Material Stream Identification System (MSI)

1. Introduction

Efficient and automated post-consumer waste sorting is a key enabler for achieving circular economy goals. Manual sorting is slow, error-prone, and expensive. This project presents an Automated Material Stream Identification (MSI) System based on computer vision and classical machine learning techniques. The system classifies waste items into predefined material categories and is capable of rejecting unknown or out-of-distribution objects.

The project emphasizes mastering the full machine learning pipeline, including data preprocessing, data augmentation, feature extraction, classifier training, evaluation, and real-time deployment.

2. Dataset Description

The provided dataset contains images organized into 6+1 class-specific folders. The defined classes are:

ID	CLASS	DESC
0	Glass	Bottles and jars made of amorphous solid materials
1	Paper	Thin cellulose-based materials
2	Cardboard	Multi-layer structured cellulose material
3	Plastic	High-molecular-weight organic compounds
4	Metal	Metallic substances such as cans
5	Trash	Non-recyclable or contaminated waste
6	Unknown	Out-of-distribution or unrecognizable inputs

3. Data Preprocessing and Augmentation

3.1 Preprocessing

- Verified all images in the raw dataset (6 classes: cardboard, glass, metal, paper, plastic, trash).
- Removed incorrect or corrupted images (e.g., files with zero pixels or unreadable format).
- Ensured all images are valid and in RGB mode for consistency.
- Checked for duplicate filenames and resolved conflicts.
- The dataset was split into training (70%), validation(15%), and test(15%) sets following best practices to ensure fair evaluation

- sized to a fixed resolution compatible with CNN-based feature extractors
- Normalized using ImageNet mean and standard deviation

3.2 Data Augmentation

To improve generalization and reduce overfitting, controlled data augmentation was applied to the training set only. Each class was augmented until reaching 550 images per class (more than 30%), ensuring class balance.
Same Image Size**:** (384, 512) pixels (width × height)
For all classes (what would be scaled later due FE step with EfficientNetB0)
Same format: .jpg
Randomization**:** Controlled with a fixed seed (SEED = 42) for reproducibility.

The augmentation pipeline included:

1. Random rotation (±10 degrees)
2. Horizontal flipping
3. Brightness adjustment (±10%)
4. Brightness and contrast adjustment (±10%)
5. Color jitter and gamma correction (±10%)
6. Gaussian blur (radius 0.5–1.0)

These transformations simulate variations in lighting, orientation, scale, and camera noise while preserving the semantic meaning of the objects.

4. Feature Extraction

Convert raw images into fixed-length feature vectors that can be fed to classical classifiers (SVM, k-NN). We evaluated three descriptor families to represent the six material classes: HOG (hand-crafted), ResNet features, and EfficientNet-B0 features (deep, pre-trained). In the final system we adopted EfficientNet-B0 because it achieved the best trade-off between accuracy, feature size, and speed.

ResNet50 vs EfficientNet-B0
EfficientNet-B0 achieves similar or slightly better accuracy than ResNet-50 with far fewer parameters and lower computational cost
For feature extraction:

- ResNet-50 → 2048-D vectors (higher dimensionality, more memory).
- EfficientNet-B0 → 1280-D vectors (lighter, faster for SVM/k-NN).

EfficientNet uses compound scaling (depth, width, resolution) for better efficiency

The final classification layer was removed, and the network output was used as a fixed-length feature vector.

- Output feature dimension: 1280
- Model pretrained on ImageNet
- Used in inference-only mode (no fine-tuning)

Dimensionality Reduction

To reduce feature dimensionality and improve classical classifier performance, Principal Component Analysis (PCA**)** was applied:

- Reduced features from 1280 → 256 dimensions
- PCA was fitted on training data only and applied to validation and test sets

5. Classification Models

5.1 A Support Vector Machine with an RBF kernel was trained using the PCA-reduced features (pca 256 components to reduce noise and speed up training)

Key parameters

- Kernel: RBF
- C = 10
- Gamma = scale
- Class weighting: balanced

Validation Accuracy: ≈ 94.55%

SVM was selected due to its strong performance in high-dimensional feature spaces and robustness to small dataset sizes.

5.2 KNN

A k-NN classifier was also evaluated for comparison. It classifies samples based on proximity in feature space.

Key parameters:

- Distance metric: cosine
- Weights: distance
- n_n_neighbors:5
- Training: On PCA-reduced features
- Validation Accuracy: 92.52%

SVM outperformed k-NN by ~2% in validation accuracy.
SVM is chosen as the final model for deployment due to better accuracy and robustness.

6. Handling the Unknown Class (Open-Set Recognition)

The dataset does not contain explicit training samples for the Unknown class. Instead, an open-set recognition strategy was implemented using confidence-based rejection.

6.1 SVM Unknown Detection

For SVM, class probabilities were obtained using predict_proba. A confidence threshold was defined:

- If the maximum predicted probability < threshold → classify as Unknown (ID = 6)
- Otherwise → assign the predicted known class

The threshold was empirically selected using the validation set to balance accuracy and rejection rate.

6.2 6.2 KNN Unknown Detection

For KNN, rejection was based on distance in feature space:

- Mean distance to k nearest neighbors was computed
- If the distance exceeded a predefined threshold → classify as Unknown

This approach assumes that unknown objects lie far from known-class clusters.

7. Performance Evaluation

7.1 The models were evaluated using:

- Accuracy
- Precision, Recall, and F1-score
- Confusion Matrix

SVM

	precision	recall	f1-score	support
Glass	0.93	0.90	0.91	48
Paper	0.95	0.96	0.95	54
Cardboard	0.97	0.90	0.93	39
Plastic	0.90	0.91	0.91	47
Metal	0.95	0.95	0.95	42
Trash	0.97	0.97	0.97	40
Unknown	0.00	0.00	0.00	0
accuracy			0.93	270
macro avg	0.81	0.80	0.80	270
weighted avg	0.94	0.93	0.94	270

KNN

	precision	recall	f1-score	support
0	0.98	0.85	0.91	48
1	0.91	0.94	0.93	54
2	0.97	0.90	0.93	39
3	0.88	0.98	0.93	47
4	0.93	0.98	0.95	42
5	0.97	0.97	0.97	40
accuracy			0.94	270
macro avg	0.94	0.94	0.94	270
weighted avg	0.94	0.94	0.94	270

8. System Deployment

The best-performing model (EfficientNet-B0 + PCA + SVM) was integrated into a real-time application.

Deployment features:

- Live camera input using OpenCV
- Real-time feature extraction
- Confidence-based Unknown rejection
- On-screen display of predicted class

The system achieved real-time performance and stable predictions under varying lighting conditions.

9. Conclusion

This project successfully demonstrates an end-to-end Automated Material Stream Identification system using classical machine learning combined with deep feature extraction. The use of EfficientNet-based features, PCA dimensionality reduction, and SVM classification resulted in high accuracy and robust open-set behavior.
The implemented rejection mechanism allows the system to safely handle unknown or out-of-distribution inputs, making it suitable for real-world waste sorting applications.
Future work may include fine-tuning feature extractors, exploring one-class classifiers, or integrating lightweight models for embedded deployment.