Improving Convolutional Neural Network Image Classification Using Alternative Forms of Neural Networks

Adrian Rodriguez-Cruz

Department of Computer Science, Georgia State University,
Atlanta, Georgia, USA
arodriguezcruz1@student.gsu.edu

Abstract— **This paper strives to demonstrate the strong suit of a few types of Neural Networks in Image Classification. When referring to Image Classification, the dataset used is the CIFAR-10 (Canadian Institute for Advanced Research, 10 class) dataset. Our focus will be on one of the more popular neural networks called a Convolutional Neural Network (CNN), we will follow a comparison of this CNN with other neural networks within a defined and constrained format. The paper shows us how Residual Neural Networks (ResNets) and Capsule Neural Networks (CapsNets) compete with CNN's. The evaluation is done by determining the validation accuracy, accuracy, and runtime of each neural network. Google Colab was a major factor in this process. It was used to determine runtimes using the T4 GPU provided. It should be noted that Google Colab limits the use of T4 GPU which lengthened the process of achieving results. In terms of validation accuracy and accuracy I demonstrate that, within the constraints, Convolutional Neural Networks perform the best overall, with only the constrained CapsNet being able to achieve greater accuracy.**

Keywords (**Include at least 5 keywords or phrases*****)***—
convolutional neural network, image classification, CIFAR-10 dataset, residual neural network, capsule network

I. Introduction

The problem of computer vision is a classic problem in computer science. The CIFAR-10 dataset is a core component within the field of computer vision and image classification that has paved the way for a variety of advancements in Artificial intelligence. An important part of artificial intelligence and computer vision is exploring a variety of alternative neural networks. Few times do neural networks get the chance to compete head-on in challenges with exact constraints in the way that I do in this paper. The neural networks in question are the Convolutional Neural Network (CNN), the Residual Neural Network (ResNet), and the Capsule Neural Network (CapsNet). The Traditional Convolutional Neural Network is a necessary choice as it is a classic neural network used in computer vision for the CIFAR-10 dataset. Following that thought process the ResNet was chosen since it's been shown to perform well with the CIFAR-10 dataset as it is a form of CNN. Finally, the CapsNet was chosen because it was proposed as an alternative to the traditional CNN, and this paper aims to put that to the test. The objective of this paper is to demonstrate which Neural Network performs best based on a few evaluative guidelines. The most important guidelines are the 2 constraints. The first constraint is that the model architecture is only allowed to use 10 model layers. The second constraint is that during training the model must strictly iterate through 10 epochs. The final evaluative guideline is the criteria that all neural networks are tested on validation accuracy, accuracy, and runtime.

II. Related Work

A. Image Classification Approaches

Several techniques are used to approach the problem of image classification. You can use traditional machine learning algorithms such as Support Vector Machines, Random Forests and Decision Trees, and even K-nearest neighbors. Most often CNNs are used, whether it's the traditional CNN, LeNet, AlexNet, VGG, GoogLeNet, or Resnet these different architectures are often proven to be successful in image classification. Following that are several techniques, that, used in the right combination, can prove to be highly successful in the task of image classification with the CIFAR-10 dataset. These optimizations include data augmentation, ensemble learning, pooling, activation functions, batch normalizations, attention mechanisms, loss functions, and regularization techniques. A combination of each of these optimization techniques is present within each neural network. It is also important to discuss the relevant strengths and weaknesses of each neural network.

B. Convolutional Neural Networks (CNNs)

Traditional Convolutional Neural Networks are highly successful in computer vision tasks, especially in image classification. They generally have a great strength in the simplicity of their architecture [2].

C. Residual Neural Networks (ResNets)

Residual Networks are used for their ability to train deep neural networks. They have a strength in addressing the vanishing gradient problem and they ease the training of deep networks using residual connections [6].

D. Capsule Neural Networks (CapsNets)

CapsNets represent an approach that was introduced to deal with the limitations of traditional CNNs. Their strengths lie in a few things, they aim to capture part-whole relationships more effectively than traditional neural networks [10]. They improve generalization to transformed inputs. They represent spatial relationships more directly than traditional pooling mechanisms [9].

E. Weaknesses

As it turns out, the Neural Networks share many common, relevant weaknesses. The major relevant weakness in this paper is the computational cost of neural networks. Neural Networks require the use of powerful hardware like GPUs or TPUs which limits their availability in resource-constrained environments like the average home desktop computer.

F. Comparative Studies

In this paper we do a comparative study of the aforementioned neural networks and provide results, in the hopes of producing something akin to an objective competition with a summary and present results [3].

III. Methodology

A. Dataset

The implemented dataset for all the neural networks was the CIFAR-10 dataset. The CIFAR-10 dataset is a 60,000-image dataset with 10 classes. The classes in the dataset are as follows airplane, automobile (this class includes sedans, SUVs, and other similar vehicle subtypes), bird, cat, deer, dog, frog, horse, ship, and truck (only big trucks, it does not include pickup trucks). Each class has 6000 images and each image is 32x32 pixel colour images. 50,000 of the images are training images and 10,000 are test images [8].

B. Model Architectures

The model architectures of all involved neural networks involved being restricted to exactly 10 model layers that were attempted to be optimized for the greatest results.

1) *Convolutional Neural Network (CNN)*: The CNN architecture consisted of 4 convolutional layers using ReLU activation. The first and third layer was followed by a max pooling layer which was employed to downsample the feature maps. The second convolutional layer was followed by batch normalization. Batch Normalization is applied after dense layers to stabilize and accelerate training. The final layers were then flattened to transition into fully connected layers

2) *Residual Neural Network (ResNet):* The ResNet architecture found success in being immediately flattened and then connected. It had dense layers that used ReLu activation and the "He normal" initialization technique. "He normal" is a type of initialization technique that sets the initial weights of neurons using this mathematical process:

$$w \sim N(0, \sqrt{\frac{2}{n}})$$

This initialization prevents exploding gradients. The dense layers were typically followed by batch normalization. One layer was a dropout layer which had a dropout rate of 50% to control the amount of regularization applied to the model. The final layer used a softmax activation because it is paired with the categorical cross-entropy loss function when training the model for multi-class classification.

3) *Capsule Neural Network (CapsNet):* The CapsNet architecture begins with 3 convolutional layers with ReLU activation followed by batch normalization. It is then followed by a capsule layer, which is then flattened. And then it is followed by connecting the layers and preparing them for output when building the model. This is the only Neural Network to not use 10 layers, it uses 8 which is within the maximum constraint.

C. *Training Details*

All models were trained for 10 epochs using the “Adam” optimizer [11]. The Adam optimizer is used for its adaptive learning rate properties and its ability to handle sparse gradients. Results were then printed, analyzed, and recorded. The following section presents them.

IV. Results

Here I demonstrate and analyze some of the best results of each neural network and compare them against one another including challenges I observed while optimizing each of them. Each neural network comes with a figure that demonstrates the accuracy and validation it achieved, followed by a figure that demonstrates a confusion matrix of the accuracy of each neural network.

A. *Convolutional Neural Networks (CNNs)*

The CNN achieved a validation accuracy of 72.57%, with a subsequent accuracy of 84.65% on the testing set. The runtime for training and evaluation was approximately 2 minutes on average. Analysis of results indicates that the neural network struggled to distinguish animals, more often it struggled with animals with similar features, like cats and dogs. Likewise, it struggled with distinguishing trucks and automobiles. I would say this struggle has to do with SUVs that can present similar features to big trucks which can be hard to distinguish for the neural network.

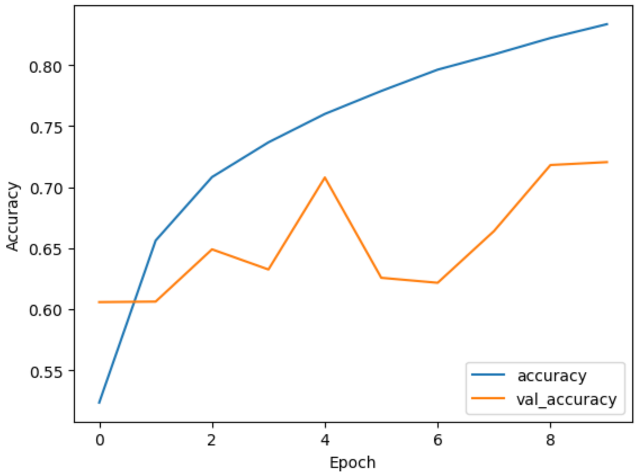
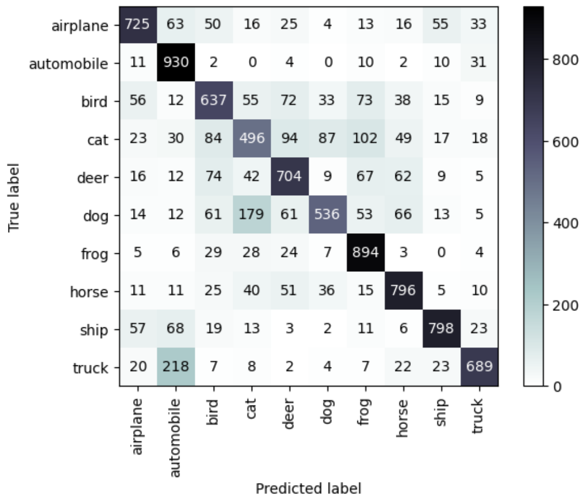


Fig. 1. Validation accuracy and accuracy results of CNN



	airplane	automobile	bird	cat	deer	dog	frog	horse	ship	truck
airplane	725	63	50	16	25	4	13	16	55	33
automobile	11	930	2	0	4	0	10	2	10	31
bird	56	12	637	55	72	33	73	38	15	9
cat	23	30	84	496	94	87	102	49	17	18
deer	16	12	74	42	704	9	67	62	9	5
dog	14	12	61	179	61	536	53	66	13	5
frog	5	6	29	28	24	7	894	3	0	4
horse	11	11	25	40	51	36	15	796	5	10
ship	57	68	19	13	3	2	11	6	798	23
truck	20	218	7	8	2	4	7	22	23	689

Fig. 2. Confusion matrix from CNN

B. *Residual Neural Networks (ResNets)*

The ResNet achieved a validation accuracy of 26.08%, with a subsequent accuracy of 44.92% on the testing set. The runtime for training and evaluation was approximately 5 minutes on average. Here It is clear that the ResNet demonstrated complete failure in learning the data in a meaningful manner. I believe this is likely due to a lack of optimization combined with working with a 10-layer constraint. The 10-layer constraint damages the ability of the ResNet to learn more deeply as it is intended to do, which doesn’t allow it to gain as deep of an understanding as the convolutional neural network might. As you can see in Figure 4, it demonstrated a bias towards labeling most things a ship, followed by airplane, bird, deer, and frog.

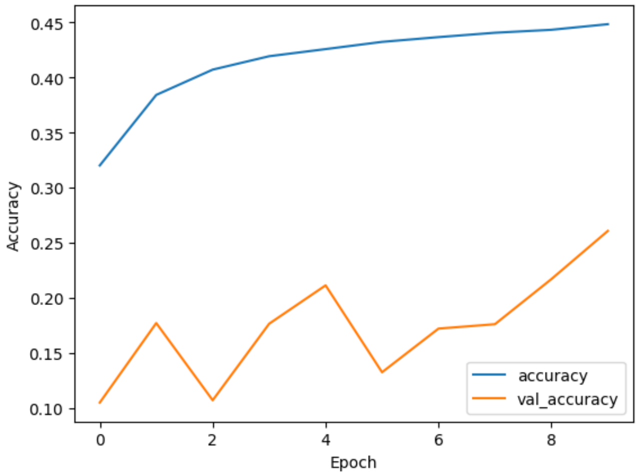


Fig. 3. Validation accuracy and accuracy results of ResNet

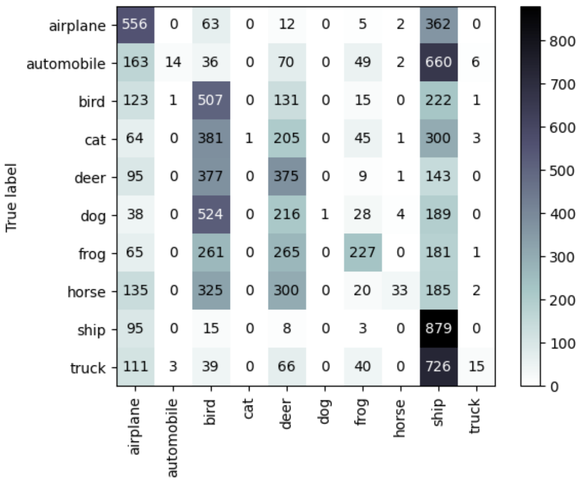


Fig. 4. Confusion matrix from ResNet

C. *Capsule Neural Networks (CapsNets)*

The CapsNet achieved a validation accuracy of 67.98%, with a subsequent accuracy of 96.85% on the testing set. The runtime for training and evaluation was approximately 3 minutes on average. The CapsNet, like the CNN, struggled with distinguishing cat images from dog images. It also struggled with truck images and automobile images. Once again, this is due to the similarity in the features of each of these classes.

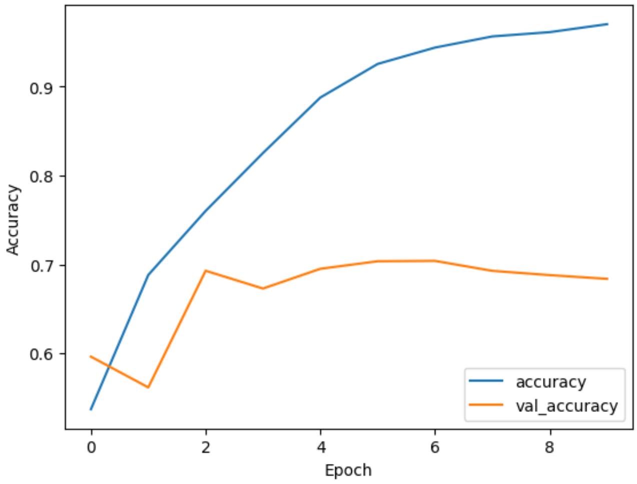


Fig. 5. Validation accuracy and accuracy results of CapsNet

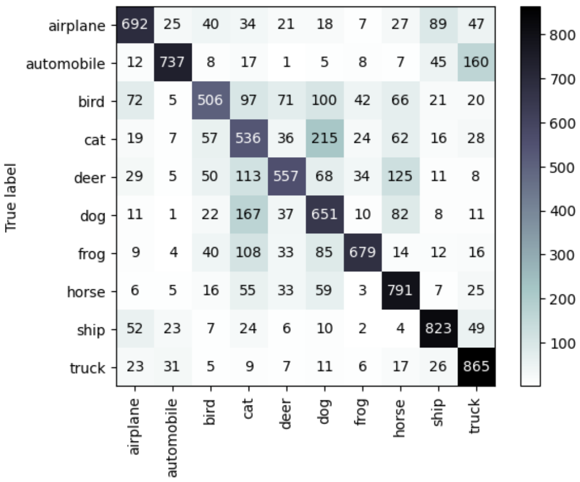
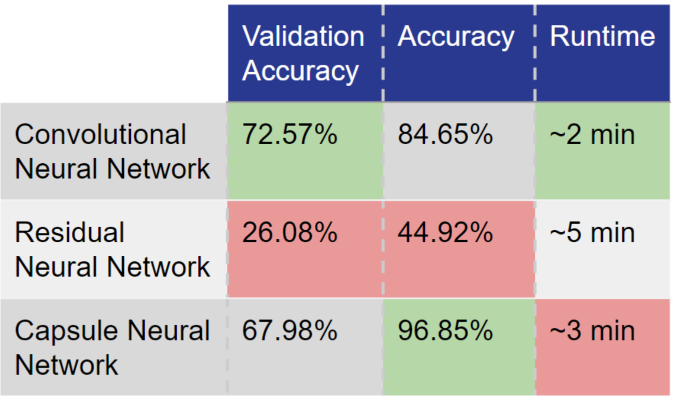


Fig. 6. Confusion matrix from CapsNet

D. *Comparative Analysis*

The three architectures revealed that CNN and CapsNet outperformed in certain aspects, whereas ResNet faced challenges in capturing the features that define each class. The runtime differences indicate that more optimization techniques could have been used for each architecture, however, it also demonstrates the complexity difference between each network. To give more insight, ResNet took the longest in terms of runtime without any optimization. However, the CapsNet, although having a better runtime, was only able to achieve that run time with added optimizations. CapsNet required a learning rate scheduler to run within the System RAM limits provided by Google Colab. This means that it was the most architecturally complex in its natural form. Below we see a comparison of the results of each neural network.

TABLE I



	Validation Accuracy	Accuracy	Runtime
Convolutional Neural Network	🟩 72.57%	84.65%	🟩 ~2 min
Residual Neural Network	🟥 26.08%	🟥 44.92%	~5 min
Capsule Neural Network	67.98%	🟩 96.85%	🟥 ~3 min

Fig. 7. Comparison of each neural network with best (green) and worst (red) performances highlighted

V. Discussion

Other observations I noticed were the ratios of validation accuracy-to-accuracy of each neural network. These ratios have implications for the degree of success each neural network has in terms of greater validation accuracy from its accuracy, in other words, greater generalization from what it's learned. CNN had the highest ratio of about 85% success in generalizing from its learning. CapsNets had a ratio of about 70% in generalizing from its learning. ResNets had a generalization from learning ratio of about 58%. The ratio of each of these neural networks I would assume can be greatly improved through optimization techniques that could have been more present in this paper, however, the aim was to try to keep them highly constrained and as close to their base model architecture as possible.

As previously discussed there were some limitations from the start in my research. First Google Colab does not allow users to use more than 12.7 gigabytes of their provided System RAM, this is what forced me to opt for introducing greater optimization for the Capsule Network. Google Colab also, reasonably, limits how much each user can continuously use their hardware accelerators, which meant that my use of the T4 GPU had to be limited, preventing me from finding results for the Neural Networks sooner and thus preventing me from including more neural networks in the paper.

I believe that to improve these findings the paper could allow for slightly less restriction to allow each neural network to compete to the higher potentials they can perform to. They can be improved through a variety of optimization techniques that were intentionally avoided (for the most part) and perhaps might be able to show just how much each optimization technique can affect and improve the results of a neural network. I also believe a greater sample size could improve the outcome of this paper, for example, if I was able to run each of these 100 times to achieve an average result, I believe it would be far more valuable than running it the limited amount of times I was able to to find a maximum result. Maximum results can represent a small portion of the upper limit of each neural network, which, although not harmful, is not representative of the average capacity of the neural network.

VI. Conclusion

In conclusion, this study compared Convolutional Neural Networks, Residual Neural Networks, and Capsule Networks in the context of image classification. Results demonstrate and create an environment to discuss the strengths of each neural network, but more importantly the weaknesses of each neural network at a high degree of constraint. I believe weakness is an important factor that is not expressly mentioned in most research and practical applications which can hide the potential for improving several methods, even here in computer vision. Finding where to improve the weaknesses of all techniques and methods would push for mass advancements in research.

VII. Acknowledgments

I, the author, would like to express gratitude to Dr. Sergey Plis for his valuable contribution to accelerating my research. I would also like to thank Dr. Yanqing Zhang, my professor, for providing the most valuable IEEExplore resource that allowed me to research the many AI topics required to make this research paper come together. Lastly, I would like to thank Georgia State University for providing me access to the various research papers required to create this paper. Without these crucial contributions, the paper would not be possible.

References

[1] S. Albawi, T. A. Mohammed and S. Al-Zawi, "Understanding of a convolutional neural network," 2017 International Conference on Engineering and Technology (ICET), Antalya, Turkey, 2017, pp. 1-6, doi: 10.1109/ICEngTechnol.2017.8308186.

[2] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," in Conf. International Conference on Learning Representations, San Diego, CA, 2015.

[3] S. Bak, C. Liu, T. Johnson, "The Second International Verification of Neural Networks Competition (VNN-COMP 2021): Summary and Results," Stony Brooks, Carnegie Mellon University, Vanderbilt University, arXiv:2109.00498v1 [cs.LO], 2021

[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in Proceedings of the 25th International Conference on Neural Information Processing Systems (NIPS), 2012, pp. 1097-1105.

[5] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," in Proceedings of the 3rd International Conference on Learning Representations (ICLR), 2015.

[6] 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 (CVPR), 2016, pp. 770-778.

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Identity Mappings in Deep Residual Networks," in Proceedings of the European Conference on Computer Vision (ECCV), 2016, pp. 630-645.

[8] A. Krizhevsky, “Object classification experiments,” in *Learning Multiple Layers of Features from Tiny Images,* Univ. of Toronto, Canada, 2009. Available: https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf

[9] S. Sabour, G. E. Hinton, and N. Frosst, "Dynamic Routing Between Capsules," in Advances in Neural Information Processing Systems (NeurIPS), 2017, pp. 3856-3866.

[10] A. Hinton, A. Krizhevsky, and S. Wang, "Matrix Capsules with EM Routing," in Proceedings of the International Conference on Learning Representations (ICLR), 2018.

[11] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in Proceedings of the International Conference on Learning Representations (ICLR), 2015.