Comparison of Different Depth of Convolutional Neural Network Deep and shallow CNN comparison based on FER-2013

Hongxin Song

University of California, San Diego San Diego, United States

h3song@ucsd.edu

Abstract. The purpose of this paper is to cross-compare the different nature of the decision process of convolutional neural networks (CNNs) of different depths, as well as the characteristics of the ensemble model. The Facial Expression Recognition (FER) problem is selected as the carrier of this experiment. The dataset chosen is FER-2013, one of the most challenging FER datasets due to its complicated and natural contents. This research first trained four models that have different architectures using FER-2013. One is the shallow convolutional neural network, and the other three are deep pre-trained CNNs, including ResNet50 with weights from ImageNet, VGG16 with weights from ImageNet, and VGG16 with weights from VGGFaceNet. Then, by using a gradient-based Class activation mapping technique as a visualization technique, this experiment successfully displayed the different decision-making processes of the selected models and illustrated how the depth of the neural network influences the feature extraction process. This work further experimented with convolutional neural networks(CNNs) having different depths by varying ensemble combinations. Finally, all three-model combinations were tested, and the experiment results show that models that are more different in architecture would result in a better ensemble performance than those with a similar architecture. Thus, one of the inspirations we could get from this work is that models with dramatically different architectures could earn a more remarkable improvement for future ensemble models.

Keywords: Component; convolutional neural networks; model comparison; ensemble; Facial Expression Recognition.

1. Introduction

In the current machine learning community, pre-trained deep neural networks are fine-tuned and utilized to solve different problems. Since training a neural network from scratch is computationally consumptive and it is hard to guarantee a promising result, adapting pre-trained neural networks seems to be a better solution. This method is especially popular in computer vision due to the high computational cost of training computer vision models. In order to examine transfer learning on computer vision problems, this work utilized multiple pre-trained neural networks, including VGG16 [1] and Resnet50 [2], on Facial Expression Recognition (FER) problems.

Facial expression recognition is an essential topic in computer vision since human emotion is an indicator of a person's physical and psychological conditions. Being able to identify human emotion in real-time correctly can help us with communication and health issues. Using computer vision models to identify emotion facilitates progress in numerous fields such as human-computer interaction and mental healthcare. The dataset used is FER-2013 [3], the most common image dataset in CNN-based FER and one of the most extensive publicly available datasets in this field. It originates from image data on Google and can well represent the naturalistic scenarios where FER may be applied in real life since it contains facial data under various conditions. The human performance on this dataset is approximately 65.5 % [3].

Although pre-trained deep neural networks are often better solutions compared to shallow new neural networks, there is no guarantee which one is better. The generalization error of pre-trained neural networks is uncontrollable. This paper uses deep convolutional neural networks as facial feature extraction tools for the FER problem. In order to achieve minimal generalization error, this

work also trained a shallow neural network. Different model architectures could have different decision-making processes and generate different results. By assembling different models, this experiment could achieve performance improvement quickly and effortlessly. The combined model would have the most optimal feature extraction ability. This paper also further examined the different decision-making processes of shallow convolutional neural networks and deep convolutional networks using gradient-weighted class activation mapping (Grad-CAM).

This paper intends to compare the performance of shallow and deep convolutional neural networks, discovering how their decision-making processes are different from each other by visualizing gradient-weighted class activation mapping of their last convolutional layer. It also intends to examine which models are suitable for ensemble, further discovering how different models extract features differently.

2. Related Work

2.1. Convolutional Neural Networks

In computer vision, CNN has always been a good solution for various problems. The emergence of deep convolutional neural networks like VGG [1] and Resnet [2] makes some previously challenging computer vision problems easier to solve. With the growth of technology, the increased computing power enables deep convolutional neural networks that are trained using extensive data to be applicable. Since training convolutional neural networks has an extremely high computational cost, transfer learning appears to be a sound solution. By adapting pre-trained deep convolutional neural networks like Imagenet, researchers were able to utilize old models for new problems quickly.

One computer vision problem that is hard to resolve is facial expression recognition. As researchers worked to find a solution, FER-2013 [3] was generated to train a usable and generalizable facial expression recognition model. It is one of the most challenging facial expression recognition datasets since it contains not only human pictures but also paints as well. As mentioned earlier, Goodfellow et al. prove that the human accuracy for this dataset is only around 65±5% [1]. Later, researchers could apply more complicated neural networks to FER-2013 and make significant progress. Recently, researchers successfully trained a single VGGNet with an accuracy of 73.28% [4] using only the FER-2013 dataset. Similarly, a state-of-the-art Resnet model obtained an accuracy of 72.4% without any add-on training data [5]. These state-of-the-art deep CNNs were able to achieve accuracies of over 72%. However, CNN with significantly fewer convolutional layers was also able to obtain an accuracy of 72.16% with only hyper parameter tuning technique [6]. The pre-trained deep convolutional neural networks failed to beat the shallow convolutional neural network with a large margin.

2.2. Ensemble

All the above single networks could not achieve an accuracy of over 75%. Currently, the model with the best performance is the ensemble of 6 networks, including ResMaskingNet [7] experimented with by Goodfellow et al. There has been much evidence that ensemble models together are one of the simplest ways of gaining model improvement. Moreover, the ensemble of different models minimizes the generalization error to the most considerable extent. For example, the ensemble of different CNN networks would help the model extract the related facial features with higher precision for facial expression recognition problems. Khanzada et al. were able to achieve the highest accuracy of the single model- ResMaskingNet at 74.14% by adding extra training data. However, all the single network models they used in the ensemble failed to achieve an accuracy of 75%. With the simple soft-voting ensemble of 7 models, they obtained a test accuracy of 76.82%, beating their highest single model by over 2.5% [7].

Ensemble models have always been a good way of improving model performance. Due to the high computational cost and extensive running time of neural networks, traditional bagging and bootstrapping methods are unsuitable for neural network ensembles. Unlike Goodfellow et al., who

ensemble seven deep neural networks [7], this research aimed to ensemble networks with different depths to generate a decent comparison of different depths of neural network models and how this would affect the ensemble process.

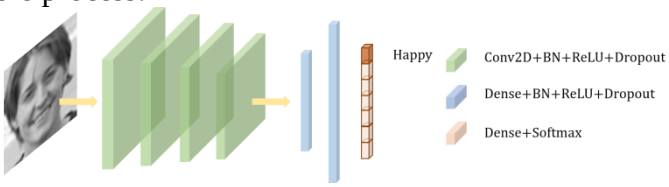

Figure 2. Architecture of Shallow CNN

2.3. Experiments

Since our goal is to compare and evaluate convolutional neural network models with different architectures, the experiment started with training different models to around the same accuracy level.

2.4. Simple Convolutional Neural Network

In order to compare different model architectures and their performances, this research first constructed a non-deep neural network. Then, the network architecture can be demonstrated in figure.1. It contains only four two-dimensional convolutional layers, each with a batch normalization layer and a dropout layer with 25% after activation. The final dense layer of this model has a softmax activation function with seven channel outputs. The model was trained for a total of over 100 epochs till it fully converged. The optimizer chosen to use is adam [7]. The learning rate is initially set to 0.001 and a reducing learning rate scheduler is added to stabilize the training process.

2.5. Deep Convolutional Neural Networks

For the deep neural network models portion, VGG16 and Resnet50 are selected as examples. These deep CNN models were used as our backbones and with several denser layers on top of them. All three models are added:

A dropout layer with a rate of 0.5
A dense layer with 4096 nodes
A dropout layer with a rate of 0.5
A dense layer with 2048 nodes
The final output dense layer with seven nodes

In total, three deep CNN models were trained, VGG16 pre-trained by ImageNet [8], VGG16 pre-trained by VGGfaceNet [9], and ResNet pre-trained by ImageNet [8]. In addition, all the models are trained with frozen convolutional layer weights to obtain quicker and easier extract facial features.

VGG16 model contains 16 layers, including 13 convolutional layers and 3 fully connected layers, which has 10 more convolutional layers compared with our simple CNN model [1]. The VGG16 network distributes a total of five pooling layers under different convolutional layers, and it use the maximum pooling layer. In VGG16, each convolutional layer contains 2 to 3 convolutional operations. The size of the convolution kernel is 3x3 with a convolution stride of 1, and the pooling kernel is 2x2 with a stride of 2. Its multi-nonlinear layer structure enables it to maximize its depth to learn complicated patterns.

ResNet50 is much deeper even than VGG16 [2]. As the name has, it contains in total 50 layers, 48 Convolution layers followed by one MaxPool and one Average Pool layer. It started with one convolutional layer with a kernel size of 7x7 and kernel number of 64. Then, it is followed by a max-pooling layer with a stride size of 2. The first convolutional block contains three convolutional layers: one with 64 kernels and a size of 1x1, one with 64 kernels and size 0f 3x3, and the last one with 256 kernels and size of 1x1. This block is repeated three times. The second block contains:

A layer with 128 kernels and size of 1x1,
A layer with 128 kernels and size 0f 3x3, and
A layer with 512 kernels and size of 1x1.

This block is repeated four times, giving us 12 layers. The third block has the same structure as the second but twice the kernel number. This block is repeated six times, in a total of 18 layers. Like the former structure, the fourth block has twice the kernel number of the third block and is repeated three times.

2.6. Visualizations

In order to examine different models' decision-making processes, this research utilized Grad-Cam to display which part of the face picture the model used to make its predictions [10].

Class activation mapping (CAM) estimates which area of the image the model mainly is used when making a class-specific prediction. To discover different decision process of different convolutional neural network models. This paper first evaluates all the models based on the test data and generate four lists of predictions. Then, it chooses the pictures correctly identified by all four of them and use their last convolutional layer weight of their correctly labeled class to generate the gradient-based class activation maps.

2.7. Ensemble

The final step of experimenting and comparing different convolutional model architecture is ensemble. There are four models for the ensemble; thus, there are four combinations of a three-model ensemble and one combination of a four-model ensemble.

This experiment started with the four-model ensemble with a simple average technique. Then, the four trained models were used directly, and their outputs were averaged to obtain the new probability list. Finally, this experiment evaluates test set results based on that probability list. To further examine the relationship between these models, this work tried the three-model ensemble one by one. Hypothetically, the smaller the improvement, the more negligible difference among the decision-making processes of the models.

After trying out all the unweighted ensembles, this experiment introduced the weighted ensemble to boost the ensemble model performance further. Similar to the simple average, the weighted ensemble averages the tensor output but averages the output based on specific weights. First, a new linear layer was constructed to obtain all four models' final output. Then, this model randomly initializes the weight of this layer. Just like how previous models are trained, the weights of these models would be optimized through gradient descent.

Table 1. Model Performance Comparison Table

Model	Test Accuracy
Shallow CNN	**68.39%**
VGG16-ImageNet	68.32%
VGG16-VGGfaceNet	67.65%
ResNet50-ImageNet	67.31%

The optimizer chosen for this model is Nadam [11], a combination of the Adam algorithm and Nesterov momentum. The weights of ensemble were finalized after 50 epochs.

3. Results

3.1. Model Performance Comparison

Surprisingly, the highest performance this experiment obtained is from our shallow neural network model. As described in Table I, the shallow CNN model obtained a test accuracy of 68.39%. The VGG16 model achieved 68.32% accuracy on the test set, and the VGGface model achieved 67.65% accuracy. The ResNet50 has the relatively lowest accuracy at 67.31%. It can be said that the pre-trained deep convolutional neural networks do not perform best for this dataset. One hypothesis made is that this is due to the small input size of the data. In order to generate a workable shallow CNN

model and to lower the computational cost, this experiment kept the images in their original sizes as 48x48, just filling the RGB channel. The input size is relatively small compared to what the deep CNN models were previously trained on, which have an input size of (255,255,3). After the deep convolution process, much helpful information could be lost due to the small input size and long convolutional process. Thus, the shallow CNN beats those deep CNN models under this particular circumstance.

3.2. Visualization Comparison

As mentioned in the previous section, this work visualized what partly contributes to the model prediction decision using Grad-CAM. To generate a sound comparison, several correctly labeled classification results from our shallow CNN model and VGG16-VGGface [9] model are selected as samples. From Fig.1, it can be proven that our shallow CNN model is generally influenced by a larger area of the original image, whereas our VGGFace model is concentrated on smaller areas. One interesting observation is that they tend to be affected by dramatically different areas. Our previous thought is that VGGface would make predictions on the smaller area but should resemble the areas used by shallow CNN to make predictions. However, in most cases, VGGface tends to focus on the area that shallow CNN considers less effective for prediction results.

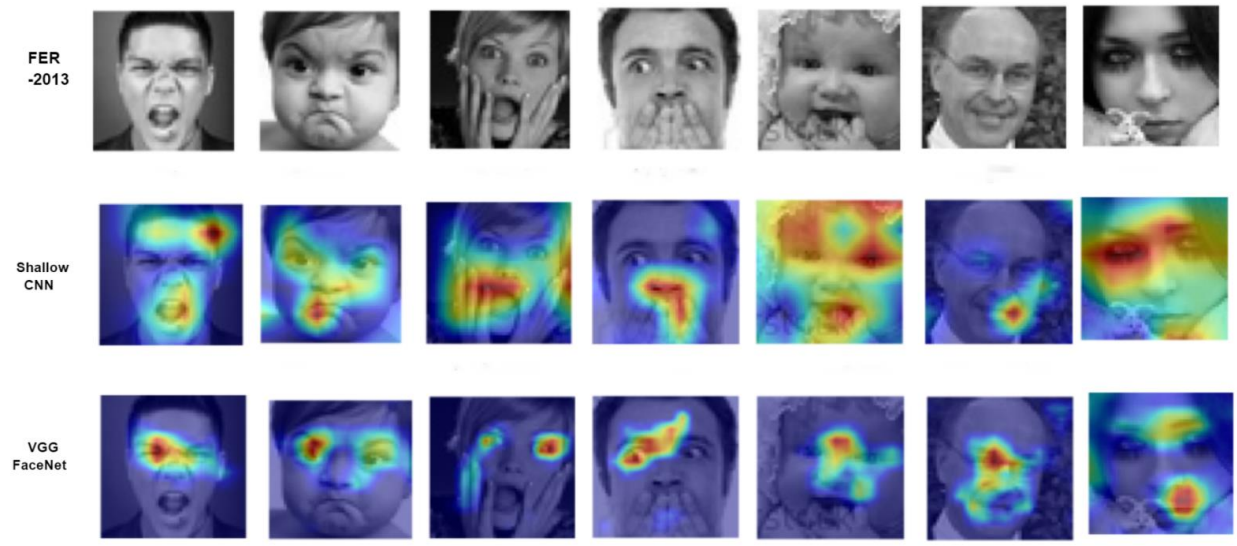

Figure 2. Example of Grad-CAM Results of Shallow CNN and VGGFace. (Second Row-Shallow CNN; third row-vggFace)

Our previous thought is that VGGface would make predictions on the smaller area but should resemble the areas used by shallow CNN to make predictions. However, in most cases, VGGface tends to focus on the area that shallow CNN considers less effective for prediction results. Like the first four graphs in Fig.4, VGGFace focuses on a less noticeable area like eyes, while the shallow CNN gets its prediction mainly based on mouths. The shallow CNN model obtained its prediction with a larger image area. The fifth graph in Fig.4 can illustrate

This comparison. Although VGGFace relies on a smaller area, it tends to locate facial features more precisely. On the other hand, in some areas, the shallow CNN used to make predictions has nothing to do with facial expression. The deep convolutional neural network structure allows VGGFace to make more accurate facial feature extraction but also loses a considerable amount of important information [9]. Reversely, the shallow CNN keeps almost all the essential features but is also worse at extracting facial features precisely.

3.3. Ensemble Comparison

Unlike the accuracy comparison results of the single model just presented, the results of the ensemble comparison are very close to what would be expected. In the article above, it was mentioned that the more structural differences in the models themselves, the more likely the ensemble results would show greater improvement. This is because different model structures lead to very different decision processes and are more likely to complement each other, thus contributing to better ensemble results. On the other hand, if the overall structures of the two models are too similar, their decision processes will be somewhat similar to each other, making their ensembles somewhat more unvarying.

As Table II. displayed, for the three-model combinations, the combination with the worst performance is the combination of VGG16 with the weights of VGGFaceNet, VGG16 with the weights from ImageNet, and Resnet50 with the weight from ImageNet. These three are all deep neural networks that use transfer learning. Their shared deep convolutional feature could be why they are the worst ensemble model.

For the four-model ensemble, test accuracy of 71.496% is achieved using the simple average method, which is 3 percent higher than the best-performed singel CNN. Adding weights to the average technique achieves final test accuracy of 71.85%, which beats the shallow CNN by 3.5%. Table III shows the weights of the final ensemble model. It is not difficult to find that VGGFace model is the most important model, occupying more than fifty percent of the share. The next model is the shallow CNN, which accounts for 0.32% of the total. The remaining two models account for relatively less, possibly because they and VGGFace model both have deep structure, making the decision process similar.

4. Conclusion

This paper resulted in a model with an accuracy of 71.85% trained using only the FER-2013 dataset. The model comparison process found that the deeper the model is, the more likely it is to lose helpful information, but the information it identifies will be relatively more accurate. In continually testing different ensemble combinations, there is evidence that the more different the models themselves are structured, the more likely their combination is to yield an ensemble model with better results. Their structural differences make their decision processes more complementary to each other, thus enhancing the effectiveness of the ensemble model. In the future, if there are other problems that can be applied to ensemble, researchers should first consider training models with different structures but similar accuracy scores.

Table 2. Ensemble Performance Comparison Table

Model	**Test Accuracy**
Shallow CNN	68.39%
VGG16-ImageNet	68.32%
VGG16-VGGfaceNet	67.65%
ResNet50-ImageNet	67.31%
Shallow CNN+VGGFace+Resnet50	**71.204%**
Shallow CNN+VGGFace+VGG16	70.716%
Shallow CNN+VGG16+Resnet50	70.605%
VGGFace+VGG16+Resnet50	70.577%
Four Combined (Unweighted)	71.496%
Four Combined (Weighted)	**71.845%**

Table 3. Weights of the Four Model Ensemble

Shallow CNN	VGG16-VGGFace	VGG16-ImageNet	Resnet50-ImageNet
0.519	0.321	0.165	0.16

References

[1] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In International Conference on Learning Representations, 2015.

[2] 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, 2016.

[3] I. Goodfellow, D. Erhan, P. Carrier, A. Courville, M. Mirza, B. Hamner, W. Cukierski, Y. Tang, D. Thaler, D. Lee, et al. Challenges in representation learning: A report on three machine learning contests. In International conference on neural information processing. Springer, 2013.

[4] Y. Khaireddin and Z. Chen. Facial emotion recognition: State of the art performance on fer2013. arXiv preprint arXiv:2105.03588, 2021.

[5] C. Pramerdorfer and M. Kampel. Facial expression recognition using convolutional neural networks: state of the art. arXiv preprint arXiv:1612.02903, 2016.

[6] A. Vulpe-Grigoraşi and O. Grigore, "Convolutional Neural Network Hyperparameters optimization for Facial Emotion Recognition," 2021 12th International Symposium on Advanced Topics in Electrical Engineering (ATEE), 2021, pp. 1-5, doi: 10.1109/ATEE52255.2021.9425073.

[7] I. J. Goodfellow, D. Erhan, P. L. Carrier, A. Courville, M. Mirza, B. Hamner, W. Cukierski, Y. Tang, D. Thaler, D.-H. Lee, Y. Zhou, C. Ramaiah, F. Feng, R. Li, X. Wang, D. Athanasakis, J. Shawe-Taylor, M. Milakov, J. Park, R. Ionescu, M. Popescu, C. Grozea, J. Bergstra, J. Xie, L. Romaszko, B. Xu, Z. Chuang, and Y. Bengio, “Challenges in representation learning: A report on three machine learning contests,” arXiv.org, 2013.

[8] J. Deng, W. Dong, R. Socher, L. -J. Li, Kai Li and Li Fei-Fei, "ImageNet: A large-scale hierarchical image database," 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 248-255, doi: 10.1109/CVPR.2009.5206848.

[9] O. M. Parkhi, A. Vedaldi, and A. Zisserman. Deep face recognition. In British Machine Vision Conference, 2015.

[10] R. R Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In Proceedings of Science, 1989.

[11] L. Fan and Z. Long, "Optimization of Nadam algorithm for image denoising based on convolutional neural network," 2020 7th International Conference on Information Science and Control Engineering (ICISCE), 2020, pp. 957-961, doi: 10.1109/ICISCE50968.2020.00197.