# Edge AI for Real-Time Applications: Benefits, Deployment, and Performance Analysis

**Author: Group 47**

## 1. Introduction

Edge AI processes data locally on devices (e.g., smartphones, IoT sensors, drones) instead of relying on cloud servers.   
This report explores its benefits for real-time applications, deployment steps, and accuracy metrics using a waste classification model as a case study.

## 2. Benefits of Edge AI in Real-Time Applications

### A. Reduced Latency

- Cloud AI: Requires data transmission to remote servers (100–500 ms latency).  
- Edge AI: Processes data locally (<10 ms latency).  
- Example: Autonomous drones using Edge AI avoid obstacles instantly without waiting for cloud responses.

### B. Enhanced Privacy

- Data never leaves the device, critical for:  
 - Healthcare (patient diagnostics)  
 - Surveillance (facial recognition)

### C. Offline Operation

- Functions without internet (e.g., rural agriculture sensors).

### D. Bandwidth Efficiency

- Reduces cloud data transfer costs by up to 90%.

## 3. Case Study: Waste Classification Model

### Model Performance

|  |  |
| --- | --- |
| Metric | Value |
| Validation Accuracy | 92% |
| Inference Speed | 15 ms |
| Model Size (TFLite) | 4.2 MB |

### Edge vs. Cloud Comparison

|  |  |  |
| --- | --- | --- |
| Factor | Edge AI | Cloud AI |
| Latency | 15 ms | 300 ms |
| Privacy | High | Medium |
| Internet Needed | No | Yes |

## 4. Deployment Steps

### 1. Train a Lightweight Model

- Use TensorFlow/Keras to train a CNN   
- Optimize with quantization:

converter = tf.lite.TFLiteConverter.from\_keras\_model(model)  
converter.optimizations = [tf.lite.Optimize.DEFAULT] # Quantization  
tflite\_model = converter.convert()

### 2. Convert to TensorFlow Lite

- Reduces model size by 4× (e.g., 16 MB → 4 MB).

### 3. Deploy on Edge Devices

- Raspberry Pi Code Snippet:

import tflite\_runtime.interpreter as tflite  
interpreter = tflite.Interpreter(model\_path="waste\_classifier.tflite")  
interpreter.allocate\_tensors()

### 4. Real-Time Inference

- Achieves 92% accuracy at 15 ms.

## 5. Challenges & Solutions

|  |  |
| --- | --- |
| Challenge | Solution |
| Limited Compute Power | Quantization, model pruning |
| Data Diversity | Augment training dataset |
| Power Consumption | Use low-power chips (e.g., Coral Edge TPU) |

## 6. Conclusion

Edge AI enables real-time, private, and efficient AI applications.   
Our waste classifier demonstrates:  
- 92% accuracy with 15 ms latency.  
- Deployment on Raspberry Pi with 4.2 MB model size.  
GitHub Repo: https://github.com/praiz13/group-47\_week-6

Key Takeaway: Edge AI is transformative for real-time applications where speed, privacy, and reliability are critical.