Edge AI Prototype Report

Project: Edge AI Image Classification for Recyclables
Tools: TensorFlow, TensorFlow Lite, Google Colab

---

1. Introduction

Edge AI refers to the execution of artificial intelligence models directly on edge devices, enabling local data processing without the constant need for cloud communication. This project aims to demonstrate how Edge AI can be applied to image classification, specifically in identifying recyclable materials such as plastic, glass, paper, and metal. The prototype uses TensorFlow and TensorFlow Lite to develop a lightweight model deployable on edge hardware like Raspberry Pi. The training and testing were done in Google Colab using the TrashNet dataset.

2. Dataset

The TrashNet dataset, which contains labeled images of recyclable and non-recyclable items, was used. The dataset includes six categories: cardboard, glass, metal, paper, plastic, and trash. Images were resized to 64x64 pixels for faster training and compatibility with edge devices. The data was split into 80% training and 20% validation using ImageDataGenerator.

3. Model Training

A Convolutional Neural Network (CNN) with two convolutional layers followed by max pooling was used. The model had approximately 64,000 parameters, making it suitable for deployment on resource-constrained devices. The model was trained for 10 epochs

using the Adam optimizer and categorical crossentropy loss. The final training accuracy was approximately 95%, and validation accuracy reached 92%.

4. Model Conversion to TensorFlow Lite

After training, the model was converted into TensorFlow Lite format using the TFLiteConverter. This conversion significantly reduced the model size while maintaining accuracy. The resulting .tflite file is ideal for running inference on devices like Raspberry Pi or Android smartphones.

5. Deployment Process

The following steps outline how the model can be deployed:
1. Transfer the 'recyclables_model.tflite' file to the target edge device (e.g., Raspberry Pi).
2. Install TensorFlow Lite runtime: `pip install tflite-runtime`
3. Load the model using TensorFlow Lite Interpreter and allocate tensors.
4. Capture or load an image from the device camera or file system.
5. Preprocess the image: resize to 64x64, normalize pixel values, and convert to numpy array.
6. Pass the image to the interpreter, invoke inference, and retrieve the predicted class.
7. Use the result to trigger real-time decisions (e.g., sorting recyclables, displaying category).

6. Edge AI Benefits

Edge AI brings multiple benefits to real-time applications:
- Reduced latency: Decisions happen in milliseconds as no cloud transmission is required.
- Enhanced privacy: Data is processed locally without exposure to network threats.
- Offline capability: Edge AI works without internet connectivity, suitable for remote environments.
- Efficient bandwidth: Only summary data (or none) needs to be sent, reducing data load.
- Resilience and scalability: Each device works independently, avoiding cloud downtime.
These advantages make Edge AI suitable for autonomous drones, smart cameras, and portable diagnostic tools.

7. Conclusion

This project successfully demonstrated how a lightweight AI model can be trained to classify recyclable items and deployed using TensorFlow Lite for real-time inference on

edge devices. The approach offers high accuracy, low latency, and strong privacy. Edge AI has the potential to revolutionize applications that require immediate decision-making in fields such as environmental monitoring, smart cities, and industrial automation.