CIFAR-10 Image Classification

Yaoching CHI

Priyanka MOHAPATRA

Cynthia WIDJAJA

10/19/2024

California Science and Technology University

MB/CSE638 Deep Learning with Tensorflow

Shih-Yu Chang

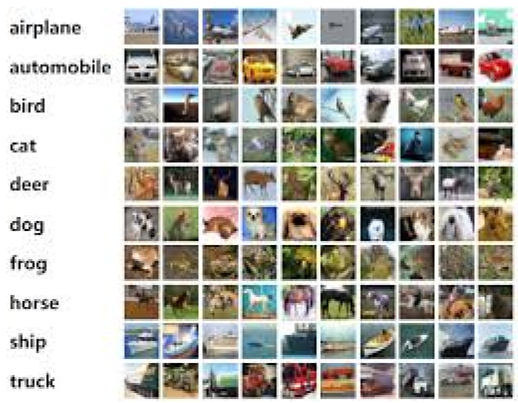
- ## Introduction

The CIFAR-10 dataset is a widely-used benchmark for evaluating computer vision models. It contains small, low-resolution images classified into 10 classes: airplanes, automobiles, birds, cats, deer, dogs, frogs, horses, ships, and trucks. The challenge lies in accurately distinguishing between similar categories (e.g., cats and dogs) while handling variations in poses, lighting, and other visual differences. The project aims to develop a CNN model to achieve high classification accuracy on this dataset.

● Dataset Overview

Dataset: CIFAR-10

- Number of Classes: 10



- Training Set Size: 48,000 images (80%)
- Test Set Size: 12,000 images (20%)
- Image Dimensions: 32x32 pixels, 3 color channels (RGB)

The dataset is pre-split into training and test sets, with an equal number of images in each class.

Methodology

Data Preprocessing:

Before feeding the data into the model, several preprocessing steps are taken to ensure optimal performance:

- Normalization: Each image's pixel values are scaled to a range of 0 to 1 by dividing by 255. This helps the model converge faster during training.

- The dataset is divided into 48,000 training images and 12,000 test images, making it a suitable benchmark for image classification tasks.
- Data Augmentation: Techniques such as rotation, flipping, zooming, and shifting are applied to the training data to artificially increase the size of the dataset and improve the model's generalization.

Training:

- Optimizer: The Adam optimizer was used to minimize the loss function.
- Batch Size: A batch size of 32 was used for training.
- Epochs: The model was trained for 5 epochs.
- Validation: 20% of the training data was set aside for validation to monitor the model's performance.

Splitting the Data:

- The dataset is split into two parts:
    - **Training Set (80%)**: 48,000 images are used to train the model.
    - **Test Set (20%)**: 12,000 images are used to evaluate the model's performance after training.

- Model Architecture:

  A CNN architecture was designed for this classification task, which includes:

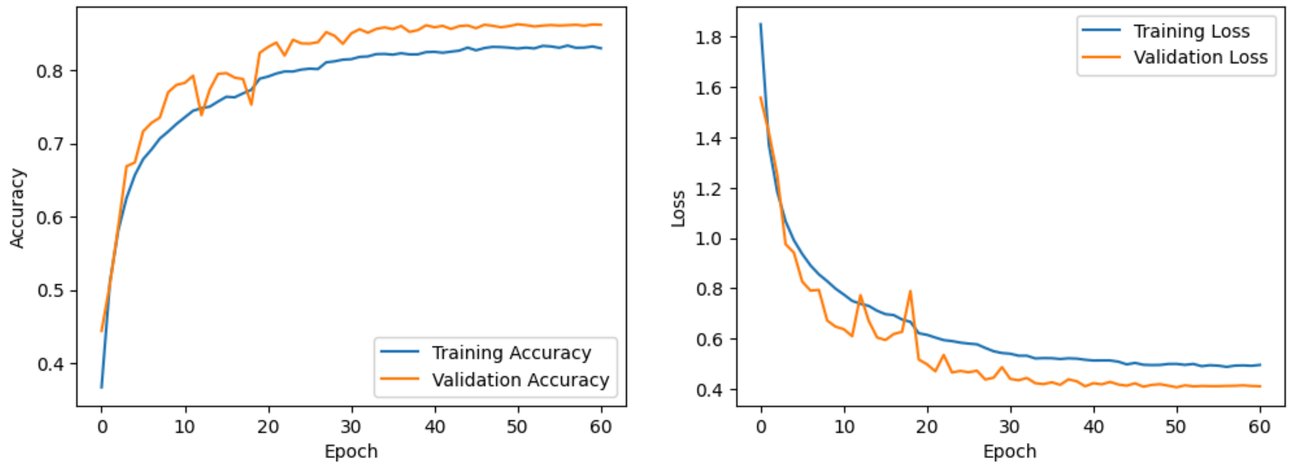
- Convolutional Layers: Three convolutional layers were employed to extract spatial features from the input images. The first layer used 32 filters, while the second and third layers used 64 filters, all with a kernel size of (3, 3).
- Max Pooling Layers: Reduce dimensionality while preserving important features.
- Batch Normalization: Improves training speed and stability by normalizing inputs in each layer.
- Dropout: Regularizes the network by randomly turning off some neurons during training, preventing overfitting.
- Fully Connected Layers: Perform classification based on the extracted features.
- Softmax Output Layer: Produces a probability distribution across the 10 classes.

- Training:
    - Optimizer: Adam optimizer
    - Loss Function: Categorical Crossentropy
    - Batch Size: 64
    - Epochs: 100
    - Validation Split: 20% of the training data was used for validation.
- Hyperparameter Tuning:
    - Learning Rate: Tuned to find the best rate for convergence without overshooting.
    - Dropout Rate: Adjusted to prevent overfitting.

- Number of Filters: Tuned across convolutional layers to find the optimal feature extraction configuration.

● Results and Evaluation

1. Training History:

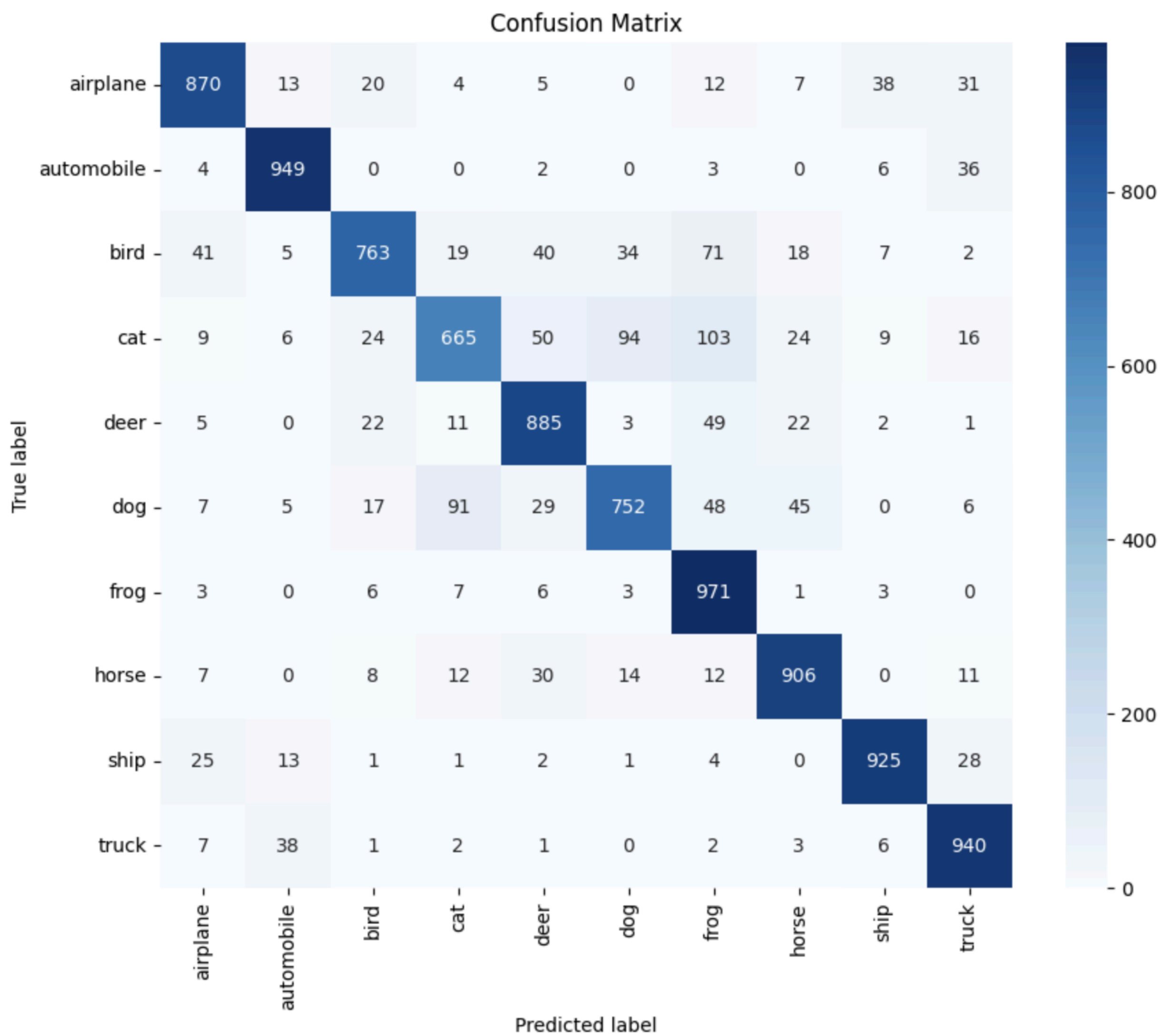


- Accuracy vs Epoch Graph
  - X-axis (Epochs)
    - Represents the number of training iterations. Each epoch refers to one complete pass of the entire training dataset through the model.
  - Y-axis (Accuracy)
    - The percentage of correctly classified images out of the total dataset.
  - Observation:
    - As the number of epochs increases, the training accuracy typically improves steadily, indicating that the model is learning from the training data.
- Loss vs Epoch Graph
  - X-axis (Epochs)
    - Represents the number of epochs or iterations of the model's learning process.
  - Y-axis (Loss)
    - A measure of the error between the predicted class probabilities and the true labels.
  - Observation:
    - As training progresses, the training loss decreases as the model's predictions get closer to the actual labels.

2. Confusion Matrix:

The confusion matrix provides insights into the model's ability to classify each of the 10 categories. The following key observations were made:

- High accuracy for categories like automobiles, ships, and trucks.
- Lower accuracy for more visually similar categories, such as cats and dogs, due to overlapping features.
- Misclassification primarily occurred between animals with similar visual features (e.g., bird vs. cat).



3. Classification Report:

Classification Report:

	precision	recall	f1-score	support
airplane	0.89	0.87	0.88	1000
automobile	0.92	0.95	0.94	1000
bird	0.89	0.76	0.82	1000
cat	0.82	0.67	0.73	1000
deer	0.84	0.89	0.86	1000
dog	0.83	0.75	0.79	1000
frog	0.76	0.97	0.85	1000
horse	0.88	0.91	0.89	1000
ship	0.93	0.93	0.93	1000
truck	0.88	0.94	0.91	1000
accuracy			0.86	10000
macro avg	0.86	0.86	0.86	10000
weighted avg	0.86	0.86	0.86	10000

4. Performance Analysis:

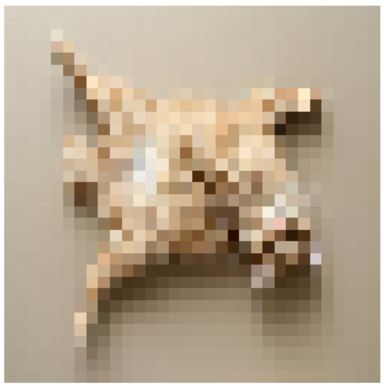
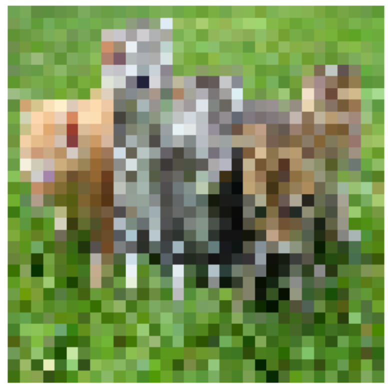
- High Precision and Recall: Classes such as automobiles, ships, and trucks performed well due to their distinct visual features.
- Challenges: Lower performance for animal categories, especially cats and dogs, was due to their visual similarity.
- Class Imbalance: Each class was balanced in terms of the number of images, which prevented bias toward any class.
- Low-Resolution Challenge: The model had to work with low-resolution images (32x32), making fine distinctions between classes difficult.

5. Visualize predictions:

• Challenges

1. Low Image Resolution

    ○ The 32x32 pixel size of the images limited the model's ability to detect fine-grained details.

2. Similar Class Features

    ○ Some classes, particularly animals like cats, dogs, and birds, share similar visual characteristics, which causes misclassifications.

3. Varied Poses

    ○ Objects in the images appear in varied orientations, making it harder for the model to generalize.

Broader Impact

This project showcases the power of Convolutional Neural Networks (CNNs) in tackling image classification challenges, even with low-resolution data like CIFAR-10. The broader impact extends beyond academic exercises and can be applied to several key fields, each with real-world significance:

1. Medical Imaging
   a. Application: CNNs are increasingly being used in medical image analysis to identify diseases from scans and imaging modalities like X-rays, MRIs, and CT scans. CNNs can also perform effectively on lower-resolution images or those with subtle differences between healthy and diseased tissues.
   b. Example: Cancer Detection
      - Challenge: Tumors may vary in size, shape, and location
      - Impact: Fast and accurate diagnoses, potentially lowering human error rates and improving early detection.
2. Autonomous Vehicles
   a. Application: Autonomous vehicles rely heavily on visual recognition systems to detect and classify objects on the road, such as pedestrians, other vehicles, traffic signs, and obstacles. These systems need to classify objects quickly and accurately for safe navigation.
   b. Example: Pedestrian Detection
      - Challenge: Pedestrians can appear at different orientations and postures, and may be partially occluded by other objects.
      - Impact: Enhances road safety, prevents collisions, and paves the way for fully autonomous driving systems.

3. Consumer Applications
    a. Application: Consumer-focused applications often rely on image recognition for tasks like facial recognition, product tagging in e-commerce, and content moderation on social platforms.
    b. Example: Facial Recognition
        - Challenge: Similar to the class confusion in CIFAR-10 (e.g., between cats and dogs), recognizing faces with similar features (e.g., family members with similar appearances) is challenging.
    c. Example: Product Tagging
        - Challenge: Cluttered images with multiple items may make product identification difficult.
    d. Impact: Enhanced user experience, personalized shopping, and efficient content moderation (e.g., identifying inappropriate content in social media).
4. Efficiency in Low-Resource Environments
    a. Perform well in areas with resource constraints, like healthcare in developing regions where medical imaging equipment may be limited.

● Conclusion

The CIFAR-10 classification project demonstrates the effectiveness of CNNs in managing complex image classification challenges, even when faced with low-resolution images and class similarity issues. With a classification accuracy of 86%, the model performed admirably across most classes, showcasing its potential for broader applications in fields like medical imaging and autonomous vehicles. However, distinguishing between similar animal classes presents opportunities for improvement. Techniques such as data augmentation and transfer learning could further enhance the model’s performance, paving the way for more robust solutions in real-world image recognition tasks.

● Future Model Improvement

1. Data Augmentation
    - Implement techniques like rotation, zooming, and flipping to enhance the diversity of the training dataset.
    - Impact: This increases the variability of the training data, making the model more robust to unseen variations in the images.
    - For example, augmented images will simulate real-world conditions, helping the model generalize better to new data.
2. Transfer Learning
    - Use pre-trained models like ResNet or VGG on CIFAR-10 to improve classification accuracy.
    - Impact: Transfer learning significantly reduces training time and often results in higher accuracy, especially when data is limited.
    - For example, using ResNet-50 pre-trained on ImageNet might improve accuracy on CIFAR-10 from 86% to over 90%, particularly in classes with subtle distinctions, like cats and dogs.
3. Ensemble Models
    - Combine predictions from multiple models to enhance classification performance.
    - Impact: By combining the strengths of different models, ensemble learning can provide higher accuracy and reduce overfitting. This can help correct class misclassifications and improve overall performance by using the collective "wisdom" of multiple models.

- # References

- Benenson, Rodrigo. https://rodrigob.github.io/are_we_there_yet/build/classification_datasets_results.html#43494641522d3130
- Cheng, Shuyang. AutoML: Automating the design of machine learning models for autonomous driving. https://waymo.com/blog/2019/01/automl-automating-design-of-machine/
- https://ijsra.net/sites/default/files/IJSRA-2023-0574.pdf
- Igbal, Saeed. On the Analyses of Medical Images Using Traditional Machine Learning Techniques and Convolutional Neural Networks. https://link.springer.com/article/10.1007/s11831-023-09899-9
- Krizhevsky, A. (2009). The CIFAR-10 dataset. https://www.cs.toronto.edu/~kriz/cifar.html
- https://paperswithcode.com/sota/image-classification-on-cifar-10
- Yadav, Samir. Deep convolutional neural network based medical image classification for disease diagnosis. https://journalofbigdata.springeropen.com/articles/10.1186/s40537-019-0276-2