# A Comparative Study on Structure on Convolutional Neural Networks and Its Performance

**Tianxiao Li**
Department of Computer Science
University of Toronto
Toronto, ON M5S 1A1
tonyli.li@mail.utoronto.ca

**Qiaoyiwen Wu**
Department of Computer Science
University of Toronto
Toronto, ON M5S 1A1
qiaoyiwen.wu@mail.utoronto.ca

**Richard Yan**
Department of Computer Science
University of Toronto
Toronto, ON M5S 1A1
rich.yan@mail.utoronto.ca

## Abstract

Deep neural networks have become the cornerstone of modern machine learning, and their performance on image classification tasks has been constantly improving over years. In this paper, we compare the performance of several classic convolutional neural networks on the CIFAR-10 dataset. Our motivation is to potentially discover the reason behind the increase of performance of CNN on the different structural techniques introduced. We will reproduce their original algorithms, discuss how they work, and provide an in-depth analysis of their results.

We conclude that AlexNet outperforms LeNet on the CIFAR-10 dataset due to the larger and deeper architecture of AlexNet, and ResNet outperforms VGG due to residue/skip connections which solves the issue of vanishing/exploding gradient. Our study highlights the importance of model architecture in deep learning and provides insights into the trends of deep learning research. These findings can inform the design of future deep neural networks for image classification tasks.

## 1 Introduction

Convolutional neural networks have demonstrated great success in tasks such as image classifying, handwritten digit classifying, human face detection and more. While convnet gets more and more powerful, there isn't a clear insight of why those specific architecture (LeNet, AlexNet, etc.) have been a success. Without deep understanding among the power expressed by those specific architecture, building convnet seems to be a guess-and-check progress. With this in mind, in this paper we perform a comparative study on LeNet [4], AlexNet [3], VGG [5], and ResNet [1], analyze and explain how the difference in such architecture impact the performance on task of image classifying on the CIFAR-10 dataset.

## 2 Related works

The field of image classification has undergone a revolution in recent years due to the rapid development of deep learning algorithms. Deep neural networks, particularly convolutional neural networks (CNNs), have revolutionized image classification by enabling the automatic learning of features directly from image data. This has eliminated the need for hand-crafted feature engineering,

Preprint. Under review.

and has allowed for the creation of more accurate and robust image classifiers. One of the first successful applications of convolutional neural networks (CNNs) for image recognition is LeNet, which was developed by Yann LeCun and his colleagues in the 1990s [4]. LeNet was primarily used for recognizing handwritten digits, but it laid the foundation for future developments in image recognition. In 2012, Alex Krizhevsky and his colleagues introduced AlexNet [3], a deep neural network architecture that achieved a dramatic improvement in image classification accuracy in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC). It was a much larger and more complex neural network than LeNet, and it was designed to recognize objects in the 1,000 categories of the ImageNet dataset. AlexNet's success demonstrated the power of deep learning for image classification, and helped to popularize the use of deep neural networks for computer vision tasks, which includes VGG and ResNet.

# 3 Method

We reproduce two classic convnet models, LeNet and AlexNet, as defined by [4] and [3], and two deeper convnet models, VGG16 [5] and ResNet50 [1]. These models take 2D image as input, then pass the input image over a sequence of layers, to generate a probability vector over output classes.

## 3.1 Convnet layers

**Convolution layer:** Performs convolution on the previous layer's output (or input image in the first layer) with specified kernal size, padding, and stride.

**Max-pooling layer:** Performs max-pooling on the previous layer's output, to reduce the dimension of the passed input.

**Activation layer:** Applies possibly different activation function to the previous layer's output. LeNet mainly used sigmoid activation function $\left(\sigma(x) = \frac{1}{1+\exp^{-x}}\right)$, while other networks mainly used ReLU (Rectified Linear Unit) activation function $(ReLU(x) = \max(0, x))$.

**Fully connected layer:** Layers that are fully connected, usually at the top of the network, to dense and reduce the dimension of the output from convolution and pooling into output class predictions.

## 3.2 CIFAR-10 Dataset

CIFAR-10 dataset contains 60000 colored image of size $32 \times 32$, labeled in 10 classes. [2]

## 3.3 Approach

We train our reproduced models on the CIFAR-10 dataset, and use cross entropy loss function along with accuracies to indicate how well the models are performing. Parameters of the network will be trained and learned via backpropagation using batch gradient descent.

# 4 Experiment

## 4.1 LeNet vs. AlexNet

We could see that under the same dataset, AlexNet achieves lower validation loss and higher accuracy 1 2, while LeNet shows evidence of over-training, where the performance of the model on the validation set fluctuates for quite a bit.

As we have a closer look towards the training results, we could see that although LeNet reduces the training loss to a smaller value than AlexNet, the fact that its validation loss and accuracy fluctuates showed that the model doesn't generalize well. Rather, AlexNet shows almost perfect generalization, where the validation loss is much smoother, and even lower than the training loss.

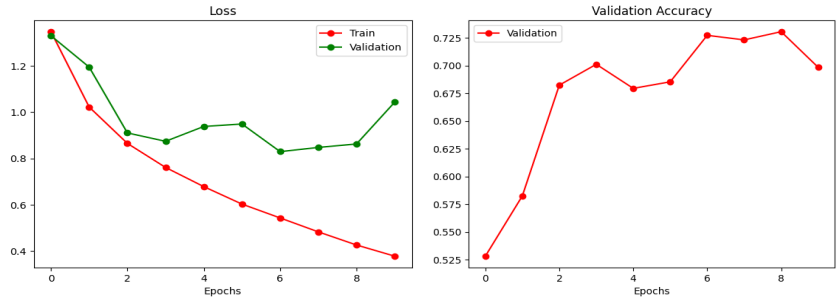


Figure 1: LeNet training loss, validation loss and validation accuracy

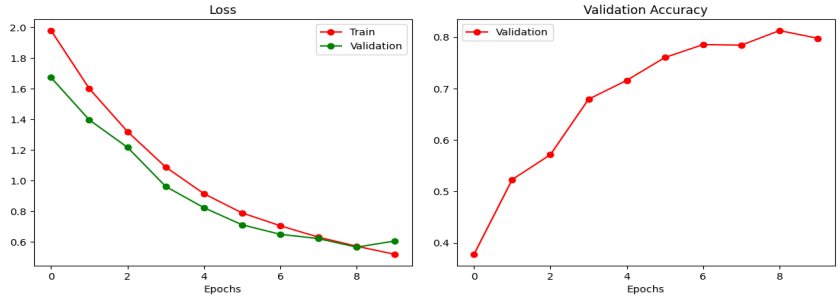


Figure 2: AlexNet training loss, validation loss and validation accuracy

### 4.2 VGG vs. ResNet

#### 4.2.1 VGG Structure

The third model we used is VGG16 from [5], which the main difference than the previous models like AlexNet is that VGG uses more smaller kernel maps during the convolution layers to replace a single large kernel size convolution layer. For example, the first convolution layer in AlexNet has kernel size 11, while if we stack together 5 convolution layers with kernel size 3, we could achieve the same receptive field after those convolutions. The benefits of doing so is reducing the weights of the layers. For example, assuming the input channels and output channels are equivalent, a single convolution layer with kernel size 9 and both input and output channels 32 will have in total $9^2 * 32 * 32 = 82944$ weights, where stacking 4 convolution layers with kernel size 3 with same input and output channels will have in total $4(3^2 * 32 * 32) = 36864$ weights. This is more than half of weights reduced! Additionally, this architecture also introduce more non-linear activation layer in between those convolution layers, which suggestively improve the power of the model.

#### 4.2.2 ResNet Structure

While VGG shows much less weights and more non-linearity, one potential limitation of introducing more layers is vanishing/exploding gradient. The fourth model we used is ResNet50 from [1]. The main difference of ResNet and VGG is that ResNet introduced residual learning, which is what we called skip connections in assignment 2. This help the gradient flow smoother back into the lower layers, where there are 'shortcuts' for the gradient back-propagation process to reach the lower layers. This improvement is designated to reduce the impact of vanishing/exploding gradient when building deeper networks.

#### 4.2.3 Comparison

Although both models show similar loss curve shape and accuracy curve shape, as we have a closer look, we could see that ResNet outperforms VGG by almost 20% validation accuracy, and ResNet achieves much lower training and validation loss than VGG does. 3 4

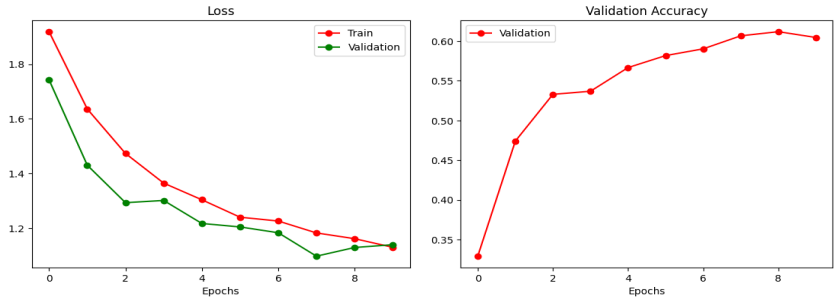


Figure 3: VGG16 training loss, validation loss and validation accuracy

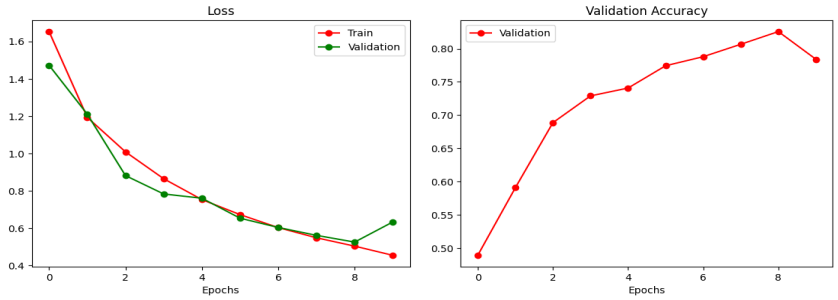


Figure 4: ResNet50 training loss, validation loss and validation accuracy

This proven our theory that as models get deeper and deeper, the vanishing/exploding gradient issue will impact the performance, and the shape of the training curve of VGG is indeed more sharper than other models.

### 4.3 Putting It Together

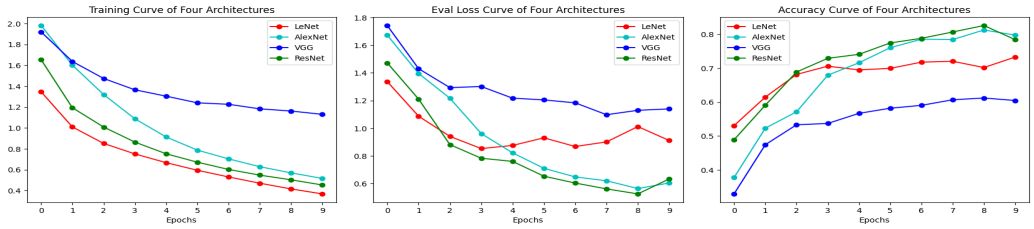


Figure 5: Combined training loss, validation loss and validation accuracy

## 5 Conclusion and Limitation

Although deeper networks have greater ability to recognize images better, without implementing specific techniques that solves issues like vanishing/exploding gradient that only deep networks have is critical on its performance.

Some limitation of this experiment is the dataset we used. CIFAR-10 dataset contains $32 \times 32$ size images, where AlexNet takes $227 \times 227$ size images as input, and VGG takes $224 \times 224$ size images as input. Resizing small images into larger resolution may not be the most efficient way to allow those more complex models to function at their max potentials. Using dataset with larger images (and down-sample them to fit the input of LeNet) may generate better results.

## References


[1] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[2] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[3] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

[4] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[5] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2015.