Natural Scenes Image Classification Using CNN and Transfer Learning

Yachen Li
Department of Analytics
Georgetown University
Washington, D.C., U.S.A
yl1062@georgetown.edu

Yating Liang
Department of Analytics
Georgetown University
Washington, D.C., U.S.A
yl1138@georgetown.edu

Xinyao Mo
Department of Analytics
Georgetown University
Washington, D.C., U.S.A
xm92@georgetown.edu

Yihan Zhou
Department of Analytics
Georgetown University
Washington, D.C., U.S.A
yz740@georgetown.edu

I. Abstract

Deep learning is the booming field for researchers since the techniques have the capability to overcome the drawbacks of already used traditional algorithms. It is worth emphasizing that deep learning is very efficient on analyzing and understanding images and has been widely used. However, image classification is a complex process and the performance could be affected by many issues. This project examines the practices, performance and problems of different classification techniques. Finally, the expected future research direction of the network has been discussed.

II. Introduction

Deep learning is a very active field in machine learning and artificial intelligence. Human beings can capture and identify objects with their eyes, while computers might be able to do it in another way. Due to the development of deep learning, computer vision is now widely used for analyzing and understanding digital images. Previous literature demonstrates that deep learning gains excellent performance in image classification of remote sensing.

The image classification accepts the given input images and produces output labels from a fixed set of categories. This is one of the core problems in computer vision and deep learning, which have received a lot of attention for the past few decades since it has a large variety of practical applications in real life. For example, face recognition can be embedded in the phone for face-to-unlock technology. In healthcare industries, deep learning is used to label whether the patient has Alzheimer's Disease or not from his MRI image.

In this project, we explore the image classification problems with theImage Classification Dataset from Kaggle. Classic computer vision models, including convolutional neural network, VGG16, ResNet and EfficientNet are trained and applied to this dataset, with image features as input and labels (‘building’, ‘forest’, ‘street’, etc.) as output. We evaluated the performance of our models using runtime, accuracy and loss on both training and testing dataset.

III. Related Work

CNN

A Convolutional Neural Network is a deep learning algorithm, most commonly applied to analyzing visual imagery. In order to categorize large images, the CNN is utilized to preprocess images for dimensional reduction [1]. Through convolution and pooling, an image can be reduced into its essential features, and considers the relationship between these features, then uses them to understand and classify the image. Figure 1 shows a traditional CNN structure, there may be multiple activation and pooling layers, depending on the CNN architecture.

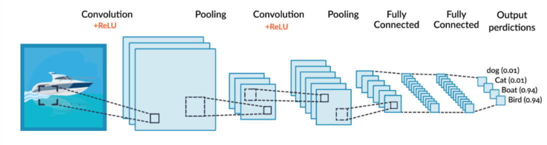

Figure 1: CNN for Image Classification

VGG16

VGG16 is a convolutional neural network architecture proposed by K. Simonyan and A. Zisserman from the University of Oxford in the paper "Very Deep Convolutional Networks for Large-Scale Image Recognition" [2]. For VGG16, they did not use a large number of hyper-parameters. Alternatively, they focused on having convolution layers of 3x3 filter with a stride 1 and used the same padding and maxpool layer of 2x2 filter for stride 2. It follows this arrangement of convolution and maxpool layers consistently throughout the whole architecture [3]. The structure of VGG16 is shown below in figure 2.

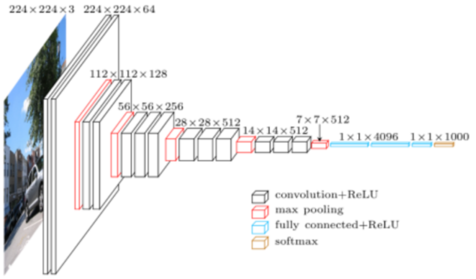

Figure 2: Architecture of VGG16 [3]

ResNet-50

Residual Networks-50 is a convolutional neural network that is 50 layers deep. The fundamental breakthrough with ResNet was it allowed us to train extremely residual networks with 50 layers successfully to achieve less error. In residual learning, we try to learn some residual rather than features directly. ResNet does this using shortcut connections that directly connect input of the nth layer to some (n+x)th layer. It has proved that training this form of networks is easier than training simple deep convolutional neural networks and also the problem of degrading accuracy is resolved [4]. Figure 3 shows the procedure.

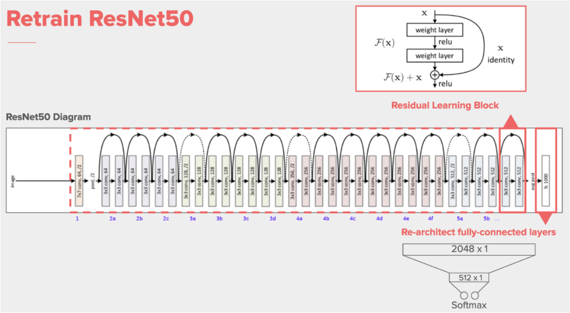

Figure 3: Architecture of ResNet-50

EfficientNet

Model scaling method that uses a simple yet highly effective compound coefficient to scale up CNNs in a more structured manner. Unlike conventional approaches that arbitrarily scale network dimensions, this method uniformly scales each dimension with a fixed set of scaling coefficients. And this set of models, called EfficientNets, which superpass state-of-the-art accuracy with up to 10x better efficiency [5].

IV. Dataset

The primary image dataset we used is the data of natural scenes around the world. It is from Kaggle which contains around 25k images of size 150x150 distributed under 6 categories(“buildings”, “forest”, “glacier”, “mountain”, “sea”, “street”). Meanwhile, the dataset is separated into the train, test and prediction small datasets. There are around 14k images in Train, 3k in Test and 7k in Prediction. And we also display the distribution of six categories by pie chart. It is shown that each category is almost evenly distributed which is helpful for our research.

V. Methods

Models: The project includes both traditional CNN models and transfer learning models for the dataset. For the CNN model, we applied a simple 2-layers model with flatten layers. For transfer learning methods, we explored into the VGG16, ResNet50 and EfficientNetB7 models with inbuilt pretrained weights from the Imagenet. All images are resized to fit into each model.

Evaluation: Model loss and accuracy are both used to evaluate the model performances. Line charts along epoch are given both on training data and testing data, which can give some information on the overfitting identification and model robustness problem. Besides, runtime is also given as a measure of model cost.

VI. Results

The plot of the training process for all models is shown below as well.

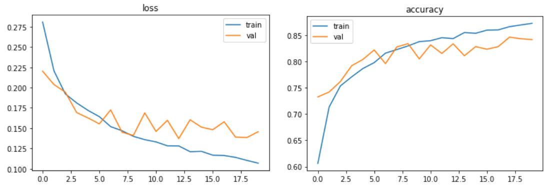

Figure 4: CNN Model Training Process

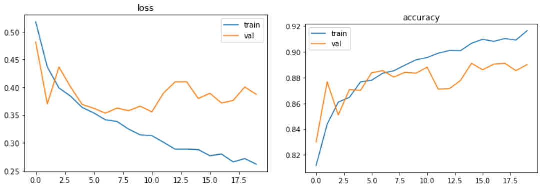

Figure 5: VGG16 Model Training Process

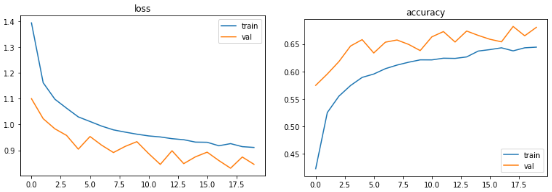

Figure 6: ResNet50 Model Training Process

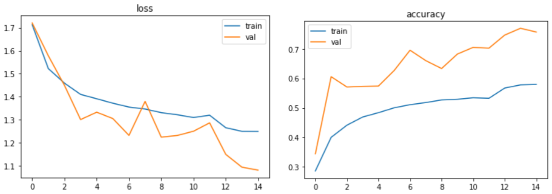

Figure 7: EfficientNetB7 Training Process

The below table summarizes the performance of all four models.

	Test Loss	Test Accuracy
CNN	0.1456	0.8417
VGG16	0.3872	0.8900
ResNet50	0.8456	0.6803
EfficientNetB7	1.0815	0.7570

Table 1: The Result Table

For the CNN model, the structure contains 7 layers with a kernel size of three. It stacks a convolution layer, follows with a pooling layer, and repeats this pattern 2 times. Then, it is transited into a fully-connected layer, and added to the last fully-connected layer, which holds the output for 6 classes. For the VGG16 model, it has some pre-trained layers and weights. We unfreeze the last block convolution layers of the model and use softmax activation function to hold the output and achieve an accuracy of 0.89 with a loss of 0.3872. For the ResNet-50 model, it also has pre-trained layers with 50 layers deep. We retrain and modify the last layer then achieve a test accuracy of 0.6803 with a loss of 0.8456. For the EfficientNet model, it has an accuracy of 0.7570 and the highest loss of 1.0815.

.

VII. Discussion of Results

The test accuracy for CNN model reaches 0.8417 and it has the lowest test loss of 0.1456. The performance is relatively good, but it might have some overfitting. Since on average for each class the datasets have around 2300 images in the training set, there are chances that it will overfit the training data for the simple convolutional neural network. For the VGG model, it achieved the highest accuracy of 0.89 and it has the best performance among all four models. For the ResNet50 model, unlike VGG, it relies more on micro-architecture modules and the residual learning. The lowest test accuracy for the ResNet model might be due to this reason. Since the datasets containing many buildings and street images, it has higher similarity and less residual errors within these two classes. The ResNet model tends to incorrectly classify “street” as “building” or the other way around. For the EfficientNet model, the test accuracy is 0.7570 as the second lowest. For this model, we might try to tune the model more by unfreezing a number of layers and refitting the model using different learning rates.

VIII. Conclusions

The project starts from 25K images input, then performs image transformation, and fits and compares different models. Based on the results, we can conclude that CNN performs well and VGG16 performs best among all transfer learning pre-trained models. Finally, the VGG16 model gets test accuracy of 0.89. In the future, the team will focus on finding out more hyperparameter tuning techniques, such as reducing learning rates, using smaller batch size, and partially adjust the freezing/unfreezing of layers. We will also try for the best performance of the transfer learning ResNet and EfficientNet model and testing other models, such as InceptionV3 and Mask R-CNN model.

References

[1] Gao, Jingyu, et al. "Natural Scene Recognition Based on Convolutional Neural Networks and Deep Boltzmannn Machines." *2015 IEEE International Conference on Mechatronics and Automation (ICMA)*, 2015, doi:10.1109/icma.2015.7237857.

[2] Simonyan, Karen, and Andrew Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition." *ArXiv.org*, 10 Apr. 2015, arxiv.org/abs/1409.1556.

[3] Thakur, Rohit. "Step by Step VGG16 Implementation in Keras for Beginners." *Medium*, Towards Data Science, 24 Nov. 2020, towardsdatascience.com/step-by-step-vgg16-implementation-in-keras-for-beginners-a833c686ae6c.

[4] He, Kaiming, et al. "Deep Residual Learning for Image Recognition." *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, doi:10.1109/cvpr.2016.90.

[5] Tan, V.Le. "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks." *ArXiv.org*, 28 May 2019, arxiv:1905.11946.

[6] M. Sornam, K. Muthusubash and V. Vanitha, "A Survey on Image Classification and Activity Recognition using Deep Convolutional Neural Network Architecture." *2017 Ninth International Conference on Advanced Computing (ICoAC),* Chennai, 2017, pp. 121-126, doi: 10.1109/ICoAC.2017.8441512.