Additionally, the computation complexity would get larger when adding it to lower layers than higher ones.

Keywords—CNN, ResNet, Non-Local, FER

1. INTRODUCTION

Nowadays, Computer Vision (CV) is of great significance under the topic of Artificial Intelligence. As a technology focusing on extracting information from images and videos, CV is widely used in all kinds of aspects. The well-known fields contain self-driving cars, making video games, face recognition, and the like [1].

Among the key problems in CV, image classification is one of the most fundamental and most important tasks. In recent years, with the rapid development of Deep Learning (DL) methods, more and more effective neural networks have been created and put into practice in image classification tasks, making a lot of progress [1]. Through the application of deep neural networks, specifically the Convolutional Neural Network (CNN), image classification have reached a high accuracy in daily tasks.

In order to further improve its performance on the computer-human interactions, facial expression recognition (FER) is raised as a crucial problem in the new era that needs more attention and waits for new developments. If facial expressions, part of the common non-verbal expressions, could be recognized by computers, the information about human emotions could contribute a lot to practical situations, especially for psychological diagnosis [2].

Taking facial images with labels as inputs, FER is aimed to classify specific emotions by analyzing any given facial expressions. According to previous studies, the facial emotions are divided into seven basic categories: happy, sad, fear, angry, disgust, surprise, and neutral [3]. In other words, the core task of FER could be simplified to a classification problem with 7 different classes.

Currently, CNN is most frequently used by researchers for image processing. Instead of taking an image as a long array containing every pixel, CNN uses filters to extract features in a repeated way, which reduces its parameters to train and enhances its performance for image tasks [4].

In 2015, ResNet was proposed, and won the first place in many competitions, because their deep residual framework solved the degradation problem, facilitating building deeper neural networks [5]. After that, ResNet got a version 2 by adding "bottleneck" framework to the original one, which gained even more accuracy [6].

For FER tasks, ResNet also performed well. Bin and Dismas (2021) combined ResNet-50 and CNN to train their FER model and got great performance [7]. However, their dataset only contains 700 images in total, and all of them were taken with the same person. Additionally, the overfitting problem could be an obstacle for further application.

Moreover, in 2018, non-local neural network could give us a new inspiration [8]. Compared with previous CNN models, non-local neural networks does not use the repeated filters to scan images; instead, it considered the whole image to catch the potential relations among local regions. At present, there are few researches that put non-local blocks into practice in the field of FER tasks. Previous researches have combined self-attention module with CNN [9], and the performance turned out to work efficiently. Since self-attention is a specific case for non-local neural networks, we planned to test the general performance of models with the non-local blocks inserted.

To fix this problem and test the performance of non-local neural networks in FER task, we decided to use the ResNetV2 to train a new model and combined it with non-local blocks, based on another official dataset with comprehensive facial expression images. We also compared the results of the two models for further discussion.

2. DATASET

In this paper, we used two datasets: Facial Expression Recognition 2013 (FER-2013) and Karolinska Directed Emotional Faces (KDEF) datasets.

Fig. 1. Random examples taken from FER-2013 dataset

The FER-2013 dataset contains 22968 images for training and 1432 images for test, both of them respectively divided into 7 categories following the 7 basic facial emotions: angry, disgust, fear, happy, neutral, sad, and surprise [3]. Each picture is a 48×48 pixel grid figure of a human face (Fig. 1).

The KDEF dataset contains 4900 pictures in total. The photos are taken through 35 males and 35 females with 7 basic facial expressions, and each expression that is viewed from 5 angles (Fig. 2), to keep its diversity. Each picture is in rgb-mode and 562×762 pixel [10].

3. METHODOLOGY

3.1 Model Building

In this paper, we will build two models, ResNetV2-50 and NonLoalResNetV2-50, and compare their performance after training on FER-2013 dataset.

3.1.1 ResNetV2-50

The first model is built following the instructions in Keras and the paper[5]. Compared with the basic blocks of residual units in the first version of ResNet, ResNetV2 is improved by adding skip connections which could directly propagate signals among units through identity mappings [6].

3.1.1.1 Bottleneck

Taking batch-normalization (BN) and ReLU as "pre-activation" for parameters in the input, the output of the new residual unit is the identity mapping for the addition of input and activated input, which is called "bottleneck" (Fig. 3).

Fig. 2. An instance of KDEF map with females from KDEF dataset

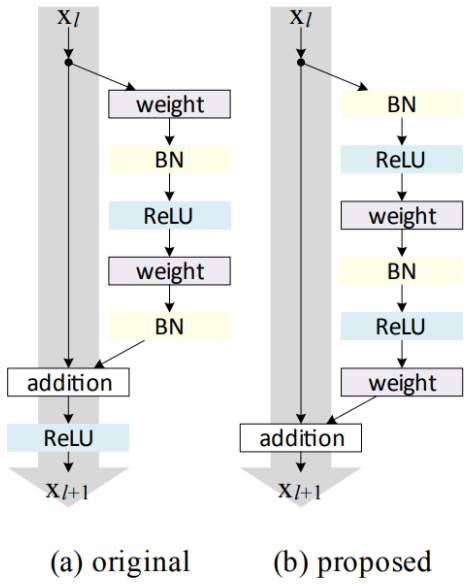

Fig. 3. ResNetV1 residual unit (left) and ResNetV2 residual unit (right) [6]

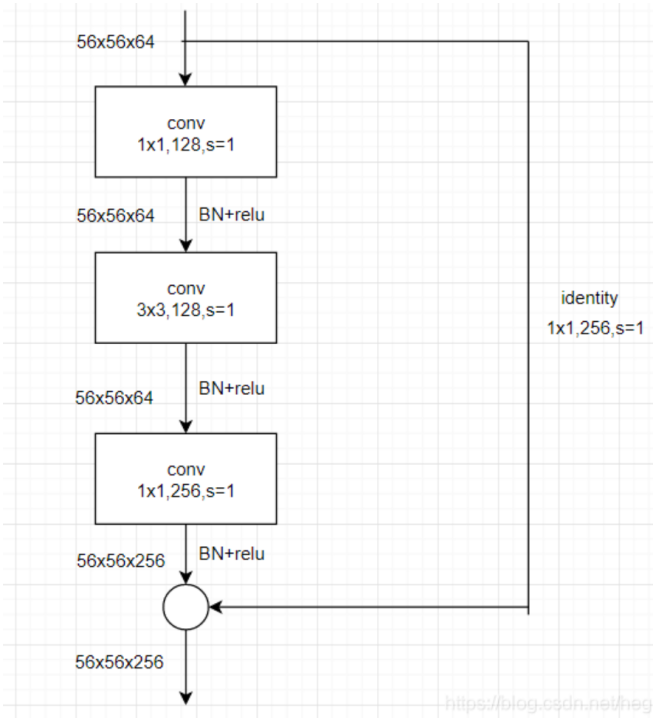

Fig. 5. Detailed structure of a single bottleneck in Stage 1

3.1.1.1 Structure of backbone

The backbone of ResNetV2-50 contains 5 stages (Fig. 4). Stage 0 is the pre-processing of data, including a 7×7 convolution with stride 2, a Batch Normalization, a ReLU activation, and a 3×3 max pooling with stride 2. Stage 1 to stage 4 are all composed of bottlenecks with certain repetitions 3, 4, 6, 3 by default.

Each repeated bottleneck contains three convolutional layers as shown in Figure 2, wrapped up by the proposed residual module in Figure 1. For instance, the bottleneck in the Stage 1 has a 1×1, 3×3, and 1×1 convolutional layers, with an addition combining the original output and identity results (Fig. 5).

3.1.1.1 Classifier

We take the official model ResNetV2-50 in Keras as the backbone. After the backbone, we added a max pooling layer, a dropout layer, and a classifier with “softmax” as activation function.

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix} \times 3$
conv3_x	28×28	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix} \times 8$
conv4_x	14×14	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix} \times 23$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix} \times 36$
conv5_x	7×7	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix} \times 3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

Fig. 4. The structure of ResNet backbone[6]

3.1.2 NonLocalResNetV2-50

The second model took the first one as backbone and added non-local blocks for the layers in stage 1, 2, 3, and 4 respectively.

3.1.2.1 Non-local block

The core for non-local neural networks is the design of non-local blocks. To pay more attention to the relations among distanced information, the non-local block computes "the sum of features at all positions" as the response at one single position [8]. In other words, its receptive field is the whole image instead of the local filters in traditional convolutional layers.

The common formula for non-local algorithm is:

$$y_i = \frac{1}{C(x)} \sum_{\forall j} f(x_i, x_j) g(x_j) \tag{1}$$

The function $f(x_i, x_j)$ represents the relation of any two pixels, while $g(x_j)$ is the feature of the targeted position.

To simplify the computation, $g(x_j)$ could be designed as:

$$g(x_j) = W_g x_j \tag{2}$$

which could be realized through 1×1 convolution in practice.

For $f(x_i, x_j)$, there are four types of relation functions:

a) Gaussian

$$f(x_i, x_j) = e^{x_i^T x_j} \tag{3}$$

b) Embedded Gaussian

$$f(x_i, x_j) = e^{\theta(x_i)^T \varphi(x_j)} \tag{4}$$

c) Dot product

$$f(x_i, x_j) = \theta(x_i)^T \varphi(x_j) \tag{5}$$

d) Concatenation

$$f(x_i, x_j) = \mathrm{ReLU}(w_f^T [\theta(x_i), \varphi(x_j)]) \tag{6}$$

We used the embedded Gaussian as the default option to test in our model.

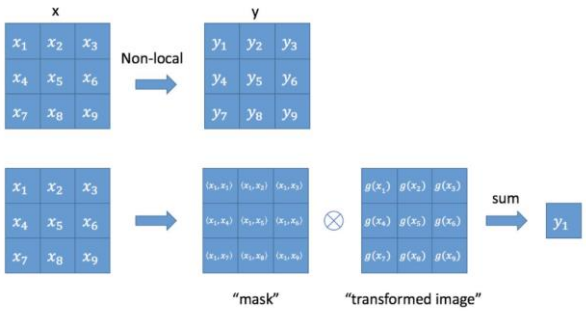

Fig. 6. Visual explanation for non-local block

The non-local block is designed as a convenient plugin that could be inserted into any network structures in need. As such, the size of output of a non-local block is the same as the input (Fig. 6), and it could be trained as a residual block with the equation:

$$z_i = W_z \cdot y_i + x_i \tag{7}$$

3.2 Data pre-processing

3.2.1 FER-2013

The original FER-2013 dataset has been divided into a train set and a test set, with 22968 and 1432 images respectively. Each image is a 48×48 pixels grid picture of human face.

When we tried to fit the images to Non-local ResNetV2-50 model, which has non-local blocks when filters are larger than 512, we found that the original image size (48, 48, 1) would be too small for training. As such, we doubled the image size and used the doubled 96×96 pixels images as training data for Non-local ResNetV2-50 model.

To compare the two models fairly, we also used the resized data for ResNetV2-50 model to create a control group.

3.2.2 KDEF

The KDEF dataset contains 140 directories to store the images from 70 different individuals, including 35 males and 35 females. Each individual has two directories of images since they were shot for twice. In each individual directory, there are 5 images for each expression, viewed from 5 different angles. With 7 expressions, there are 35 images in one directory. In total, there are 4900 562×762 pixel pictures with color.

To facilitate the training and validation, we reorganized the dataset and divide all the photos into 7 directories based on 7 different facial expressions. We maintained the size and color for each image.

3.2.3 Data Augmentation

In order to avoid the over-fitting, we augmented the data both in FER-2013 and KDEF by width shift, height shift, horizontal flip, and rescaling. We set both the width shift range and height shift range to 0.1, and rescaling to 1/255.

4. RESULTS

After the training with batch 64 and 50 epochs, the result of ResNetV2-50 reaches 89.2%.

With the same batch and epochs, the accuracy of NonLocalResNetV2-50 on FER-2013 (Table. 1) and KDEF (Table. 2) were generally improved with non-local blocks on each four stage. However, when the non-local block is added to stage 1 or 2, the rising of validation accuracy would be worse than those when non-local blocks are added to stage 3 or 4.

TABLE I. COMPARASON OF DIFFERENT MODELS ON FER-2013

MODELS	ACCURACY(%)
RESNETV2-50	89.21
NON-LOCALRESNETV2-50 (STAGE 3-4)	90.13
NON-LOCALRESNETV2-50 (STAGE 1-2)	89.82

TABLE II. COMPARASON OF DIFFERENT MODELS ON KDEF

MODELS	ACCURACY(%)
RESNETV2-50	76.02
NON-LOCALRESNETV2-50 (STAGE 3-4)	77.14
NON-LOCALRESNETV2-50 (STAGE 1-2)	76.68

5. DISCUSSIONS

As shown by the results, the non-local blocks have positive impact on the performance of ResNet model in the FER tasks in general, both for grid pictures and colorful ones. However, there are obvious limitations that worth more reflection.

The first limitation is the position to add non-local blocks. The different extent of progress from ResNet with non-local blocks added on lower stages and on higher stages have proved that not all models equipped with non-local blocks would get obvious improvement. Adding non-local blocks to higher stages would give more rise to the accuracy.

The second limitation is the extremely increasing computation complexity resulting from non-local blocks. For ResNet, the trainable parameters were turned from 23,527,431 to 36,131,847 and 48,714,759 respectively when adding non-local blocks for stage 3 and 4. Although adding it to lower stage would lead to less rise to computation complexity, the performance would be worse.

When putting it into practice, we need to find a balance between the computation complexity and the expected performance.

Our model only takes the ResNet50 as backbone, and did not test the other three relation functions of non-local blocks. In the future, further researches on the non-local blocks with other models and relation functions need to be done in order to testify its overall performance on facial expression recognition.

6. DISCUSSION

Based on the ResNet50 model and Non-local Neural Networks, we tried to modify the original ResNet50 through adding non-local blocks and testify its performance on FER tasks with datasets containing grid pictures, FER-2013, and colorful pictures, KDEF, respectively. During the experiment, we found that the modified model performed better than the original model in general, but the progress would differ for distinct position to add non-local blocks: adding the non-local blocks to stage 3 or 4 would gain more increase in its validation accuracy than adding them to stage 1 or 2. Considering the extent of improvement and the large amount of rising computation complexity, more researches need to be done to find a balance between maintaining a high-level performance and relatively low complexity.

REFERENCES

[1] M. Modzelewska-Kapituła and S. Jun, “The application of computer vision systems in meat science and industry – A review,” *Meat Science*, vol. 192, p. 108904, Oct. 2022, doi: 10.1016/j.meatsci.2022.108904.

[2] D. Canedo and A. J. R. Neves, “Facial Expression Recognition Using Computer Vision: A Systematic Review,” Applied Sciences, vol. 9, no. 21, p. 4678, Nov. 2019, doi: 10.3390/app9214678.

[3] I. J. Goodfellow et al., “Challenges in representation learning: A report on three machine learning contests,” Neural Networks, vol. 64, pp. 59–63, Apr. 2015, doi: 10.1016/j.neunet.2014.09.005.

[4] M. Valizadeh and S. J. Wolff, “Convolutional Neural Network applications in additive manufacturing: A review,” *Advances in Industrial and Manufacturing Engineering*, vol. 4, p. 100072, May 2022, doi: 10.1016/j.aime.2022.100072.

[5] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” Dec. 2015.

[6] K. He, X. Zhang, S. Ren, and J. Sun, “Identity Mappings in Deep Residual Networks,” Jul. 2016.

[7] C. Pramerdorfer and M. Kampel, “Facial Expression Recognition using Convolutional Neural Networks: State of the Art,” Dec. 2016.

[8] X. Wang, R. Girshick, A. Gupta, and K. He, “Non-local Neural Networks,” Apr. 2018.

[9] J. Li, K. Jin, D. Zhou, N. Kubota, and Z. Ju, “Attention mechanism-based CNN for facial expression recognition,” Neurocomputing, vol. 411, pp. 340–350, Oct. 2020, doi: 10.1016/j.neucom.2020.06.014.

[10] Lundqvist, D., Flykt, A., & Öhman, A. “The Karolinska Directed Emotional Faces - KDEF (CD ROM).” Department of Clinical Neuroscience, Psychology section, Karolinska Institutet, ISBN 91-630-7164-9.