Comparing CNN and ResNet Architectures for CIFAR-10 Classification

Arnav Jain
Computer Science
Texas A&M University
College Station, TX, USA
arnavkj11@tamu.edu

Fatma Ozair
Genetics and Genomics
Texas A&M University
College Station, TX, USA
fo2788@tamu.edu

Neha Manghnani
Computer Science
Texas A&M University
College Station, TX, USA
nehajm@tamu.edu

Priyanka Askani
Computer Science
Texas A&M University
College Station, TX, USA
askani2407@tamu.edu

Shreya Gubbi Prakash
Computer Science
Texas A&M University
College Station, TX, USA
shreyagp@tamu.edu

***Abstract*—This paper presents an extensive evaluation of three deep learning architectures, a basic Convolutional Neural Network (CNN), ResNet18 and ResNet34 for image classification using the CIFAR-10 dataset. Comprising images across ten distinct categories, CIFAR-10 serves as a widely recognized benchmark for evaluating the performance of image classification models. Our study aims to determine the most accurate model by applying normalization, Principal Component Analysis (PCA) for dimensionality reduction and a range of data augmentation techniques, such as random cropping, flipping, rotation and CutOut regularization, to improve model generalization. Each model is optimized using the AdamW optimizer and a OneCycle learning rate schedule to achieve efficient convergence. We evaluate model performance with metrics including accuracy, F1 score, precision, recall and a confusion matrix for detailed class-wise accuracy visualization. Our findings reveal that ResNet34 surpasses CNN and ResNet18 in both accuracy and generalization, making it the most effective choice for CIFAR-10 classification. This research demonstrates ResNet34's superior performance and robustness in handling complex image classification tasks.**

***Index Terms*—Convolutional Neural Network, ResNet18, ResNet34, CIFAR-10, PCA, CutOut, Confusion Matrix, Image Classification.**

I. Introduction

The increase in volume and complexity of image datasets have made it crucial to choose the right model for efficient classification tasks. Deep learning models, notably Convolutional Neural Networks (CNNs) and Residual Networks (ResNets) have become essential for high-performance image classification. CNNs are widely used due to their layered approach to extracting spatial features from images, making them effective for various visual recognition tasks [1]. Simple CNNs extract patterns, such as textures and edges, from convolutional layers of a training dataset. The model is trained by adjusting the weights of the convolutional layers using backpropagation. Due to its simplicity, it may struggle to recognize subtle patterns from larger, heterogeneous datasets, ultimately resulting in overfitting. ResNets, on the other hand, have introduced residual connections to solve issues like the vanishing gradient, enabling deeper networks with better learning efficiency and generalization [2]. These models are unique to simple CNNs in that they have skip connections in which the model can refer to connections already learned in a previous block. Thus, it can retain "memories" of features as it reaches deeper layers to help distinguish a more complex image dataset. The CIFAR-10 dataset, a standard benchmark containing 60,000 labeled images across ten classes, is often used to evaluate the effectiveness of such models in distinguishing complex visual patterns . This study presents a detailed comparison of three models: a basic Convolutional Neural Network (CNN), ResNet18, and ResNet34 on the CIFAR-10 image classification task. CIFAR-10, with its diverse set of 60,000 low-resolution images spread across ten classes, presents significant challenges in terms of feature extraction and generalization. To enhance the model's performance and robustness, we apply several pre-processing techniques, including normalization, principal component analysis (PCA) for dimensionality reduction, and a range of data augmentations. These augmentations are random cropping, horizontal flipping, rotation, and CutOut regularization. They have been shown to improve model generalization by simulating real-world variability, thereby helping models better adapt to unseen data [3]. PCA, a commonly used method for minimizing dimensionality, is included here to reduce computational load while preserving key image features [5]. ResNet models are trained using the AdamW optimizer, which decouples weight decay from gradient-based updates to improve regularization, and a OneCycle learning rate schedule to accelerate convergence and enhance stability [4]. To assess model performance comprehensively, we evaluate each model on metrics including accuracy, F1 score, precision, recall, and utilize a confusion matrix to analyze class-wise accuracy and misclassifications.

II. Literature Survey

He et al. (2015) introduced ResNets to address the degradation problem in deep networks, where adding layers increased errors due to vanishing gradients. Residual blocks use identity shortcuts, allowing the input x to bypass weight layers and directly add to the output F(x). This simplifies optimization and enables training of very deep networks. ResNets demonstrated state-of-the-art results on CIFAR-10 and ImageNet, showcasing the advantages of deep architectures for accuracy. [2].

Lorente Corominas et al. (2021) compare traditional and deep learning methods for image classification, including BoVW with SVM, MLPs, and CNNs [6]. While BoVW achieves moderate accuracy, CNNs like InceptionV3 excel with deep feature extraction. They also introduce TinyNet, a lightweight CNN using depthwise convolutions and adaptive pooling, offering competitive accuracy with low computational cost. The study emphasizes the shift from handcrafted features to efficient deep-learning models for robust classification.

Liang (2020) highlights ResNet's effectiveness in image classification, addressing vanishing gradients with residual connections that improve gradient propagation and enable deep networks to learn residual functions. Experiments on CIFAR-10 show ResNet's superior performance and robustness, emphasizing its ability to handle complex features with better generalization and reduced training challenges[7].

Chen et al. (2021) review convolutional neural network (CNN)-based image classification algorithms, highlighting CNN advancements from early models to modern architectures like ResNet and Inception [8]. The review covers CNN fundamentals, such as convolutional and pooling layers, and examines optimization strategies that enhance classification performance. Applications in remote sensing are also discussed, showcasing CNN's versatility in diverse image tasks and scene classification. This work provides an overview of CNN evolution and its impact on image recognition across fields.

III. Methodology

A. Data Processing and Cleaning

The CIFAR 10 dataset is loaded from the PyTorch datasets website. The data was cleaned by removing corrupted and duplicated images from the dataset. After cleaning the dataset, the class distributions were calculated to ensure the dataset is equally distributed among the 10 classes. To bring diversity and variation to the CIFAR 10 dataset, transformations such as flipping, rotating, cropping, and changing the contrast and brightness were applied. These methods bring diversity in the training dataset, which make the models' predictions more accurate even when there are variations in the input image. In contrast, the test dataset underwent only basic transformations, including resizing and normalization, to maintain consistent and unbiased evaluation conditions.

The dataset is split into training, validation and test data. The validation dataset is 10% of the training dataset which is used to tune the parameters of the model. We split the data into batches for efficient training.

As part of the pre-processing , the image data is flattened and scaled using StandardScaler. This step makes the image data more compatible with the machine learning models by normalizing the features of the image data. Scaling ensures that all features contribute equally to the learning step by normalizing them to have uniform variance.

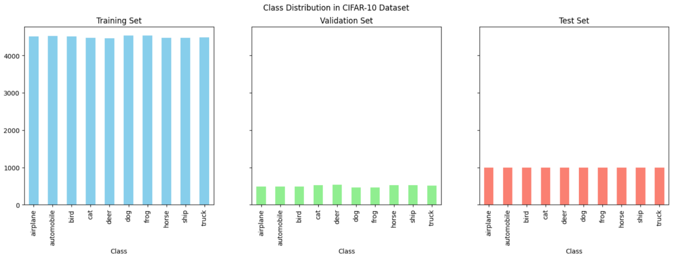

Fig. 1: Distribution of classes among train, validation and test

B. Exploratory Data Analysis

The CIFAR-10 dataset contains 60,000 images of the objects: airplane, automobile, bird, cat, deer, dog, frog, horse, ship and truck. To get a better understanding of the dataset, we perform an exploratory analysis. We get the number of images for each class in the train, validation and test dataset. It is observed from Table 1 that the number of images in the datasets are equally distributed among the classes. The distributions are also equal in the train, validation and test datasets respectively as observed in Figure 1. This makes CIFAR 10 a balanced dataset.

Class	Train Count	Validation Count	Test Count
airplane	4515	485	1000
automobile	4519	481	1000
bird	4508	492	1000
cat	4479	521	1000
deer	4467	533	1000
dog	4533	467	1000
frog	4543	457	1000
horse	4481	519	1000
ship	4471	529	1000
truck	4484	516	1000

TABLE I: Class-wise Train, Validation, and Test Counts

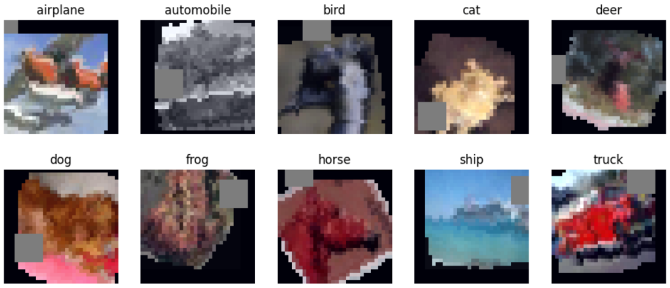

Fig. 2: Example Images from the Training Dataset

We first start with observing an image from each class in the dataset, as shown in Figure 2. Additionally, Figure 4 shows images transformed during the data preparation and cleaning phase of the project. These transformations include rotation, cropping, resizing, and greyscaling to introduce some diversity into the training dataset.

Following the pre-processing steps, we apply dimensionality reduction to the dataset using Principal Component Analysis. PCA uses a vector space transform to reduce the dimensionality of the image dataset. It finds the directions of maximum variance in high-dimensional data that is equivalent to the least squares line of best fit through the plotted data and projects it onto a smaller dimensional subspace while retaining most of the information. In practice, it is sufficient to include enough principal components that cover about (70 - 80%) of the data variation. The reduced-dimension dataset allows users to interpret, analyze, and process data in an easy way [9].

The plot in Figure 3 shows the PCA by projecting the image data in 2D using two principal components. The unclear separability among the classes indicated by the data points suggests that more than two features would be required to clearly distinguish between the classes.

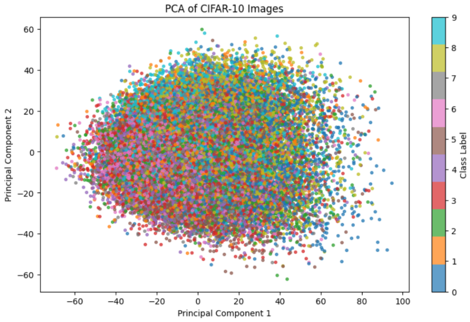

Fig. 3: PCA of CIFAR-10 Images

The graph in Figure 5 indicates that the first 500-1000 principal components are the most predominant features in distinguishing the images into different classes. It retains 95% of the image information without loss while also reducing the complexity and redundancy in the dataset, as expected with dimensionality reduction techniques.

Fig. 4: Augmented Images

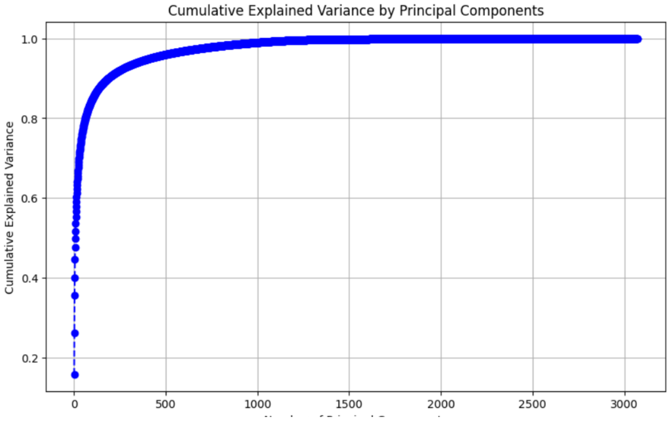

Fig. 5: Cumulative Explained Variance by Principal Components

We also calculate data statistics like mean and standard deviation to better understand the image dataset. As shown in Table 2, the training and validation have similar values for the metrics, while the test dataset has different values. The difference in the values is because of the transformations which are applied to the training dataset and not on the test dataset.

Dataset	Mean	Standard Deviation
Training Set	[-0.5883, -0.5943, -0.5316]	[1.4250, 1.4271, 1.4041]
Validation Set	[-0.5946, -0.6004, -0.5364]	[1.4286, 1.4298, 1.4099]
Test Set	[0.0139, 0.0147, 0.0194]	[1.2192, 1.2181, 1.3015]

TABLE II: Mean and Standard Deviation for Training, Validation, and Test Sets (R, G, B Channels)

C. Model Selection

To classify the CIFAR-10 image dataset we selected two models - CNN and ResNET.

Convolutional Neural Network - Convolution Neural Network (CNN), often called ConvNet, has deep feed-forward architecture and has an astonishing ability to generalize in a better way as compared to networks with fully connected layers. CNNs are widely being used in various domains due to their remarkable performance such as image classification, object detection , face detection , speech recognition, vehicle recognition, diabetic retinopathy, facial expression recognition and many more.[10]

In this project, we have applied basic sequential CNN with three convolution layers with a kernel filter of size 3x3 to extract features from the input image. Convolution layers are used to extract features at every layer. The combination of convolution kernels and deep and hierarchical structures are very effective in learning good representations (features) from images for visual recognition tasks. ReLU activation functions are applied to introduce nonlinearity into the CNN, allowing it to model complex patterns in the data. [11] Pooling layers help in retaining important features and downsampling the image data. We consider this as our baseline model for the image classification of the CIFAR-10 dataset.

ResNet - Deep convolutional neural networks have led to a series of breakthroughs for image classification. Network depth is of crucial importance with good results. The deeper the network the better the results. However, deep neural networks have the problem of vanishing or exploding gradients. ResNet is a deep learning architecture in which the layers learn residual functions with reference to the layer inputs [12]. ResNET is originally designed for computer vision applications.

In this project, we apply ResNET18 and ResNET34 architecture on the CIFAR-10 image dataset. ResNet models leverage residual learning to resolve the vanishing gradient problem often met in deep networks. These residual blocks make it possible for the network to learn identity mappings in ResNet34 to propagate gradients effectively through layers for better feature extraction and stability during training [13]. ResNet34 consists of four groups of residual layers, while the number of blocks in each group is [3, 4, 6, 3], respectively. These groups allow for hierarchical feature learning: low-level to high-level features can be progressively extracted. Each residual block has skip connections, which are very important in improving gradient flow, hence ensuring that as depth is added to the network, no performance degradation happens- a problem that goes by the name degradation problem. We finally have a global average pooling layer at the end to reduce the spatial dimensions, followed by a fully connected layer for classifying the extracted features into respective classes.

In our experiments, ResNet34 performed better than ResNet18 by 2.8%. Such an improvement is justified because the depth in ResNet34 allows it to capture more difficult patterns and features that might be present in the dataset. More importantly, ResNet34 had significantly lower training error and generalized much better on the validation data. Improved performance indicates that residual connections in ResNet34 can capture more sophisticated image features while still allowing flows of gradients to work efficiently.

IV. Experimental Results and Discussion

A. *Model Evaluation*

The models are trained on 50,000 images of the CIFAR-10 dataset. The metrics on training and test dataset are shown in Table 3 and Table 4.

All the models are run for 80 epochs at learning rate of 0.001. The CNN model reaches a training accuracy of 98.62% and a training loss of 0.0412. The extremely low loss indicates that the model has overfit the training image dataset. Evaluating the model on the test data confirms this, as the test accuracy is 67.29% which is significantly low from the training accuracy. This shows that the model is not able to generalize well and does not give a good performance for classifying images it has not encountered before. ResNet18 model reaches a training accuracy of 94.72% and a test accuracy of 83.90%. The ResNET34 model reaches a training accuracy of 89.47% and a training loss of 0.3077. The validation accuracy is maintained at 85.14% which indicates that the model is able to generalize well and maintain accuracy with the new images. The high metrics on precision, recall and f1 score for the model indicates that ResNet 34 fits the CIFAR-10 image dataset well.

Model	Accuracy	Precision	Recall	F1 Score
CNN	97.28%	0.9732	0.9728	0.9728
ResNet18	94.72%	0.9432	0.9472	0.9435
ResNet34	89.47%	0.8966	0.8947	0.8954

TABLE III: Metrics for Training Data with Different Models

Model	Accuracy	Precision	Recall	F1 Score
CNN	67.29%	0.6745	0.6729	0.6723
ResNet18	83.90%	0.8390	0.8382	0.8392
ResNet34	90.76%	0.9091	0.9076	0.9078

TABLE IV: Metrics for Test Data with Different Models

We select ResNet34 as the best model for the image classification task and tune the hyperparameters - epochs and learning rate, for the selected model. The table shows the results of the hyperparamter tuning. We get the best performance at 80 epochs and learning rate of 0.001 which we select as the hyperparameters to compare with the other models as well.

Epoch	Learning Rate	Train Accuracy	Test Accuracy
10	0.001	64.04%	70.72%
10	0.003	66.68%	71.83%
30	0.001	80.19%	84.69%
30	0.003	81.66%	86.40%
50	0.001	85%	88.72%
50	0.003	86.57%	89.35%
80	0.001	89.47%	86.40%

TABLE V: Hyperparameter Tuning on ResNet34

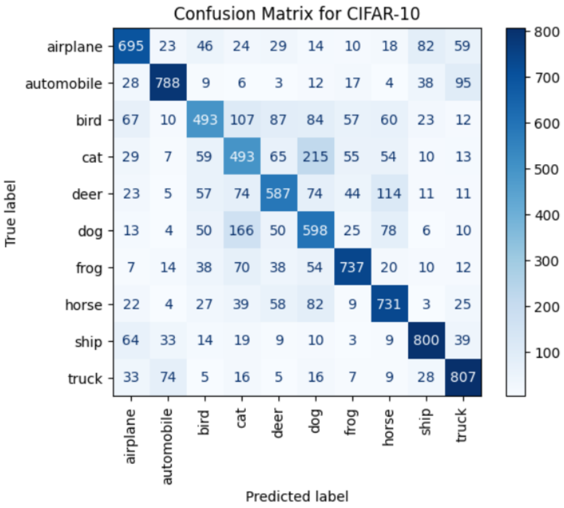

Fig. 6: Confusion Matrix from CNN

The confusion matrix in Figure 6 and Figure 7 depict how accurately the images from the test data were classified by the CNN and ResNet34 model respectively.

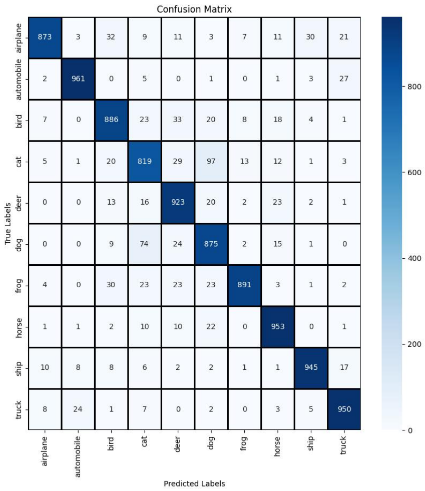

Fig. 7: Confusion Matrix from ResNet34

B. Model Comparison

In this project, we compare two models - CNN and ResNet for image classification task.

The ResNET model is built on top of the CNN architecture and is expected to perform better than the basic sequential CNN as it supports more number of layers, and addresses the issue of vanishing gradients.

We also compare the performance of ResNet18 and ResNet34 to evaluate the effect of increased number of layers on the model accuracy. The ResNET 18 model, with 80 epochs, gives a training accuracy of 94.72% and loss of 0.13675. The training accuracy is better compared to ResNet34. However, evaluating the ResNet18 with the test data, there is a drop in the accuracy to 83.90%. The drop is not that significant but the difference between the accuracy in training and test data for ResNet18 is more compared to the ResNet34 model where the difference in the accuracy is only 1%. This shows that compared to the ResNet18 model, the ResNet34 model is better able to extract features of the image data due to the increased number of layers in the model. This makes the ResNet34 model generalize well and classify unseen images more accurately compared to the ResNet18 model. This shows that increasing the number of layers helps extract more features from the images which helps in increasing the classification accuracy.

C. Model Interpretability

ResNet34 achieves its predictive power through residual blocks, which allow features from earlier layers to flow directly to deeper ones via skip connections. This structure not only enhances gradient flow but also supports the extraction of both simple and complex features, such as edges and object shapes, improving classification accuracy. Tools like confusion matrices and Class Activation Mapping (CAM) can provide further interpretability by revealing which features or image regions the model prioritizes for specific predictions, such as distinguishing similar classes like ”cat” and ”dog.”

D. Business Insights

The study highlights ResNet34’s ability to generalize well on unseen data, making it suitable for real-world applications. Businesses can leverage this model for tasks like product categorization in e-commerce or defect detection in manufacturing, where consistent and accurate image classification is critical. Its robustness reduces the need for frequent retraining, allowing companies to scale their operations efficiently across diverse datasets while ensuring high reliability in decision-making processes.

V. Conclusion

In this project, we compare the performance of CNN and ResNet for the image classification task. CNN is widely used for computer vision tasks. ResNet helps in mitigating the problems encountered by vanishing/exploding gradients when increasing the number of layers in the model. We benchmark the performance of CNN, ResNet18 and ResNet34 on the CIFAR-10 dataset. We observe that CNN performs well on the training data with the accuracy of 97.28%. The high accuracy indicates that the model likely overfits, which is confirmed with the drop in accuracy with the test data to 67.29%. ResNet34 gives a training accuracy of 89.47% and test accuracy of 90.76% which indicates that the model generalizes well and is able to predict unseen image data with high accuracy. This shows that ResNet is better able to extract the features from the image data to classify the images into the appropriate classes. We also compare the performance of two ResNet architectures - ResNet18 and ResNet34, and observe that ResNet34 has better performance indicating that more number of layers in the neural network architecture helps in better extracting the features of the image data.

References

[1] C. Szegedy et al., ”Going deeper with convolutions,” 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 2015, pp. 1-9, doi: 10.1109/CVPR.2015.7298594.

[2] K. He, X. Zhang, S. Ren, and J. Sun, ”Deep Residual Learning for Image Recognition,” arXiv preprint arXiv:1512.03385, Dec. 2015.

[3] Shorten, C., Khoshgoftaar, T.M. A survey on Image Data Augmentation for Deep Learning. J Big Data 6, 60 (2019).

[4] I. Loshchilov and F. Hutter, ”Decoupled Weight Decay Regularization,” arXiv preprint arXiv:1711.05101, Jan. 2019.

[5] J. Shlens, ”A Tutorial on Principal Component Analysis,” arXiv preprint arXiv:1404.1100, Apr. 2014.

[6] Ò. Lorente, I. Riera, and A. Rana, ”Image Classification with Classic and Deep Learning Techniques,” arXiv preprint arXiv:2105.04895, May 2021.

[7] Liang, Jiazhi. (2020). Image classification based on RESNET. Journal of Physics: Conference Series. 1634. 012110. 10.1088/1742-6596/1634/1/012110.

[8] Chen, L., Li, S., Bai, Q., Yang, J., Jiang, S., Miao, Y. (2021). Review of Image Classification Algorithms Based on Convolutional Neural Networks. Remote Sensing, 13(22), 4712.

[9] Salem, Nema, and Sahar Hussein. ”Data dimensional reduction and principal components analysis.” Procedia Computer Science 163 (2019): 292-299.

[10] Indolia, Sakshi Goswami, Anil Mishra, S.P. Asopa, Pooja. (2018). Conceptual Understanding of Convolutional Neural Network- A Deep Learning Approach. Procedia Computer Science. 132. 679-688. 10.1016/j.procs.2018.05.069.

[11] J. Wu, Introduction to Convolutional Neural Networks. LAMDA Group, National Key Lab for Novel Software Technology, Nanjing University, China.

[12] Deng, Jia Dong, Wei Socher, Richard Li, Li-Jia Li, Kai Li, Fei-Fei. (2009). ImageNet: a Large-Scale Hierarchical Image Database. IEEE Conference on Computer Vision and Pattern Recognition. 248-255. 10.1109/CVPR.2009.5206848.

[13] He, Kaiming Zhang, Xiangyu Ren, Shaoqing Sun, Jian. (2015). Deep Residual Learning for Image Recognition. 7.