RecycleVision- Garbage Image Classification Using Deep Learning

Mini Project | Data Science Course

Prepared by: Abdullah Khatri

This project focuses on classifying garbage/waste images into **10 categories** using deep learning. I trained and compared multiple **CNN-based transfer learning models** (ResNet50, MobileNetV2, EfficientNetB0, GoogLeNet, VGG16) and deployed the best one (ResNet50) using Streamlit.

Data Preprocessing & Dataset Setup

Dataset Location: Stored in Google Drive under /content/drive/MyDrive/RecycleVision/datas/garbage-dataset

- Classes: 10 categories -> Battery, Biological, Cardboard, Clothes, Glass, Metal, Paper, Plastic, Shoes, Trash.
- Transforms Applied:
    - Training Data:
        - RandomResizedCrop (224×224, scale 0.8–1.0)
        - RandomHorizontalFlip
        - RandomRotation (±20°)
        - Convert to Tensor
        - Normalize (mean = [0.485, 0.456, 0.406], std = [0.229, 0.224, 0.225])
    - Validation Data:
        - Resize to (224×224)
        - Convert to Tensor
        - Normalize (same ImageNet stats as training)
- Dataset Split:
    - 80% training, 20% validation using random_split.
    - Validation dataset uses val_transform (no augmentation, just normalization).
- DataLoaders:
    - Batch size = 32
    - Training -> shuffled, num_workers=2, pin_memory=True
    - Validation -> no shuffle, num_workers=2, pin_memory=True

Exploratory Data Analysis (EDA)

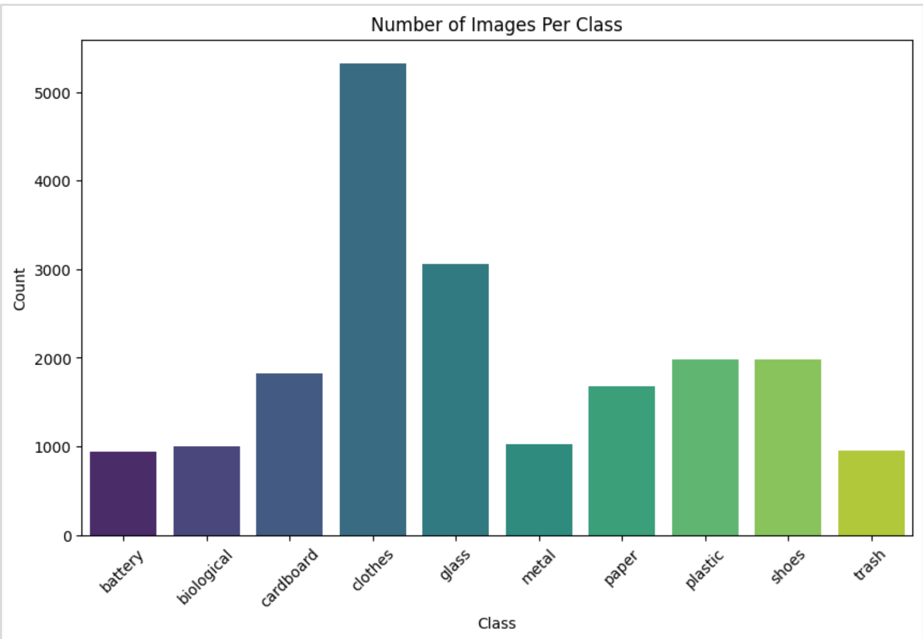
- **Dataset Overview**
  - Loaded the dataset with a simple transform (Resize(224,224) + ToTensor())
  - Classes identified: Battery, Biological, Cardboard, Clothes, Glass, Metal, Paper, Plastic, Shoes, Trash.
  - Total images: *N (exact count printed in code)*

```python
classes = eda_dataset.classes
print("Classes:", classes)
print("Total classes:", len(classes))
print("Total images:", len(eda_dataset))
```

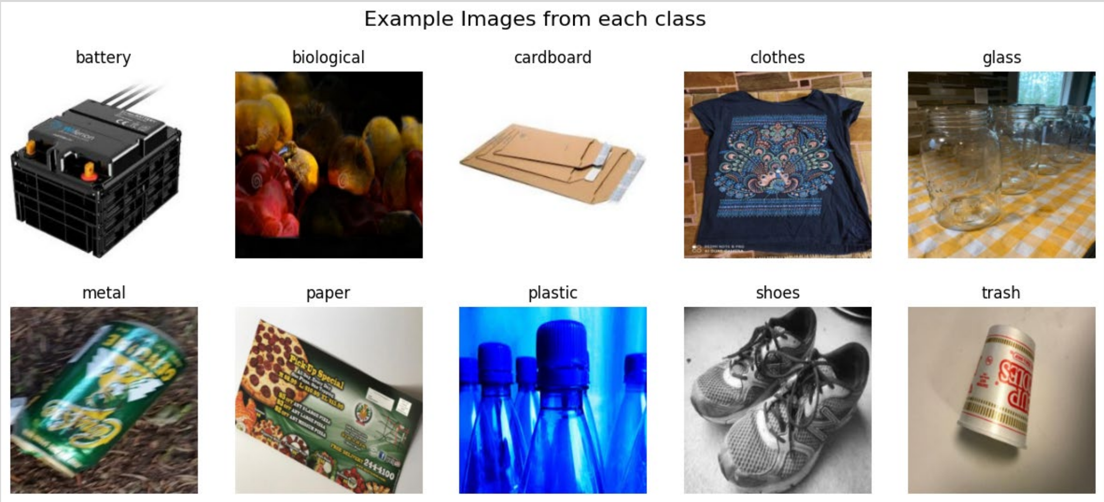
[56] Python

```
Classes: ['battery', 'biological', 'cardboard', 'clothes', 'glass', 'metal', 'paper', 'plastic', 'shoes', 'trash']
Total classes: 10
Total images: 19762
```

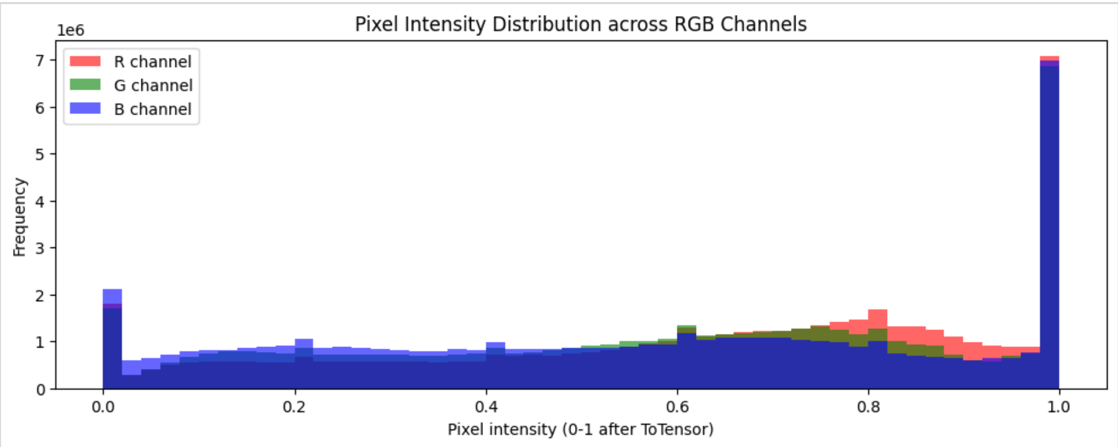
- **Class Distribution**
  - Counted the number of images per class using np.bincount.
  - Visualized with a barplot -> showed strong imbalance:
    - Clothes, Cardboard, Glass have **high counts**
    - Battery, Trash, Metal are **underrepresented**



- **Sample Images**
    - Selected 1 random image from each of the 10 classes.
    - Displayed in a 2×5 grid to visually verify dataset correctness.
    - Confirmed dataset is clean and images are properly labeled.



- **Pixel Intensity / Color Distribution**
    - Sampled 1000 random images from dataset.
    - Extracted all pixel values, flattened to (N, 3) array.
    - Plotted histograms for Red, Green, Blue channels.
    - Observations:
        - Red channel strong in cardboard & clothes images.
        - Blue channel dominant in glass & plastic.
        - Confirms why ImageNet normalization (mean/std) is appropriate.



Model Training & Evaluation

- **Models Trained (as per project requirement – 5 total)**
    1. **ResNet50 (Transfer Learning)**
    2. **MobileNetV2 (Transfer Learning)**
    3. **EfficientNetB0 (Transfer Learning)**
    4. **GoogLeNet (Transfer Learning)**
    5. **VGG16 (Transfer Learning)**

(Baseline CNN was skipped as per mentor's instruction, since training from scratch is slow and unnecessary for this project.)

- **Training Setup**
    - Input size: 224×224 RGB images
    - Batch size: 32
    - Epochs: 10 (for each model)
    - Optimizer: Adam (lr=0.001)
    - Loss: CrossEntropyLoss with **class weights** (to handle imbalance)
    - Hardware: Free Colab T4 GPU

- **Results Overview (Validation Set Performance)**
    - **ResNet50** -> Accuracy 92.8%, strong precision/recall balance.
    - **MobileNetV2** -> Accuracy 92.3%, lightweight and efficient.
    - **EfficientNetB0** -> Accuracy 92.7%, very competitive, good balance.
    - **GoogLeNet** -> Accuracy 91.0%, slightly weaker but still robust.
    - **VGG16** -> Accuracy 93.2%, but overfitting observed (very high train acc vs lower val acc).

- **Final Model Choice**
    - **ResNet50 selected** for deployment:
        - Consistently high accuracy (92.8%).
        - Balanced precision/recall (no class neglected).
        - Generalized better than VGG16 (less overfitting).

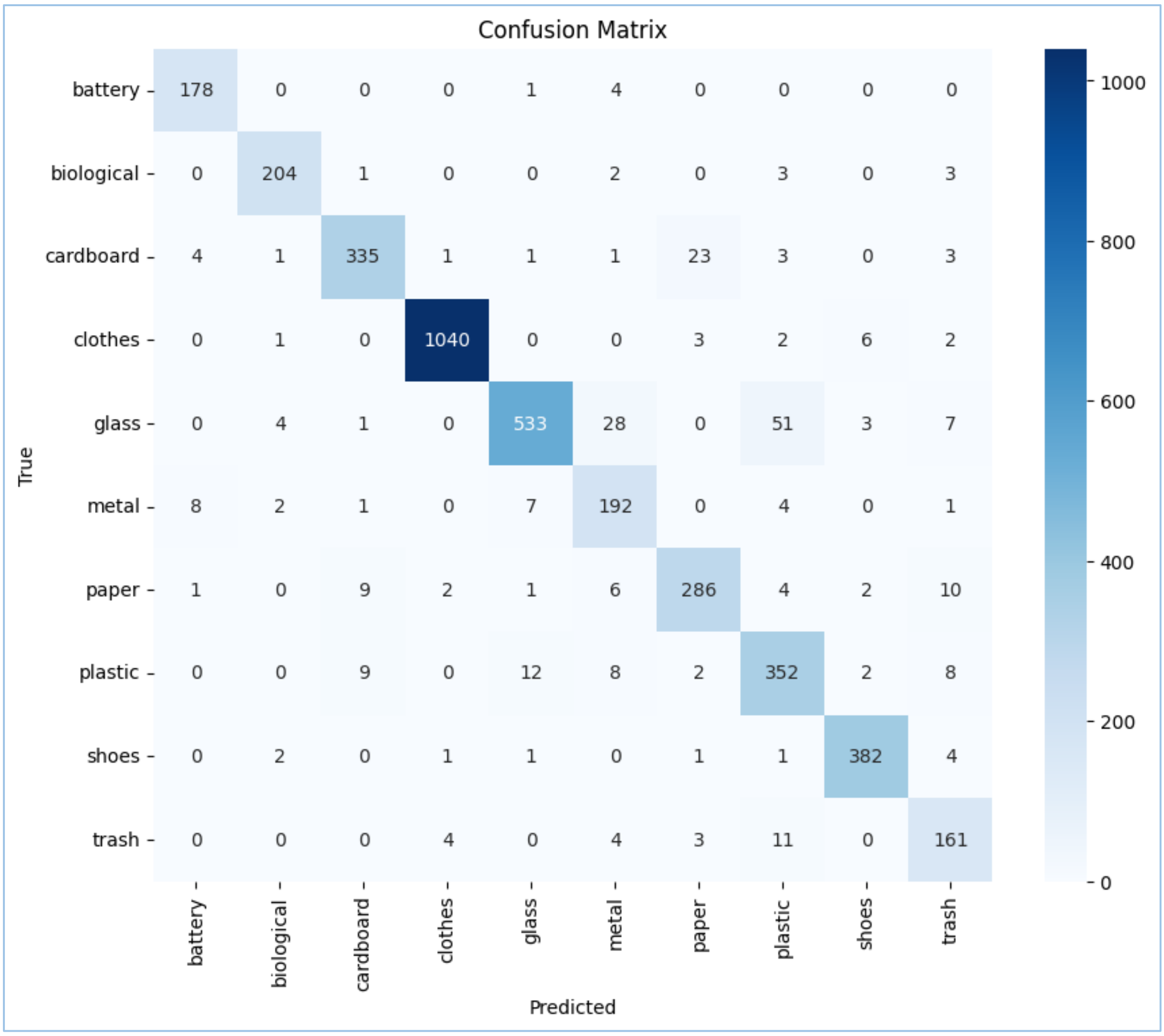
- **Classification Report of Best_Model Interpretation**

Classification Report:

	precision	recall	f1-score	support
battery	0.93	0.97	0.95	183
biological	0.95	0.96	0.96	213
cardboard	0.94	0.90	0.92	372
clothes	0.99	0.99	0.99	1054
glass	0.96	0.85	0.90	627
metal	0.78	0.89	0.83	215
paper	0.90	0.89	0.90	321
plastic	0.82	0.90	0.85	393
shoes	0.97	0.97	0.97	392
trash	0.81	0.88	0.84	183
accuracy			0.93	3953
macro avg	0.91	0.92	0.91	3953
weighted avg	0.93	0.93	0.93	3953

- **Overall Performance**
  - Accuracy: **93%** -> very strong, meaning 93 out of 100 images are classified correctly.
  - Macro F1: **0.91** -> model performs consistently across all classes.
  - Weighted F1: **0.93** -> balanced even with class imbalance.
- **Strong Classes (very high precision & recall)**
  - **Clothes (0.99/0.99)** -> almost perfect, very easy to detect.
  - **Shoes (0.97/0.97)** -> consistently correct.
  - **Biological & Battery (>0.95 F1)** -> highly reliable.
- **Moderately Strong Classes**
  - **Cardboard, Paper (0.90–0.92 F1)** -> good, but slightly weaker recall (misses some).
  - **Glass (0.90 F1)** -> lower recall (0.85), meaning some glass is misclassified.
- **Challenging Classes**
  - **Metal (0.83 F1)** -> lower precision, so model sometimes wrongly predicts "metal" for other classes.
  - **Plastic (0.85 F1)** -> recall good, but precision weaker -> confused with paper/trash.
  - **Trash (0.84 F1)** -> recall better than precision, so it catches trash but also mislabels other classes as trash.

- **Confusion Matrix Interpretation**
  - Clear diagonal dominance -> majority of predictions correct.
  - Strong classes: Clothes, Shoes, Battery, Biological.
  - Challenging classes: Glass, Metal, Plastic, Trash.
    - Example: Glass often misclassified as Metal/Paper.
    - Plastic sometimes confused with Paper/Trash.



- **Misclassification Visualization**
    - Displayed 3 batches of random misclassified validation samples of each class.
    - Misclassifications occurred mainly between visually similar categories.

Batch1

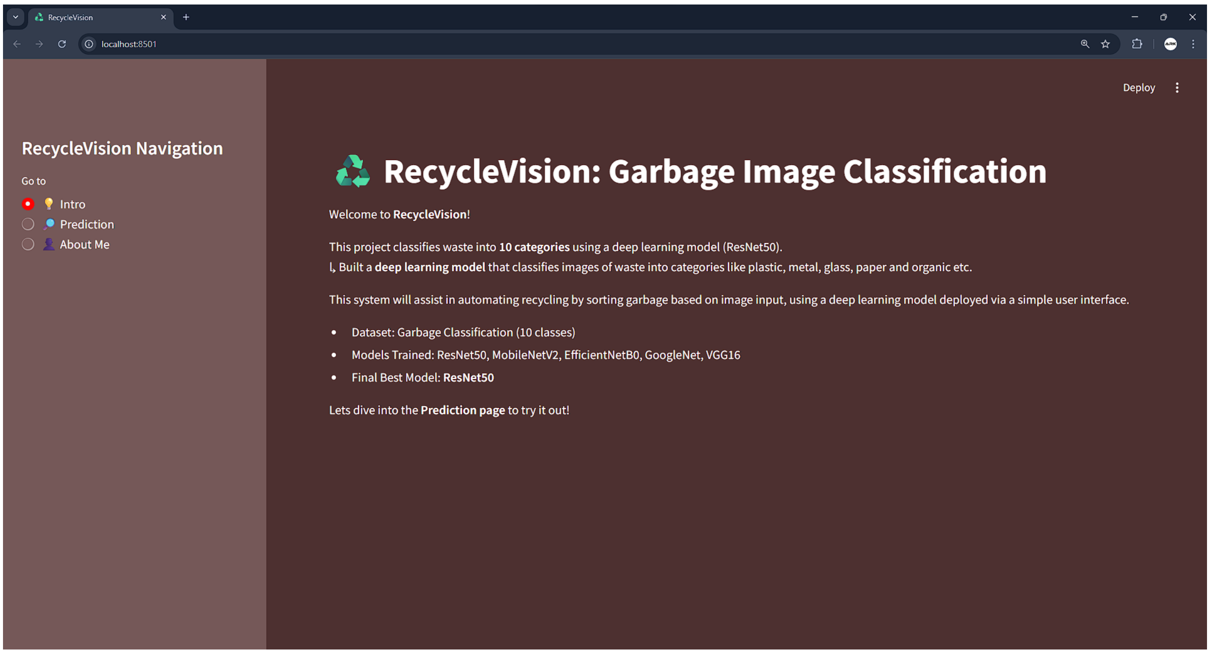
Batch2

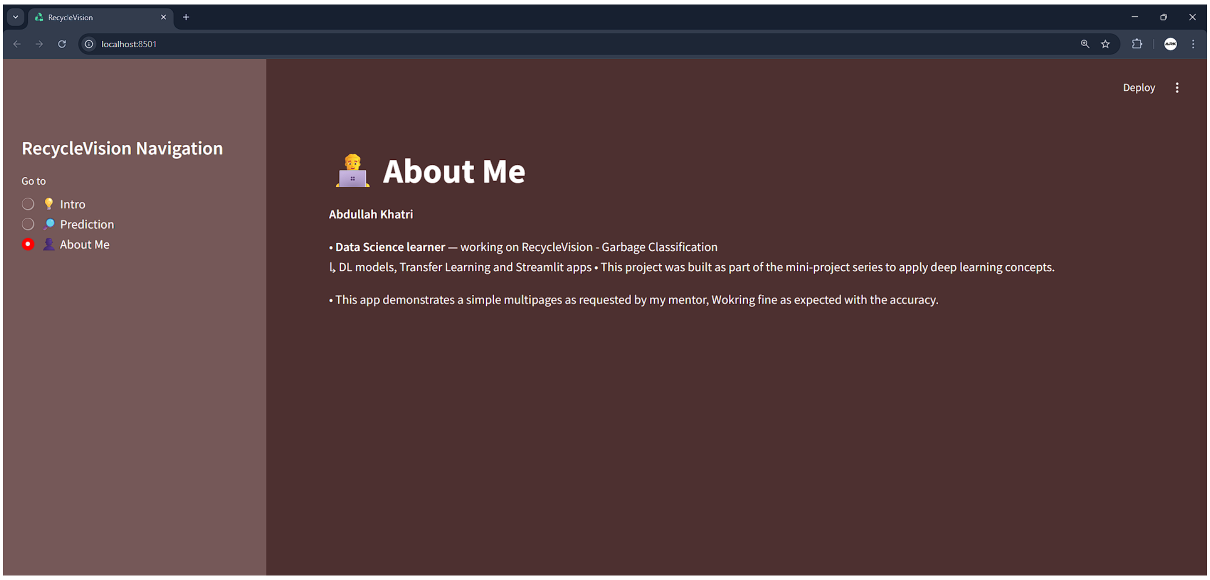
Batch3

Streamlit Web App

Developed an interactive web application with **5 navigation pages**:

1. **Intro** – Project overview
2. **Prediction** – Upload Image or Paste the image url -> Garbage Classification Prediction
3. **About Me** – Personal introduction





RecycleVision Navigation

Go to

- 💡 Intro
- 🔎 Prediction
- 👤 About Me

🔍 Garbage Image Prediction

📁 Upload an image...

Drag and drop file here
Limit 200MB per file • JPG, JPEG, PNG

Browse files

clothes_18.jpg 21.7KB

🗒️ Or paste an image URL here:

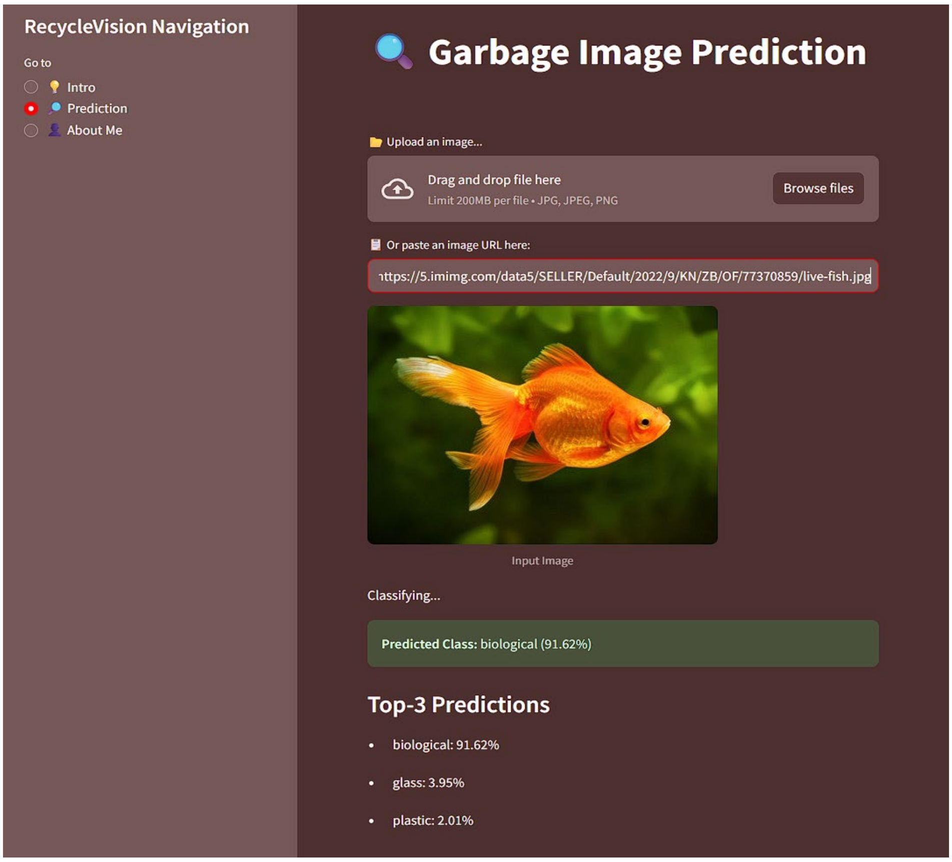
Input Image

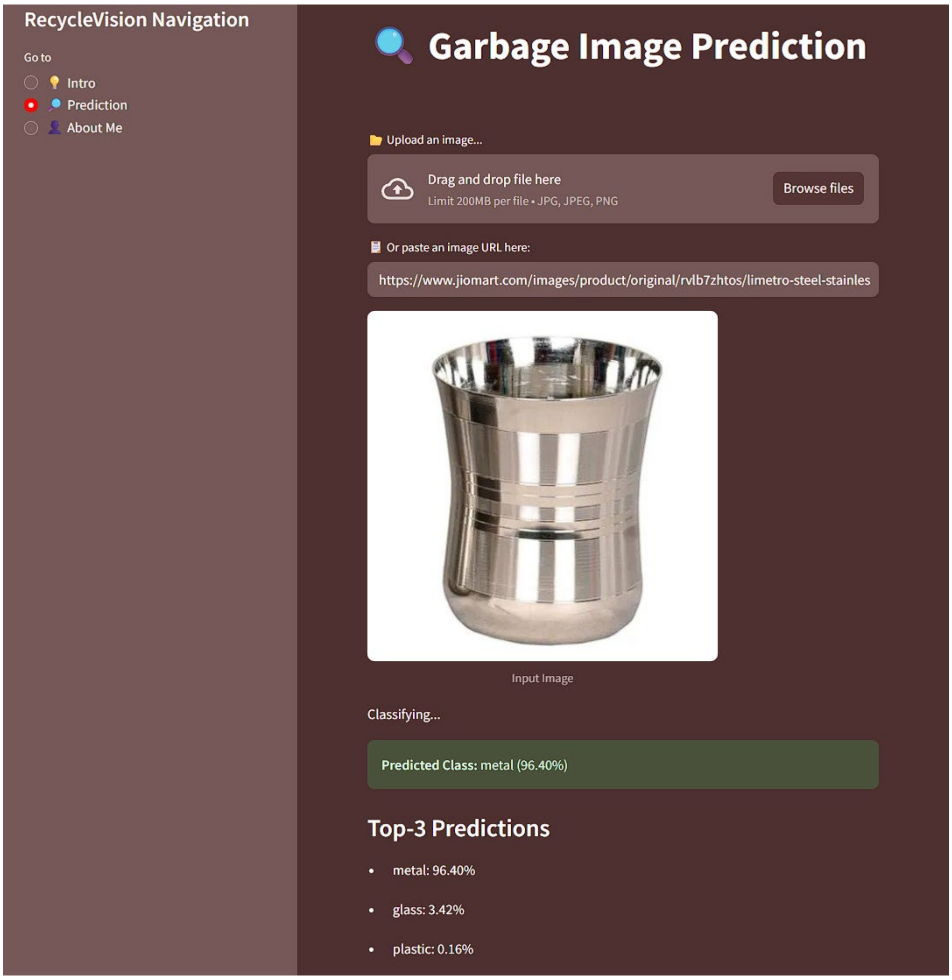
Classifying...

Predicted Class: clothes (100.00%)

Top-3 Predictions

- clothes: 100.00%
- paper: 0.00%
- plastic: 0.00%


RecycleVision Navigation
Go to
Intro
Prediction
About Me
Garbage Image Prediction
Upload an image...
Drag and drop file here
Limit 200MB per file • JPG, JPEG, PNG
Browse files
Or paste an image URL here:
ıttps://5.imimg.com/data5/SELLER/Default/2022/9/KN/ZB/OF/77370859/live-fish.jpg
Input Image
Classifying...
Predicted Class: biological (91.62%)
Top-3 Predictions
biological: 91.62%
glass: 3.95%
plastic: 2.01%


RecycleVision Navigation
Go to
Intro
Prediction
About Me
Garbage Image Prediction
Upload an image...
Drag and drop file here
Limit 200MB per file • JPG, JPEG, PNG
Browse files
Or paste an image URL here:
https://www.jiomart.com/images/product/original/rvlb7zhtos/limetro-steel-stainles
Input Image
Classifying...
Predicted Class: metal (96.40%)
Top-3 Predictions
metal: 96.40%
glass: 3.42%
plastic: 0.16%