Assignment 2 - Deep Learning Fundamentals

Dang Thinh Nguyen
The University of Adelaide
dangthinh.nguyen@student.adelaide.edu.au

Abstract

This study investigates the performance of multiple Convolutional Neural Networks (CNNs) for image classification on the CIFAR-10 dataset. Three CNN architectures (ResNet-18, AlexNet, and MobileNet) were evaluated in terms of their accuracy, computational efficiency, and generalizability. To optimize performance, each model was fine-tuned with different learning rates and optimizers, using a training-validation split to determine the best hyperparameter configuration. The ResNet-18 model with a learning rate of 0.001 and the SGD optimizer achieved the highest validation accuracy (91.99%), making it the best model for CIFAR-10 in this study. When tested, this model achieved a test accuracy of 72.79% and an F1 score of 72.64%. These results underscore the effectiveness of ResNet's residual connections in learning complex patterns in image classification, although further optimization is possible.

1. Introduction

Image classification is a foundational task in computer vision, with applications across fields such as healthcare (e.g., medical image analysis), autonomous driving (object recognition), and security (surveillance systems) [1]. Convolutional Neural Networks (CNNs) are widely adopted for this task due to their capacity to hierarchically capture spatial patterns within image data, making them highly effective in recognizing complex visual structures [1].

The CIFAR-10 dataset is a well-known image classification benchmark composed of 60,000 labeled images across 10 distinct classes, each 32x32 pixels in size [2]. This dataset presents a balanced classification challenge, as models must distinguish between visually similar classes such as "cat" and "dog". In this study, we evaluate the performance of three CNN architectures (ResNet-18, AlexNet, and MobileNet-V2) on CIFAR-10, focusing on their accuracy, computational efficiency, and generalizability. ResNet-18's residual connections allow it to handle deeper layers without encountering the vanishing gradient problem [3], AlexNet, a simpler but highly influential architecture, provides a performance benchmark [4], and MobileNet, optimized for efficiency, offers an alternative for resource-constrained environments [5].

The objectives of this study are to identify the most effective architecture for CIFAR-10 among these models, optimize each model through hyperparameter tuning, and assess the final performance on the test set. This process will highlight the benefits and trade-offs between accuracy and computational efficiency in selecting CNN architectures for image classification tasks.

2. Methodology

This section outlines the data preprocessing steps, architecture descriptions, and hyperparameter tuning strategies used to ensure a fair and effective comparison between the CNN models. By standardizing these components, we aim to create a consistent foundation for evaluating each model's performance.

2.1. Data preprocessing

Data preprocessing is essential for optimal model training, especially when working with images. For CIFAR-10, all images were normalized using the dataset's standard mean and standard deviation values across the RGB channels, reducing variance across pixel values and facilitating faster model convergence. Data augmentation was applied to the training set to increase its diversity; transformations included random horizontal flips and random cropping. This augmentation helps prevent overfitting by introducing slight variations, forcing the model to generalize better.

The dataset was split into training (80%), validation (20%), and test sets. The training set was used for learning, the validation set for hyperparameter tuning and model selection, and the test set to assess final model performance. DataLoader objects in PyTorch managed the batching, shuffling, and efficient loading of images, with a batch size of 200 chosen to balance GPU memory usage and computational efficiency.

2.2. Model architectures

Three CNN architectures were selected for evaluation based on their unique structural properties and potential advantages in image classification:

ResNet-18

ResNet-18 is part of the ResNet family of models, known for their innovative use of residual connections. The residual connection allows the model to bypass certain layers, effectively enabling it to handle deeper architectures without encountering the vanishing gradient problem [3]. This structure allows the network to focus on learning relevant features across a more complex layer hierarchy, which can be particularly beneficial when working with a small, diverse dataset like CIFAR-10 [3]. ResNet-18 consists of 18 convolutional layers with multiple shortcut connections and does not include fully connected layers before the final classification layer [3]. In this study, the fully connected layer was adjusted to output 10 classes for CIFAR-10, and the first convolutional layer and max-pooling layer were modified to better suit the smaller image dimensions.

AlexNet

AlexNet is a classic CNN architecture that marked a breakthrough in deep learning for image classification when it won the ImageNet competition in 2012 [4]. It uses five convolutional layers with relatively large filter sizes, followed by three fully connected layers [4]. The use of ReLU activation functions and dropout in AlexNet significantly improved training stability and regularization [4]. AlexNet is known for its simpler architecture, which enables it to be trained relatively quickly on smaller datasets [4]. For this study, AlexNet's final fully connected layer was modified to produce 10 output classes, and the first convolutional layer was adjusted to better handle CIFAR-10's smaller image size.

MobileNet

MobileNet is an efficient CNN model optimized for mobile and embedded devices, using depthwise separable convolutions to significantly reduce computational costs without sacrificing too much accuracy [5]. It includes inverted residuals with linear bottlenecks, making it lightweight and fast compared to deeper architectures [5]. MobileNet is advantageous in resource-constrained environments due to its efficient layer structure [5]. For CIFAR-10, the final fully connected layer was adapted to output 10 classes, ensuring compatibility with the dataset's classification task.

2.3. Training process

To determine the optimal configuration for each model, a range of hyperparameters was explored:

Learning rates: Learning rates of 0.1, 0.01, and 0.001 were tested to observe their impact on convergence speed and accuracy. Lower learning rates generally result in more stable training, though they can increase training time.

Optimizers: Three optimization algorithms were evaluated:

1. Stochastic Gradient Descent (SGD), which updates the weights based on the gradient of the loss function calculated on a single mini-batch [6].
2. Adam, an adaptive optimizer that adjusts the learning rate dynamically during training [6].
3. RMSprop, another adaptive optimizer that modifies the learning rate based on the magnitude of recent gradients [6].

Early stopping: Early stopping was simply implemented if validation accuracy did not improve over five consecutive epochs. This approach reduces training time while preventing overfitting, preserving model generalizability.

Each model was trained and validated across different learning rates and optimizers, with the best configuration selected based on validation accuracy.

3. Experimental analysis

This section provides an in-depth look at the implementation process for each CNN model, training and validation procedures, and the experimental results obtained from the best model configurations. Each model's performance is analyzed based on validation and test accuracies, with an emphasis on training strategies, customization steps, and evaluation metrics.

3.1. Model customization and training procedure

Each of the three CNN architectures was adapted to classify CIFAR-10's 10 classes by modifying their final fully connected layers. Given CIFAR-10's smaller 32x32 image size, adjustments were made to the input layers of the models to accommodate the smaller image dimensions while preserving the primary structural elements of each architecture.

ResNet-18's residual connections were left intact to enable the model to efficiently process deeper layers without the risk of vanishing gradients. The final fully connected layer was modified to output 10 classes, with the model's initial convolutional layer adapted to handle CIFAR-10's smaller images. The max-pooling layer was replaced with an identity function to reduce information loss in early layers.

AlexNet was modified in its initial convolutional layer to better suit CIFAR-10. Additionally, the last fully connected layer was adapted to output 10 classes.

MobileNet was adjusted to output 10 classes in its last fully connected layer.

3.2. Training and validation procedure

A grid search was conducted, where the following parameters were tested:

- Models: ResNet-18, AlexNet, MobileNet
- Learning reates: 0.1, 0.01, 0.001
- Optimizers: SGD, Adam, RMSprop

Early stopping was used, with training halting if validation accuracy failed to improve for five consecutive epochs. Each combination of parameters was evaluated by training the model on the training set and monitoring its performance on the validation set. The validation accuracy and loss were recorded for each configuration, and the best-performing model was selected based on its ability to generalize to the validation data.

3.3. Evaluation metrics

To provide a comprehensive performance overview, multiple evaluation metrics were utilized:

- Accuracy: The primary measure of each model's classification performance.
- Precision, recall, and F1-score: Used to provide insight into class-specific performance and the model's ability to handle class imbalances.
- Loss: Tracked during training and validation to gauge model convergence and identify overfitting.

4. Result

The results were evaluated based on each model's performance on the test set after identifying the best hyperparameter configuration.

4.1. Training and validation models

The table below summarizes the best validation accuracy achieved by each model with the optimal learning rate and optimizer configuration:

Model	Best learning rate	Optimizer	Validation accuracy (%)
ResNet-18	0.001	SGD	91.99
AlexNet	0.001	SGD	77.73
MobileNet	0.001	RMSprop	82.08

Table 1. Optimal hyperparameters and validation accuracy for each model

ResNet-18 outperformed the other architectures with a validation accuracy of 91.99% when configured with a learning rate of 0.001 and the SGD optimizer. This model was selected for further evaluation on the test set.

4.2. Test performance of the best model

The ResNet-18 model, trained with the best configuration, was evaluated on the CIFAR-10 test set. The following results were achieved:

- Test Accuracy: 72.79%
- Precision: 72.69%
- Recall: 72.79%
- F1 Score: 72.64%

Class	Precision	Recall	F1 score
Airplane	0.74	0.81	0.77
Automobile	0.80	0.89	0.85
Bird	0.63	0.59	0.61
Cat	0.56	0.53	0.55
Deer	0.66	0.64	0.65
Dog	0.69	0.66	0.67
Frog	0.75	0.81	0.78
Horse	0.72	0.77	0.75
Ship	0.87	0.81	0.84
Truck	0.85	0.77	0.81

Table 2. Class-specific performance

The classification report provided insights into class-specific performance, revealing that ResNet-18 performed particularly well on classes such as "automobile" and "ship" with F1 scores around 85%. In contrast, more visually ambiguous classes, such as "cat", "bird", "deer" and "dog" presented greater challenges, with lower precision and recall values.

These results underscore ResNet-18's capability to learn complex features and generalize well on unseen data while handling CIFAR-10's relatively small images effectively.

5. Code availability

The full implementation of the code used for this experiment can be accessed via the following link: https://github.com/Think-Nguyen/Deep-learning-fundamentals-A2.git
This repository contains all relevant Python scripts and dependencies needed to reproduce the results discussed in this report.

6. Discussion

The results demonstrate that ResNet-18 is highly effective for CIFAR-10 classification, achieving the highest validation accuracy. The residual connections in ResNet-18 allowed it to avoid the vanishing gradient problem, a common issue in deeper networks, and improve its ability to learn complex visual patterns. AlexNet and MobileNet, while achieving reasonable accuracy, did not match ResNet-18's performance. AlexNet's simpler architecture made it susceptible to overfitting, whereas MobileNet's efficient design, optimized for mobile

devices, resulted in slightly lower accuracy.

The hyperparameter tuning process underscored the importance of selecting the appropriate learning rate and optimizer for each model. ResNet-18 benefited from a low learning rate (0.001) and the SGD optimizer with momentum, which helped stabilize its learning. The early stopping criterion was effective in balancing training time and model performance. Although AlexNet and MobileNet did not reach ResNet-18's performance, they represent viable options in environments where computational resources are limited.

7. Conclusion

This study provides a comprehensive comparison of CNN architectures for CIFAR-10, with ResNet-18 emerging as the best-performing model. Its residual connections allowed it to achieve superior accuracy on CIFAR-10, making it highly effective for complex image classification tasks. With a validation accuracy of 91.99% and a test accuracy of 72.79%, ResNet-18 is well-suited for image classification on smaller datasets with complex class structures.

While AlexNet and MobileNet were less accurate, they are valuable in applications prioritizing computational efficiency. These results provide insights into the trade-offs between accuracy and computational resources, with ResNet-18 recommended when accuracy is the priority, and AlexNet or MobileNet suited to resource-constrained applications.

Future Work

Further research could explore newer architectures like EfficientNet, which balances network depth, width, and resolution scaling for optimized accuracy and efficiency. EfficientNet's scalable design makes it an interesting candidate for CIFAR-10, where larger and more complex networks may achieve even higher accuracies.

Additionally, ensemble methods could be applied to combine the strengths of multiple architectures, potentially yielding higher accuracy by leveraging each model's unique learning characteristics. Exploring data augmentation techniques, such as mixup or CutMix, might also improve the generalization of CNN models, especially for smaller datasets like CIFAR-10.

Testing these models on larger, more complex datasets like ImageNet could provide deeper insights into their scalability and robustness. Further experimentation with transfer learning, where models pre-trained on large datasets are fine-tuned on CIFAR-10, might offer additional performance improvements.

References

[1] Review of Image Classification Algorithms Based on Convolutional Neural Networks

[2] Leiyu Chen, Shaobo Li, Qiang Bai, Jing Yang, Sanlong Jiang and Yanming Miao. Review of Image Classification Algorithms Based on Convolutional Neural Networks. *Remote Sensing*, 13(22), 2021.

[3] Alex Krizhevsky and Geoffrey Hinton. Learning Multiple Layers of Features from Tiny Image, 7, 2019.

[4] Aditya Thakur, Harish Chauhan and Nikunj Gupta. Efficient ResNets: Residual Network Design. *arXiv preprint arXiv:2306.12100v1*, 2023.

[5] Wenhao Tang, Junding Sun, Shuihua Wang and Yudong Zhang. Review of AlexNet for Medical Image Classification. *arXiv preprint arXiv: 2311.08655*, 2023.

[6] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto and Hartwig Adam. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv preprint arXiv:1704.04861v1*, 2017.

[7] Ruoyu Sun. Optimization for deep learning: theory and algorithms. *arXiv preprint arXiv:1912.08957v1*, 2019.