CleanTech - Dataset Collection & Processing Report

Collect the Dataset

There are many popular open sources for collecting data (e.g., Kaggle, UCI repository).

In this project, we used three classes of images:

- **Biodegradable**
  -
  -
- **Recyclable**
  -
  -
- **Trash**
  -

The dataset was downloaded from Kaggle.
Link: *Dataset*

Once downloaded, the data was unzipped and visualized using tools like `matplotlib`, `pandas`, and IPython.

Note:

While there are multiple ways to understand datasets (EDA, statistical summaries, class imbalance checks), this project focused on image-level analysis using visualization and predictions.

Activity 1.1: Importing Libraries

python
CopyEdit

```
import tensorflow as tf  import keras  import pandas as pd  import
numpy as np  import matplotlib.pyplot as plt  import os, random
```

Activity 1.2: Read the Dataset

Data formats supported: `.csv`, `.json`, `.txt`, `.zip`

Steps:

1.

Unzip the dataset.

2.
3.

Load image paths.

4.
5.

Use `pandas` for any metadata (if applicable).

6.
7.

Organize files into class-wise folders.

8.

Data Visualization

Python scripts used to:

-

Randomly select images from folders.

-
-

Display them using IPython.

-

Each test run correctly identified the class of the following:

- ✅ **Biodegradable**
- ✅ **Recyclable**
- ✅ **Trash**

This confirmed model functionality on unseen examples.

Prediction Flow (All Classes)

- The model uses random sampling from a specified `folder_path`.
- IPython displays the image.
- Model predicts based on VGG16 features.
- Correct class is shown with the image.

Example classes tested:

- Biodegradable → predicted ✅

-
-

Recyclable → predicted ✅

-
-

Trash → predicted ✅

-

Data Augmentation

Although data augmentation techniques such as:

-

Rotation

-
-

Flipping

-
-

Brightness/contrast changes
…are useful in improving accuracy,

-

In this case, the dataset was already augmented and preprocessed before training. Therefore, augmentation was skipped in training, and the model maintained acceptable accuracy.
Note: Training time increased slightly due to larger input size.