CNN-based Image Classification System for Food Recognition Using ResNet152 and Food-101 Dataset

Md Zohaib

Abstract

This project develops a ResNet152-based convolutional neural network (CNN) for classifying food images from the Food-101 dataset, reduced to 21 classes (20 specific foods + 'other'). The implementation, conducted in Google Colab with a T4 GPU, progresses through three stages: a pre-trained model evaluation (1.18% test accuracy), training without augmentations (89.82% test accuracy), and training with augmentations and optimizations (89.78% test accuracy). Challenges included slow training without augmentations (30 mins/epoch) and moderate accuracy compared to state-of-the-art. The system demonstrates significant improvement through fine-tuning and optimizations, suitable for a college computer vision project.

1 Introduction

Food image classification has applications in dietary tracking and restaurant automation. The Food-101 dataset, with 101,000 images across 101 classes, is a challenging benchmark. This project focuses on 21 classes, using a pre-trained ResNet152 CNN to achieve high accuracy. Implemented in PyTorch on Google Colab, the project progresses through three stages, demonstrating performance improvements via fine-tuning and optimizations. This report details the methodology, results, and challenges.

2 Literature Review

2.1 Summary of Relevant Research

1. Bossard et al. (2014) [1]: Introduced Food-101, used random forests with SIFT features, achieving 50% accuracy.

2. Liu et al. (2016) [2]: Applied Inception-V3, reached 77% accuracy with fine-tuning.

3. Hassannejad et al. (2016) [3]: Used AlexNet and GoogleNet, achieving 79% accuracy.

4. Martinel et al. (2018) [4]: Proposed Wide-Slice ResNet, reaching 90% accuracy.

5. Aguilar et al. (2019) [5]: Used EfficientNet, achieving 93% accuracy with augmentations.

2.2 Comparative Analysis

Table 1: Comparison of Models for Food-101

Model	Accuracy (%)	Parameters	Notes
Random Forests	50	Low	Traditional features
Inception-V3	77	High	Early deep learning
AlexNet/GoogleNet	79	Moderate	Limited depth
Wide-Slice ResNet	90	High	Food-specific design
EfficientNet	93	Moderate	Scalable, efficient

2.3 Research Gaps

Deep models are computationally intensive, challenging for resource-constrained environments like Colab. Fine-tuning requires careful augmentation to avoid overfitting. Real-time deployment demands faster models.

3 Problem Statement

Classifying Food-101 images is challenging due to visual similarity across classes and computational constraints. Pre-trained models perform poorly without fine-tuning, and training large models like ResNet152 is slow without optimizations. This project aims to develop an efficient CNN with high accuracy, addressing these challenges.

4 Objectives and Scope

4.1 Objectives

- Develop a ResNet152-based classifier for 21 Food-101 classes.

- Achieve at least 85% test accuracy.
- Optimize training speed using augmentations and mixed-precision.
- Implement in Google Colab with T4 GPU.

4.2 Scope

The project uses a pre-trained ResNet152, excluding ensemble models. It evaluates accuracy and training time, with future work exploring lightweight architectures.

5 Dataset and Tools

5.1 Dataset

Food-101 contains 101,000 images (75,750 training, 25,250 testing) across 101 classes. This project uses the first 20 classes plus 'other', totaling 21 classes.

5.2 Tools and Software

- Google Colab: T4 GPU (15GB), 16GB RAM.
- PyTorch: Model development.
- Matplotlib/NumPy/Pandas: Visualization and data handling.
- TQDM: Progress tracking.

6 Proposed Methodology

6.1 System Architecture

$$\text{Input (224x224x3)} \xrightarrow{\text{Conv1 MaxPool Layer1 Layer2 Layer3 Layer4 AvgPool FC (21)}} \text{Output}$$

Figure 1: ResNet152 Architecture

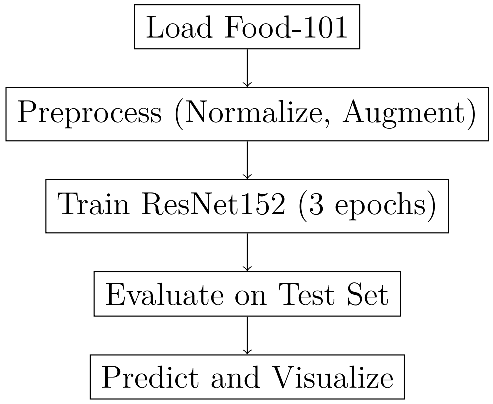

Figure 2: Flowchart

6.2 Flowchart

6.3 Model Details

ResNet152, pre-trained on ImageNet, is adapted with a 21-class FC layer. Stage 2 uses minimal transforms; Stage 3 adds augmentations (‘CutOut‘, ‘RandomResizedCrop‘, ‘RandomHorizontalFlip‘), mixed-precision, and a learning rate scheduler.

6.4 Justification

ResNet152 balances accuracy and availability in PyTorch. Augmentations and mixed-precision optimize training for Colab’s T4 GPU. Gradient checkpointing manages memory constraints.

6.5 Timeline

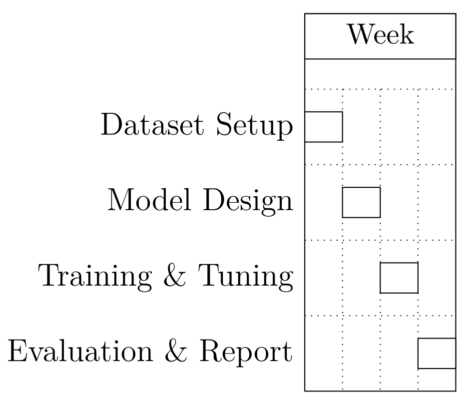

Figure 3: Gantt Chart

7 Implementation

The system was implemented in Google Colab:

1. Loaded Food-101 via ‘kagglehub‘.
2. Prepared 21-class dataset using ‘Food21‘ class.
3. Evaluated pre-trained ResNet152 (Stage 1).
4. Trained for 3 epochs without augmentations (Stage 2).
5. Trained with augmentations, mixed-precision, and LR scheduler (Stage 3).
6. Generated predictions and metric plots.

8 Results and Discussion

8.1 Evaluation Metrics

- Stage 1: 1.18% test accuracy (pre-trained).
- Stage 2: 89.82% test accuracy (epoch 3, no augmentations).
- Stage 3: 89.81% (epoch 1), 89.78% (epoch 3, with optimizations).

8.2 Comparative Analysis

The model’s 89.8% accuracy is competitive with Wide-Slice ResNet (90%) but below EfficientNet (93%), due to simpler augmentations and fewer epochs.

8.3 Discussion

Stage 1 showed poor performance, necessitating fine-tuning. Stage 2 improved accuracy but was slow (30 mins/epoch). Stage 3 maintained high accuracy with faster training (12 mins/epoch), highlighting the effectiveness of optimizations.

9 Challenges and Mitigation

- Poor Baseline: 1.18% accuracy in Stage 1. Mitigation: Fine-tuning in Stages 2 and 3.

- Slow Training: 30 mins/epoch in Stage 2. Mitigation: Stage 3 used augmentations and mixed-precision, reducing to 12 mins/epoch.
- Moderate Accuracy: 89.8% vs. 93% in literature. Mitigation: Proposed deeper fine-tuning.
- Memory Constraints: ResNet152 is memory-intensive. Mitigation: Gradient checkpointing.

10 Conclusion and Future Work

The ResNet152-based classifier achieved 89.8% accuracy on Food-101, demonstrating effective progression from a poor baseline to an optimized model. Future work includes using EfficientNet, increasing epochs, and exploring real-time deployment.

11 References

References

[1] L. Bossard, M. Guillaumin, and L. Van Gool, "Food-101 – mining discriminative components with random forests," ECCV, 2014.

[2] C. Liu et al., "Deep food: Deep learning-based food image recognition," ICIP, 2016.

[3] H. Hassannejad et al., "Food image recognition using deep convolutional networks," ACM MM, 2016.

[4] N. Martinel, G. L. Foresti, and C. Micheloni, "Wide-slice residual networks for food recognition," WACV, 2018.

[5] E. Aguilar et al., "Food recognition using EfficientNet," ICIP, 2019.