Faculty of Computing and Artificial Intelligence, Cairo University

Machine Learning Course – Fall 2025

Project: Material Stream Identification System (MSI)

GitHub repo link: https://github.com/abdomohsen228/material-stream-identification-system

TA: Mohammed Atta

Name	ID	Group
Duaa Abd-Elati Abd-Elazeem	20230609	4IS3
Abdelrahman Mohsen Farid	20221093	4IS3
Mariam Matrawy Ahmed Ali	20220334	4IS2
Ziad Mohammed Abdullah	20221067	4IS3
Tasneem Gomaa Anter	20220089	4IS1

Abstract

Automated waste sorting is a critical requirement for modern recycling pipelines and circular economic initiatives. This project presents a **Material Stream Identification (MSI) System** based on classical machine learning classifiers operating on fixed-length image feature vectors. The system follows a complete machine learning pipeline, including data augmentation, feature extraction, classifier training, performance evaluation, rejection-based unknown handling, and real-time deployment using a live camera feed. Two foundational classifiers—Support Vector Machines (SVM) and k-Nearest Neighbors (k-NN)—are implemented and compared under identical feature representations.

---

1. Introduction

Manual waste sorting is inefficient, costly, and prone to error. Automated vision-based systems offer a scalable solution by identifying material types directly from images. This project aims to design and implement an end-to-end MSI system using **fundamental machine learning techniques**, emphasizing feature engineering, classical classifiers, and system robustness.

The system classifies waste items into seven categories: six primary recyclable/non-recyclable material classes and one mandatory *Unknown* class for out-of-distribution inputs.

---

2. Material Classes

The system classifies images into the following predefined categories:

ID	Class Name	Description
0	Glass	Bottles, jars, and other glass containers
1	Paper	Newspapers, office paper
2	Cardboard	Corrugated and layered cellulose materials
3	Plastic	Bottles, packaging films
4	Metal	Aluminum cans, metallic waste
5	Trash	Non-recyclable or contaminated waste
6	Unknown	Out-of-distribution or uncertain inputs

3. Dataset and Preprocessing

The dataset consists of labeled RGB images organized into class-specific directories. Each directory represents one material category. Initial inspection revealed:

- Significant class imbalance
- Limited sample size
- High intra-class variation

These challenges motivated the use of **data augmentation** and **class balancing** strategies

---

4. Data Augmentation

4.1 Motivation

Data augmentation is a mandatory step to improve model generalization and reduce overfitting. The original dataset size is insufficient to cover variations in lighting, orientation, and scale.

4.2 Applied Techniques

The dataset size was increased by **40%**, exceeding the minimum required 30%. The following augmentation techniques were applied:

- Random rotation (±40 degrees)
- Horizontal and vertical flipping
- Scaling (0.85× to 1.15×)
- Brightness and contrast adjustment

These transformations preserve semantic meaning while increasing visual diversity.

4.3 Class Balancing

Augmentation was applied proportionally per class to ensure balanced representation across material categories.

---

5. Feature Extraction

5.1 Image-to-Vector Conversion

Machine learning classifiers require fixed-length numerical input. Each RGB image is converted into a **2048-dimensional feature vector** using a pretrained ResNet50 convolutional neural network.

5.2 Justification of Feature Extractor

A pretrained ResNet50 model (ImageNet weights) is used **strictly as a fixed feature extractor**:

- The final classification layer is removed.
- No fine-tuning or backpropagation is performed.
- The network operates in inference-only mode.

This approach treats CNN as a high-level descriptor generator, analogous to handcrafted feature extractors, while all learning and decision-making are performed by classical machine learning algorithms.

5.3 Preprocessing Pipeline

- Image resizing to 224×224 pixels
- Normalization using ImageNet statistics
- Forward pass through ResNet50 backbone
- Output feature vector flattened to length 2048

The same preprocessing pipeline is used during **training and real-time inference**, ensuring full consistency.

---

6. Classification Models

6.1 Support Vector Machine (SVM)

An SVM classifier with an **RBF kernel** is trained on the extracted features.

The complete pipeline includes:

- Feature standardization
- Principal Component Analysis (PCA)
- SVM classification with probability estimation
- Class-weighted training to address imbalance

Hyperparameters were optimized using stratified cross-validation. Fixed parameters were selected to ensure training stability.

6.2 k-Nearest Neighbors (k-NN)

A k-NN classifier is implemented using:

- Feature standardization
- PCA dimensionality reduction
- Distance-based and uniform weighting schemes

Hyperparameters are optimized using grid search with stratified cross-validation.

7. Handling the Unknown Class

To prevent unreliable predictions, a **rejection mechanism** is implemented. If the maximum predicted class probability is below a predefined confidence threshold, the prediction is rejected and labeled as **Unknown**.

This approach improves system robustness and reflects real-world deployment requirements where unseen or ambiguous inputs are common.

---

8. Model Evaluation

Performance is evaluated using:

- Classification accuracy
- Precision, recall, and F1-score
- Stratified train/test splits

Evaluation focuses primarily on the six primary classes, while the Unknown class is handled through rejection logic rather than direct training targets.

---

9. Real-Time System Deployment

The best-performing model is integrated into a real-time application using OpenCV. The system:

- Captures frames from a live camera
- Extracts feature in real time
- Applies classification and rejection logic
- Displays predicted class and confidence score

To maintain responsiveness, inference is performed on every Nth frame.

---

10. Comparison Between SVM and k-NN

Criterion	SVM	k-NN
Training Time	Moderate	Low
Inference Speed	Fast	Slower for large datasets
Memory Usage	Moderate	High
Robustness	High	Moderate
Best Accuracy	Higher	Slightly lower

SVM demonstrated superior generalization and was selected for deployment.

---

11. Model Selection Decision

Although both classifiers achieved reasonable performance, **SVM consistently outperformed k-NN in terms of accuracy, robustness, and inference efficiency**. SVM demonstrated better generalization in high-dimensional feature space and provided more stable confidence estimates, which are critical for rejection-based unknown handling.

Given its superior performance and suitability for real-time deployment, **SVM was selected as the final deployed model**.

---

12. Conclusion

This project presents a complete and robust **Material Stream Identification System** based on classical machine learning techniques. By combining data augmentation, fixed CNN feature extraction, robust classification, rejection mechanisms, and real-time deployment, the system satisfies all project requirements.

Experimental results confirm that **Support Vector Machines provide the best trade-off between accuracy, robustness, and deployment efficiency**, making them the optimal choice for this application.