Comparison of Decision Tree and CNN Models on CIFAR-10 Dataset

Xun Fu - r0928025

Yixuan Li - r0927212

Abstract

Our project explores the application of Decision Tree and Convolutional Neural Network (CNN) models to classify the CIFAR-10 dataset, a task aimed at understanding the strengths and limitations of traditional machine learning and deep learning approaches in image classification. While the Decision Tree model achieved a test accuracy of 23.65%, Random Forest reached an accuracy of 46.86%, CNN significantly outperformed it with an accuracy of 89.79%, highlighting the importance of spatial feature extraction in image classification tasks.

Introduction

Image classification is a fundamental task in computer vision, where models predict the category of an input image from a predefined set of classes. The CIFAR-10 dataset, comprising 60,000 labeled images across ten categories with 32×32 pixels, serves as an excellent benchmark for evaluating machine learning models. This study compares the performance of a traditional Decision Tree model and a deep learning CNN model to classify CIFAR-10 images.

The input to the Decision Tree model consists of features extracted as 16-bin normalized histograms for grayscale and RGB channels, along with the mean and standard deviation of grayscale values, and the input for random forest is flattened image data processed by PCA. CNN processes raw image data after normalization and augmentation. Outputs for the models are the predicted class labels of the images.

Related Work

Previous research on CIFAR-10 has predominantly focused on deep learning methods, with CNNs achieving state-of-the-art performance. Traditional machine learning approaches such as Decision Tree, have shown limited success due to their inability to exploit the spatial structure of image data effectively and tendency to overfitting. Ensemble methods like Random Forests provide improvements by combining multiple trees, but they remain less effective than CNNs, which leverage convolutional operations to capture hierarchical patterns in data.

Dataset and Features

The CIFAR-10 dataset consists of 60,000 color images of size 3×32×32 pixels. The dataset is divided into a training set (40,000 images), validation set (10,000 images), and test set (10,000 images). Preprocessing steps include:

1. **Decision Tree:**
    a. Extract normalized histograms for grayscale and RGB channels.
    b. Extract mean and standard deviation of grayscale values.
    c. Combine the extracted features into 66-dimensional feature vectors..
2. **Random Forest:**:
    a. Flattening images into 1D arrays.
    b. Standardizing features using StandardScaler.
    c. Reducing dimensionality via PCA, retaining 95% variance.
3. **CNN**:
    a. Normalizing pixel values to [0, 1].
    b. Augmenting data through rotations, shifts, flips, and zooms.
    c. One-hot encoding of labels.

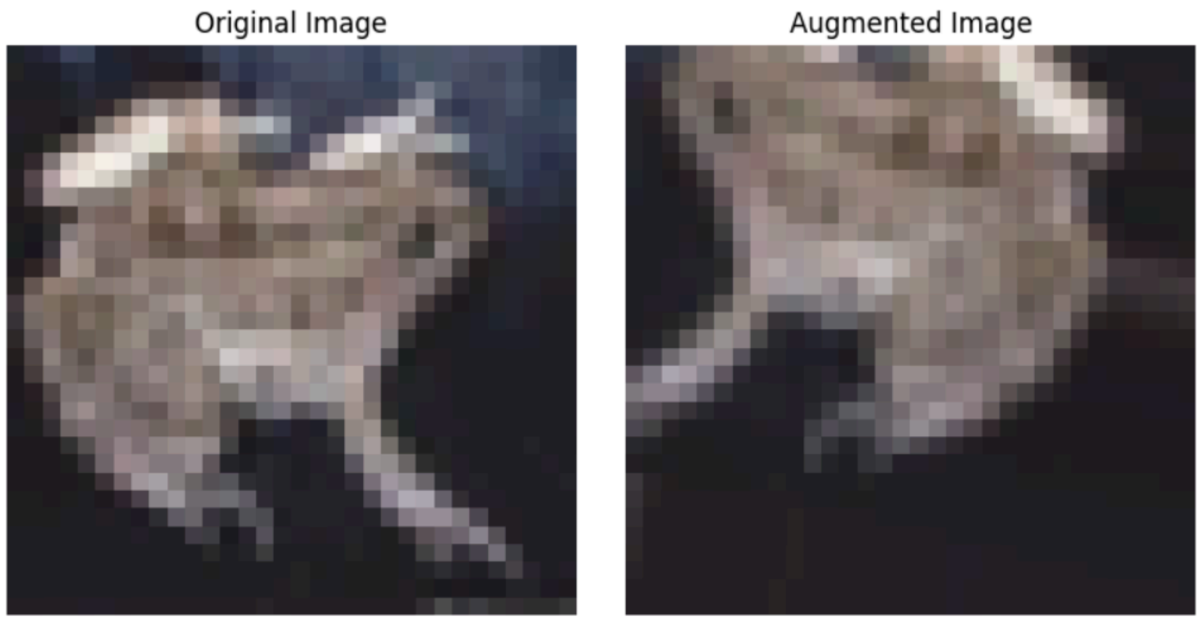

Figure 1. Original image(left) and Augmented Image(right)

Methods

Decision Tree

A Decision Tree classifier was implemented with the following workflow:

- **Feature Selection**: Selected the best feature for splitting based on information gain.
- **Training**: Fit the tree using a maximum depth of 6 to prevent overfitting.
- **Evaluation**: Assessed on test sets.

In the decision tree model, we used code in the assignment to calculate information gain and best feature.

$$H(p_1) = -p_1 \log_2(p_1) - (1 - p_1)\log_2(1 - p_1)$$

$$\text{Information Gain} = H(p_1^{\text{node}}) - (w^{\text{left}} H(p_1^{\text{left}}) + w^{\text{right}} H(p_1^{\text{right}}))$$

Formula 1. Formula of Entropy and Information Gain

Random Forests were also applied to leverage ensemble learning, combining predictions from multiple trees trained on random subsets of data by using the result of majority vote.

- **Feature Input**: The input of random forest is flattened 1D arrays processed with PCA.
- **Training**: Used 200 trees with depth of 20, minimal sample split of 5, and minimal sample leaf of 3.
- **Evaluation**: Assessed on test sets.

CNN

The CNN model architecture included:

- **Convolutional Layers**: Used ReLU activations, Batch Normalization, and MaxPooling to extract spatial features.
- **Fully Connected Layers**: Included Dropout and regularization to reduce overfitting.
- **Output Layer**: A softmax activation function predicted class probabilities.

Training was performed using SGD with a learning rate scheduler and early stopping based on validation loss.

Cross-Entropy Loss is used as the output layer of CNN model, which is defined as below:

$$logloss = -\frac{1}{N}\sum_{i}^{N}\sum_{j}^{M} y_{ij}\log(p_{ij})$$

Formula 2. Cross-entropy loss function

In this cross-entropy loss above, N is the number of rows, M is the number of classes.

Softmax is used as the cost function of CNN model, which is defined as below:

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

Formula 2. Softmax function

Results and Discussion

Decision Tree

Decision Tree Accuracy	Random Forest Accuracy
23.65%	46.86%

Decision Tree struggled with the high-dimensional nature of CIFAR-10 data, as they rely on single feature splits that fail to capture spatial relationships. Random Forests showed improvement due to ensemble averaging but remained inadequate compared to CNN.

CNN

- Test Accuracy: **89.79%**

CNN's ability to learn hierarchical representations of image features resulted in significantly better performance. Training and validation accuracy curves (Figure 2) indicate effective convergence without overfitting.

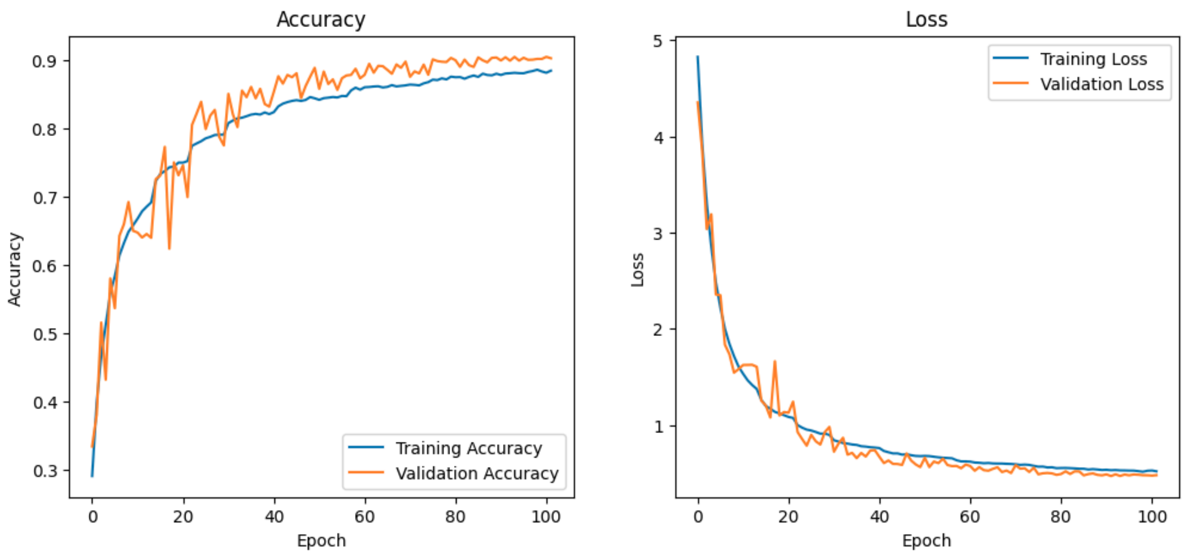

Figure 2. Accuracy and Loss curve of CNN model

Key Insights

- Traditional models like Decision Trees and Random Forests are limited for image data due to their inability to exploit spatial patterns.
- Deep learning models, especially CNNs, are well-suited for image classification tasks, thanks to their hierarchical feature extraction capabilities.

Conclusion and Future Work

This study demonstrates the significant advantages of CNNs over traditional machine learning models for image classification. The CNN model’s ability to learn and utilize spatial and hierarchical features of images enables it to achieve a much higher accuracy (89.79%) compared to Decision Trees (23.65%) and Random Forests (46.86%). This superiority highlights the necessity of leveraging advanced feature extraction techniques for tasks involving high-dimensional structured data like images. Additionally, CNN's flexibility in architectural design and optimization makes it an ideal choice for complex computer vision tasks.

Future research could focus on several avenues to improve and extend the current work:

1. **Better Feature Extraction:** In this project the feature input to the decision tree is processed by hand. If CNN is implemented to select the features, the accuracy of the decision tree can be further improved.
2. **Exploring Deeper Architectures**: Investigate advanced CNN architectures like ResNet, DenseNet, or EfficientNet to further enhance classification performance.
3. **Hyperparameter Tuning**: Conduct extensive experiments with different learning rates, optimizers, dropout rates, and regularization techniques to identify the optimal configuration.
4. **Transfer Learning**: Leverage pre-trained models on larger datasets (e.g., ImageNet) to reduce training time and improve performance.

Contributions

Yixuan Li: Implemented Decision Tree and Random Forest models, performed PCA and feature selection, and analyzed results.

Xun Fu: Developed CNN architecture, conducted data augmentation and training, and created visualizations.

Reference

[1] Krizhevsky, A., Sutskever, I., Hinton, G. E. (2012). *ImageNet Classification with Deep Convolutional Neural Networks*.