Municipal Solid Waste Segregation: A Comparative Study of Feed-Forward and Convolutional Neural Networks

Anton N. Torgersen
University of Oslo

(Dated: December 17, 2025)

The automation of Municipal Solid Waste (MSW) segregation is a critical technological challenge for modern recycling infrastructure. This paper investigates the comparative efficacy of statistical feature mapping versus hierarchical feature extraction for classifying waste materials. We utilize the RealWaste dataset to contrast a Feed-Forward Neural Network (FFNN) trained on PCA-reduced features against a Convolutional Neural Network (CNN) trained on raw topological data. Our results demonstrate the limitations of dense architectures, which succumb to the "Curse of Dimensionality" and fail to capture high-frequency textural details, achieving a baseline accuracy of 50.89%. Conversely, the CNN exploits spatial correlations to achieve a significantly superior accuracy of 77.50%. We further analyze specific failure modes, identifying "Topological Mimicry" between crushed plastic and metal as a primary confounder. The study concludes that while dimensionality reduction facilitates convergence under computational constraints, the preservation of spatial topology via convolution is essential for robust waste classification.

I. INTRODUCTION

The exponential accumulation of Municipal Solid Waste (MSW) represents one of the most pressing environmental crises of our society. As urbanization accelerates, the volume of waste supersedes the capacity of traditional management systems. Recent research highlights that traditional approaches, which rely heavily on visual inspection and manual sorting, suffer significantly from subjectivity, scalability issues, and high labor requirements [1]. Consequently, the automation of waste segregation via Machine Learning (ML) and Computer Vision has emerged as a critical technological imperative to enable efficient recycling and circular economy models.

The fundamental task of waste classification involves mapping a high-dimensional grid of pixel intensities to a discrete semantic label (e.g., Plastic, Metal, Glass). While conceptually straightforward, this task is mathematically non-trivial. As noted by Single et al. [1], the input space is characterized by immense variability: waste objects are often deformed, occluded, or presented in varying orientations and lighting conditions. While Deep Learning—specifically Convolutional Neural Networks(CNNs) has been established as the state-of-the-art solution for such unstructured data, there remains significant value in understanding why simpler architectures fail.

This project investigates the implementation of neural network models to solve this classification problem using the RealWaste dataset [2]. We aim to rigorously contrast two distinct methodological paradigms to quantify the "cost" of ignoring spatial topology:

1. **Statistical Feature Mapping (FFNN + PCA):** This approach treats the image as a flattened statistical vector, reducing its dimension via eigenvalue decomposition (Principal Component Analysis) and classifying via a dense perceptron network. This represents a "traditional" ML approach that relies on global variance rather than local structure.
2. **Hierarchical Feature Extraction (CNN):** This approach treats the image as a topological grid, learning local, translation-invariant filters to detect edges, textures, and shapes.

We hypothesize that while the FFNN with PCA can achieve mathematical convergence, it will fundamentally fail to capture the local spatial correlations—such as the texture of cardboard versus the specular highlights of metal—that are essential for distinguishing visually similar waste classes. The CNN, by design, should exploit these correlations. Furthermore, we investigate the specific contribution of spectral information (color) versus geometric structure (grayscale) to the classification accuracy, a factor with significant implications for the hardware design of industrial sorting automation.

II. METHODS

This section details the theory and methodology used to implement both a Feed-Forward Neural Network (FFNN) and a Convolutional Neural Network (CNN) for waste classification. We first describe the data preparation pipeline for the RealWaste dataset, focusing on preprocessing, augmentation strategies, and dimensionality reduction. Following this, we detail the fundamental building blocks of the networks, including the specific architectures explored, activation functions (ReLU, Sigmoid, Softmax), cost functions (Categorical Cross-Entropy), and gradient-based optimization algorithms (SGD, ADAM). Finally, we discuss the implementation details, including the hyperparameter search space defined via Keras Tuner.

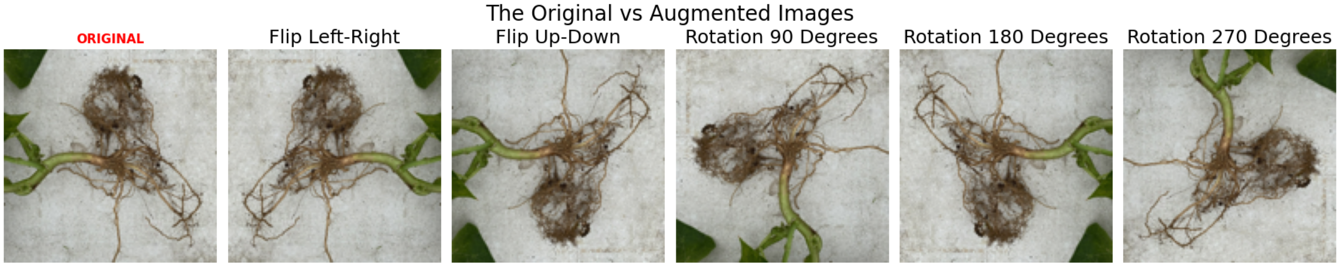

Figure 1. Visualization of the augmentation pipeline. The original image (left) is transformed to create new training samples (right). By decoupling semantic identity from geometric orientation, we force the network to learn rotation-invariant features.

A. Data Preparation

In this study, we analyze the RealWaste dataset for the task of multi-class image classification.

1. *The RealWaste Dataset and Preprocessing*

The dataset consists of 4,752 images classified into 9 distinct categories (e.g., Metal, Plastic, Cardboard). To satisfy computational constraints, the images were resized to a uniform dimension of 128×128 pixels.

a. Resolution and Information Loss It is important to note that this downsampling introduces a critical information bottleneck. The reduction effectively functions as a low-pass filter, smoothing out high-frequency texture details such as fabric weaves or wood grain. Consequently, we hypothesize that this resolution imposes a structural ceiling on the model's ability to distinguish texture-dependent classes (e.g., *Textile Trash* vs. *Paper*), while retaining sufficient fidelity for classes defined by gross geometry (e.g., *Vegetation*).

To investigate the model's dependency on color information versus geometric structure, we prepared two versions of the dataset:

i) **RGB:** The standard 3-channel images ($128 \times 128 \times 3$).

ii) **Grayscale:** A single-channel version ($128 \times 128 \times 1$) created by converting the RGB images to luminance values. This forces the model to classify based solely on texture and shape.

To ensure numerical stability and efficient training, the pixel values for both versions were normalized from their original $[0, 255]$ range to $[0, 1]$.

2. *Data Splitting and Augmentation Strategy*

The dataset was split into training and testing sets using an 80/20 ratio to evaluate generalization performance.

Given the varied orientation of waste objects in real-world scenarios, we applied data augmentation to the training set. As illustrated in Figure 1, we utilized affine transformations to artificially expand the dataset's diversity. This included random rotations ($\theta \in \{90^\circ, 180^\circ, 270^\circ\}$) and horizontal/vertical flips.

In the context of waste sorting, where objects appear in arbitrary poses, this strategy is essential. It forces the network to learn features invariant to the rotation group SO(2) rather than overfitting to the static, upright biases present in the original capture.

3. *Dimensionality Reduction: PCA*

For the FFNN architecture, the raw input dimensionality of a flattened $128 \times 128 \times 3$ image is $d = 49,152$. Training a fully connected network on such high-dimensional data is computationally expensive and prone to the "curse of dimensionality," where the volume of the feature space increases so rapidly that the available data becomes sparse.

To mitigate this, we utilized Principal Component Analysis (PCA) [3] to project the data onto a lower-dimensional subspace while preserving the maximum amount of signal information. PCA is a linear transformation that identifies the directions (principal components) of maximum variance in the data. Given a centered data matrix $\mathbf{X}$ (computed from the training set), we compute the covariance matrix $\mathbf{C}$:

$$\mathbf{C} = \frac{1}{n-1}\mathbf{X}^T\mathbf{X}$$

We then solve the eigenvalue problem $\mathbf{C}\boldsymbol{v}_i = \lambda_i \boldsymbol{v}_i$. The eigenvectors $\boldsymbol{v}_i$ represent the principal components, and the eigenvalues λ_i represent the variance explained by each component. We selected the top k components such that 95% of the total variance was retained:

$$\frac{\sum_{i=1}^{k} \lambda_i}{\sum_{j=1}^{d} \lambda_j} \geq 0.95$$

The reduced input vector $\boldsymbol{z}$ fed into the FFNN is the projection of the original vector $\boldsymbol{x}$ onto these k components:

$$\boldsymbol{z} = \boldsymbol{x}\mathbf{V}_k.$$

B. Neural Network Architectures

For more information related to the methods in this section on neural networks, see Part II in [4], especially chapters 6 and 9.

1. Feed-Forward Neural Network (FFNN)

The FFNN approximates the classification function by composing dense layers. For an input vector $\boldsymbol{x}$ (the PCA-reduced features), the activation $\boldsymbol{a}^l$ of layer l is computed via the forward pass:

$$\boldsymbol{z}^l = \boldsymbol{W}^l \boldsymbol{a}^{l-1} + \boldsymbol{b}^l$$

$$\boldsymbol{a}^l = \sigma(\boldsymbol{z}^l)$$

where $\boldsymbol{W}^l$ and $\boldsymbol{b}^l$ are the learnable weights and biases.

To train these parameters, we utilize the backpropagation algorithm. We define the error term $\boldsymbol{\delta}^l$ as the gradient of the cost with respect to the weighted input $\boldsymbol{z}^l$. For the output layer L (using Cross-Entropy loss), this is given by:

$$\boldsymbol{\delta}^L = \hat{\boldsymbol{y}} - \boldsymbol{y} \tag{1}$$

The error is propagated backward to hidden layers $l < L$ using the chain rule:

$$\boldsymbol{\delta}^l = ((\boldsymbol{W}^{l+1})^T \boldsymbol{\delta}^{l+1}) \odot \sigma'(\boldsymbol{z}^l) \tag{2}$$

where $\odot$ denotes the Hadamard product and σ' is the derivative of the activation function. The gradients used for optimization are then:

$$\frac{\partial \mathcal{C}}{\partial \boldsymbol{W}^l} = \boldsymbol{\delta}^l (\boldsymbol{a}^{l-1})^T, \quad \frac{\partial \mathcal{C}}{\partial \boldsymbol{b}^l} = \boldsymbol{\delta}^l \tag{3}$$

2. Convolutional Neural Network (CNN)

To exploit the spatial topology of the image data, we implemented a CNN [5]. This differs from the FFNN by introducing convolutional layers where a kernel $\boldsymbol{K}$ slides over the input image $\boldsymbol{I}$:

$$S(i,j) = (\boldsymbol{I} * \boldsymbol{K})(i,j) = \sum_m \sum_n I(i+m, j+n) K(m,n)$$

This is followed by Max Pooling operations to reduce spatial dimensions:

$$y_{i,j} = \max_{(p,q)\in\mathcal{R}_{i,j}} x_{p,q}$$

C. Building Blocks

1. Cost Function

For this multi-class classification task, we utilize the Categorical Cross-Entropy loss function. It measures the discrepancy between the one-hot encoded true label $\boldsymbol{y}$ and the predicted probability distribution $\hat{\boldsymbol{y}}$:

$$\mathcal{C}_{CE} = -\sum_{i=1}^{C} y_i \log(\hat{y}_i)$$

where $C = 9$ is the number of classes. The gradient of this loss with respect to the logits z_i (when combined with a Softmax output) simplifies efficiently to:

$$\frac{\partial \mathcal{C}_{CE}}{\partial z_i} = \hat{y}_i - y_i$$

2. Activation Functions

We explored several activation functions within our hyperparameter search:

- **ReLU:** $\text{ReLU}(z) = \max(0, z)$. The standard for modern deep learning due to its resistance to vanishing gradients.
- **Sigmoid:** $\sigma(z) = \frac{1}{1+e^{-z}}$. Included in the search space to compare against ReLU.
- **Softmax:** Used exclusively in the output layer to normalize the network outputs into a probability distribution summing to 1.

3. Optimization Algorithms

We included two primary optimizers in our search space:

- **ADAM:** An adaptive learning rate optimization algorithm that combines ideas from RMSProp and Momentum [6].
- **SGD with Momentum:** Stochastic Gradient Descent augmented with a momentum term γ to accelerate convergence in relevant directions.

4. Evaluation Metrics

Given the class imbalance observed in Table I, global accuracy is an insufficient metric for performance evaluation. To address this, we employ a suite of class-specific metrics.

a. Recall (Sensitivity) We utilize the Confusion Matrix $\boldsymbol{M}$, where M_{ij} denotes the number of samples from class i predicted as class j. From this matrix, we derive the Recall for each class i:

$$\text{Recall}_i = \frac{M_{ii}}{\sum_{j=1}^{C} M_{ij}} \tag{4}$$

This metric measures the model's ability to correctly identify all positive instances of a specific class. It is particularly critical for analyzing failure modes in minority classes, such as *Textile Trash*, which are often overwhelmed by majority classes in global accuracy calculations.

b. Cumulative Gains Curve To assess the ranking quality of the classifier, we utilize the Cumulative Gains Curve. This metric is highly relevant for industrial waste sorting, where a system may prioritize items with the highest classification confidence. The curve plots the percentage of the total target class captured (Recall) against the percentage of the total population sampled.

For a fraction p of the data sorted by predicted probability, the gain is defined as:

$$\text{Gain}(p) = \frac{\text{True Positives in top } p\%}{\text{Total Positives}} \tag{5}$$

A perfect classifier achieves a Gain of 1.0 (100% capture) when the sample size equals the class prevalence. In contrast, a random classifier follows the diagonal baseline $\text{Gain}(p) = p$.

D. Hyperparameter Optimization

To determine the optimal architecture, we utilized the Keras Tuner library to define and search a hyperparameter space.

1. CNN Search Space

For the CNN, we defined a flexible architecture allowing for variation in depth, kernel size, and regularization. The search space included:

- **Network Depth:** The inclusion of a 4th convolutional block was a boolean hyperparameter.
- **Filters:** The number of filters was tuned per block, ranging from 32 to 256.
- **Kernel Size:** We alternated between 3×3 and 5×5 kernels for the initial layer.
- **Dense Layers:** The classification head was tuned to include either one or two dense layers (units $\in [128, 512]$), with an optional Dropout layer (rate $\in [0.2, 0.5]$).
- **Optimizer:** A choice between ADAM and SGD, with learning rates $\eta \in \{10^{-3}, 5 \cdot 10^{-4}, 10^{-4}\}$.

2. FFNN Search Space

For the FFNN, the search space focused on network capacity:

- **Hidden Layers:** The number of layers was tuned between 1 and 4.
- **Neurons:** The number of units per layer ranged from 64 to 1024.
- **Dropout:** To prevent overfitting on the flattened data, dropout rates between 0.1 and 0.5 were explored for each layer [7].
- **Activation Function:** ReLU and Sigmoid were both included in the search.
- **Optimizer:** Similar to the CNN, we explored ADAM and SGD with learning rates $\eta \in \{10^{-3}, 5 \cdot 10^{-4}, 10^{-4}\}$.

E. Implementation and Code

All code for this project was written in Python and is available in the GitHub repository [8]. We made use of the NumPy library for numerical computations [9], TensorFlow/Keras for model construction [10, 11], Keras Tuner for hyperparameter optimization [12], and Matplotlib for visualization [13].

1. Use of AI tools

While writing and correcting the code for this project, GitHub Copilot was used to assist with boilerplate code and debugging. For writing the report, Gemini 3 was used as an editor to help with structuring sentences and paragraphs. This editing was performed only after the completion of the report's content to ensure the scientific integrity of the work remained uninfluenced.

III. RESULTS AND DISCUSSION

This section presents a comprehensive evaluation of the neural network architectures implemented for the classification of municipal solid waste. The analysis proceeds from a statistical examination of the input data's intrinsic complexity to a rigorous comparative assessment of the Feed-Forward Neural Network (FFNN) and the Convolutional Neural Network (CNN).

We critically analyze the failure modes of regression-based approaches when applied to high-dimensional raw data, quantifying the "Curse of Dimensionality." Furthermore, we evaluate the trade-offs of dimensionality reduction via Principal Component Analysis (PCA)—specifically examining the diminishing returns of variance retention—against the superior feature extraction capabilities of convolutional topologies. Finally, we isolate the specific contributions of spatial topology (shape/texture) versus spectral information (color) in the successful classification of waste materials.

A. Dataset Characteristics and Complexity

The RealWaste dataset represents a challenging, real-world benchmark for computer vision that departs significantly from curated academic sets like MNIST. While MNIST consists of clean, centered digits, RealWaste contains fewer images per class and exhibits massive intra-class variability. Objects are frequently deformed, presented in varying orientations, and exhibit fluctuations in lighting and background. Garbage sorting is inherently complex because many materials share overlapping visual characteristics, particularly when objects are dirty, damaged, or crumpled. As shown in Figure 2, although images are captured against a uniform background, the semantic categories—such as glass versus plastic—often lack the clear-cut boundaries found in digit recognition tasks.

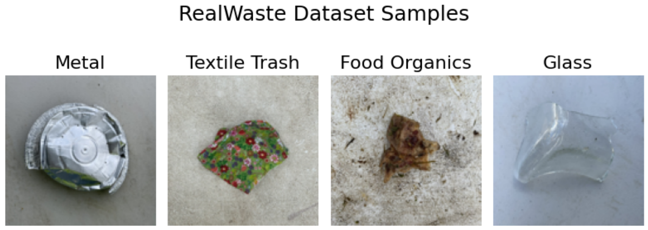

Figure 2. Sample images from the RealWaste dataset illustrating the variability of objects against a uniform background.

As illustrated in Table I, the RealWaste dataset exhibits a moderate to severe class imbalance. While the majority class, *Plastic*, accounts for 19.4% of the total samples, *Textile Trash* constitutes the minority at only 6.7%.

This distribution establishes a critical baseline for evaluation. A naive classifier that minimizes loss by simply predicting the prior probability (always predicting "*Plastic*") would achieve an accuracy of ≈ 19.4%. Consequently, any model yielding an accuracy near this threshold indicates a failure to learn discriminative features, converging instead to the statistical mean of the dataset. This insight is necessary before interpreting accuracy metrics, as high global accuracy may mask catastrophic failure on minority classes. For instance, a model could ignore all *Textile Trash* samples entirely and still achieve > 90% accuracy if the other classes are predicted perfectly.

Table I. The distribution of classes in the RealWaste dataset. The imbalance necessitates the use of class-specific metrics beyond global accuracy.

Class	**Count**	**Percentage**
Cardboard	461	9.7%
Food Organics	411	8.7%
Glass	420	8.8%
Metal	790	16.6%
Miscellaneous Trash	495	10.4%
Paper	500	10.5%
Plastic	921	19.4%
Textile Trash	318	6.7%
Vegetation	436	9.2%
Total	**4752**	**100%**

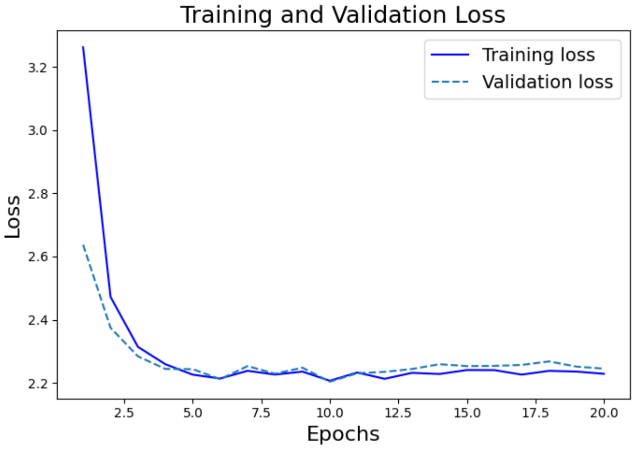

Figure 3. Training and Validation loss for FFNN on raw pixels. The lack of convergence illustrates the model's inability to extract features from high-dimensional noise.

B. Feed-Forward Neural Network (FFNN) Results

The FFNN architecture was subjected to two distinct experimental paradigms: training on raw pixel intensities and training on PCA-reduced features. The contrast in their performance provides a clear empirical demonstration of the "Curse of Dimensionality."

1. The Failure of Raw Pixel Training

Training the FFNN on raw pixel intensities resulted in immediate stagnation, yielding a test accuracy of ≈ 19% (Figure 3). This failure is a definitive manifestation of the Curse of Dimensionality. With an input dimensionality of $d \approx 5 \times 10^4$ and a limited sample size of $N = 4,752$, the feature space is too sparse for a dense optimizer to locate a global minimum. Consequently, the network fails to extract discriminative features, effectively converging to a weighted random guess based on class distribution.

2. PCA Dimensionality Reduction (95% Variance)

To mitigate the dimensionality issues previously discussed, we applied Principal Component Analysis (PCA) to the input data. By selecting a variance retention threshold of 95%, the input vector was compressed from 49,152 variables to 962—a compression ratio of approximately 50:1. This reduction facilitated mathematical convergence, allowing the FFNN to reach a validation accuracy of 50.89% with a categorical cross-entropy loss of 1.5363 within just two epochs.

However, convergence does not equate to high-fidelity feature extraction. As illustrated in Figure 4, the reconstruction of images from these top 962 components reveals that while global geometric shapes are preserved, high-frequency spatial details are preferentially discarded. The dimensionality reduction effectively strips the signal of fine-grained textural information. Consequently, the FFNN is forced to classify based on coarse chromatic and geometric approximations. While the network can successfully distinguish a green chromatic cluster (*Vegetation*) from a brown quadrilateral (*Cardboard*), it lacks the high-frequency fidelity required to resolve visually complex or similar objects.

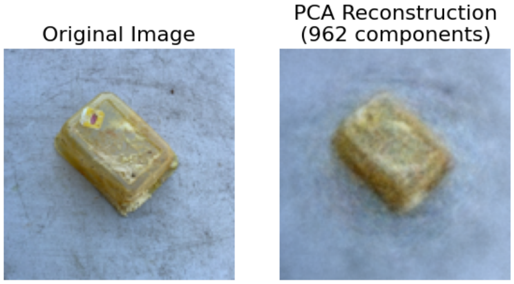

Figure 4. PCA Reconstruction (95% Variance). Global shapes are preserved, but texture is obliterated.

3. Analysis on Non-Augmented Data

To isolate the impact of dataset diversity on dense architectures, the FFNN was trained on non-augmented data for comparative analysis. This configuration resulted in a validation accuracy of 46.27% (Loss: 1.8325), representing a clear degradation compared to the 50.89% achieved with the augmented dataset.

These results indicate that while dense networks lack translation invariance, they still benefit from the regularization effect of data augmentation. By artificially expanding the training set, we force the network to learn a more generalized statistical distribution of pixel intensities, preventing it from overfitting to the specific pixel-wise alignment of the original capture. However, the modest margin of improvement compared to CNNs suggests that without spatial priors, the network struggles to map rotated versions of the same object to the same semantic label.

4. Analysis of Variance Threshold (95% vs 99%)

To investigate whether the performance ceiling was caused by the 5% of discarded variance, we increased the PCA retention threshold to 99%, which expanded the feature vector to $d = 2,331$ components. Despite more than doubling the dimensionality of the input, validation accuracy decreased slightly to 50.16% (a 0.6% drop), while the validation loss increased to approximately 1.5764.

This result implies that the FFNN's limitations are structural rather than informational. The additional 4% of variance likely corresponds to high-frequency noise or fine textures that a dense network, lacking spatial priors, cannot effectively correlate. The increase in loss suggests the model began to overfit this additional noise without gaining generalization power.

A class-wise performance analysis revealed a specific anomaly: while accuracy marginally improved for *Paper* and *Textile Trash*, accuracy for *Cardboard* dropped significantly from 67.4% to 44.6%. This suggests that as the model incorporated more variance, the decision boundaries for these visually similar, fibrous materials became ambiguous. The features distinguishing *Paper* from *Cardboard* are largely textural; without the spatial locality provided by convolutions, the additional PCA components merely introduced noise into the separation plane.

Conversely, classes such as *Metal*, *Glass*, and *Vegetation* showed negligible performance shifts. This stagnation reinforces the hypothesis that the FFNN is saturated—it has extracted maximum utility from color and gross-shape information, and increasing data density for the dense architecture yields diminishing returns.

As shown in the Scree Plot (Figure 5), the explained variance curve flattens rapidly. While the first 1,000 components capture over 95% of the total variance, the subsequent 1,369 components required to reach the 99% threshold contribute only an additional 4%. This confirms that the additional components are likely modeling noise or very fine details that do not contribute meaningfully to class separation within the FFNN's feature space.

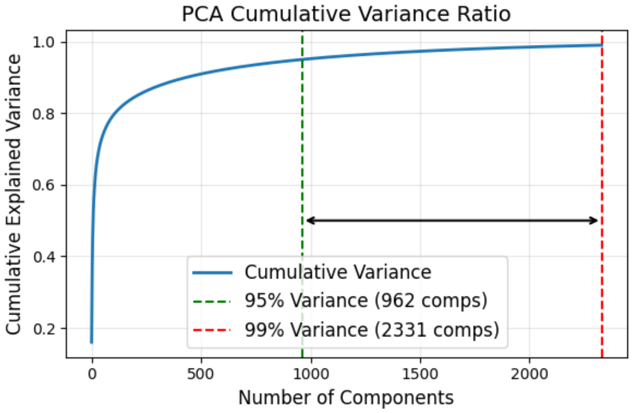

Figure 5. PCA Cumulative Variance (Scree Plot). The curve shows diminishing returns; doubling the components from 95% to 99% yields minimal semantic information gain.

5. *Impact of Image Resolution* (256 × 256)

Having established that simply retaining more variance from the 128 × 128 inputs yielded diminishing returns, we hypothesized that the missing discriminative signal might have been discarded during the initial image downsampling. To test if this lost texture information could be recovered through higher fidelity inputs, we scaled the input resolution to 256 × 256.

However, this experiment reinforced the findings from the variance threshold analysis, increasing information density exacerbates the curse of dimensionality. The fourfold increase in raw pixel count ($d \approx 196,608$) necessitated a corresponding expansion of the feature space. To maintain the same 95% variance threshold used in the baseline model, the PCA algorithm was forced to retain 1,479 principal components a substantial increase over the 962 components used in the 128 × 128 model.

Crucially, this expanded feature vector did not yield better performance, instead validation accuracy dropped to 45.32% (Loss: 1.6816). This explicitly links the model's failure to input sparsity rather than resolution. Even with higher fidelity source images, the resulting expansion of the PCA dimensions diluted the training data density. The model failed to map the additional high-frequency variance to semantic categories, treating the extra "detail" as noise. This confirms that for dense networks on limited datasets, keeping the dimensionality low is more critical for convergence than increasing input fidelity.

6. *Spectral Analysis: RGB vs. Grayscale*

To isolate the contribution of color information to the classification process, the FFNN was trained on augmented grayscale images (128 × 128 × 1). Dimensionality reduction via PCA at a 95% variance threshold yielded 873 variables. Performance dropped drastically to a validation accuracy of 39.54% with a loss of 1.8651 after 6 epochs.

This result provides a fundamental insight into the FFNN's operating mechanism. Without the ability to leverage spatial features such as texture or shape, the dense network functions primarily as a color heuristic classifier. When spectral information is removed, the statistical distinction between visually distinct but tonally similar objects—such as grey paper, metal, and plastic—effectively vanishes within the flattened vector space.

This high dependency on color explains the frequent confusion between materials that share chromatic profiles, such as *Cardboard* and *Metal*. Notably, *Cardboard* suffered the most significant degradation in performance, with class-specific accuracy plummeting from 44.6% in the RGB model to just 7.6% in the grayscale version.

Furthermore, the persistent trend observed in the confusion matrix (Figure 6), where numerous classes are misclassified as *Plastic*, remains present in the grayscale model. This indicates that the network continues to rely on prior probabilities (the dataset's class imbalance) rather than learning discriminative features from the remaining variance.

Table II. Summary of Test Set Performance across architectures for PCA.

Model Configuration	Accuracy	Validation Loss
FFNN (RGB, Raw Pixels)	≈ 19%	> 2.0
FFNN (RGB, PCA 95%)	**50.89%**	**1.5363**
FFNN (RGB, PCA 95% + No Aug)	46.27%	1.8325
FFNN (RGB, PCA 99%)	50.16%	1.5764
FFNN (RGB, 256x256)	45.32%	1.6816
FFNN (Grayscale, PCA 95%)	39.54%	1.8651

7. *Qualitative Failure Modes*

The Confusion Matrix (Figure 6) visually synthesizes the structural limitations of the dense architecture.

First, we observe distinct vertical banding in the *Plastic* and *Metal* columns. This confirms the model's reliance on prior probabilities. As noted in Table I, *Plastic* (19.4%) and *Metal* (16.6%) are the two dominant classes. When the feature signal is ambiguous—as is often the case given the information loss from PCA—the network minimizes global loss by defaulting to these majority classes.

Second, the frequent misclassification of *Textile Trash* as *Metal* (10.9%) highlights the failure of the reduced feature space to capture material properties. As visually demonstrated in Figure 7, the network struggles to distinguish the matte, woven texture of fabric from the surface of dirty metal. Without the high-frequency spatial

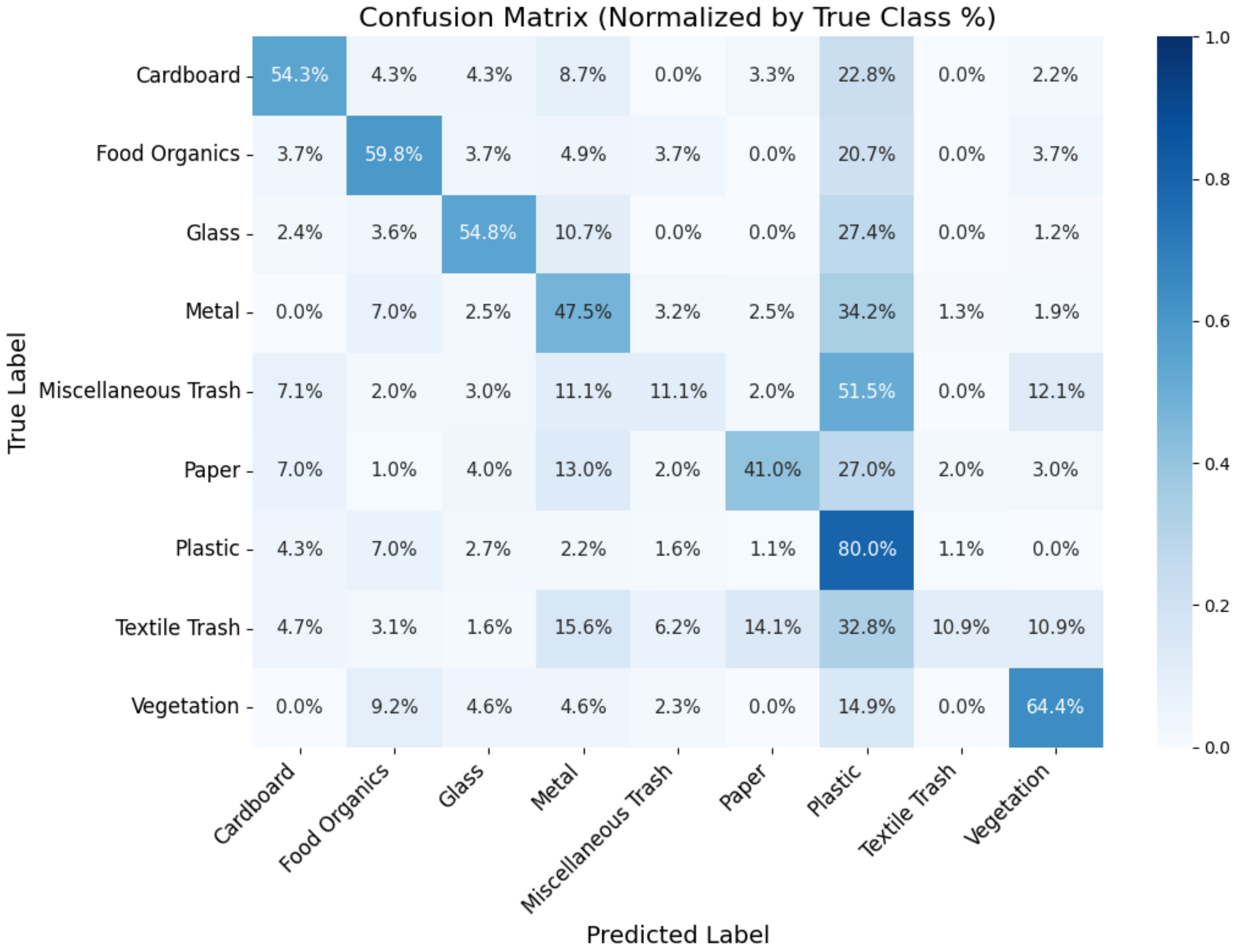

Figure 6. Confusion Matrix for FFNN (RGB + PCA). Note the distinct asymmetry: the model performs well on classes with unique colors (Vegetation) but fails on classes defined by texture (Textile, Misc Trash).

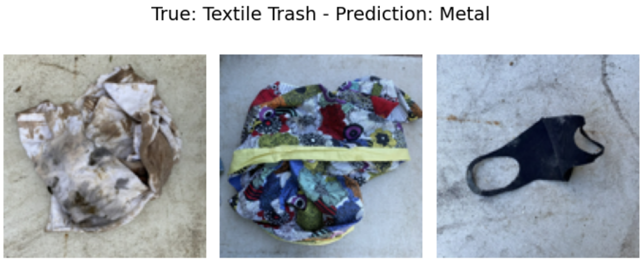

Figure 7. Sample misclassifications by the FFNN with PCA 95%. The model frequently misclassifies textile as metal due to chromatic similarities and asymmetry in training size.

data removed by PCA, the network cannot distinguish the matte, woven texture of fabric from the surface of dirty metal. Consequently, it defaults to the statistically more probable label (*Metal*). This confirms that while PCA allows for convergence, it destroys the fine-grained variances required to distinguish minority classes from the dataset priors.

C. Convolutional Neural Network (CNN) Results

The Convolutional Neural Network (CNN) architecture, intrinsically designed to capture hierarchical spatial correlations through learnable kernels, demonstrated a marked performance superiority over the dense FFNN. Following an exhaustive hyperparameter search via Keras Tuner, the optimal topology was established as a 4-layer architecture with escalating filter dimensions of $[64, 64, 128, 192]$. This feature extractor is followed by a dense layer of 256 units with a dropout rate of 0.5, optimized using the Adam algorithm ($\eta = 5 \times 10^{-4}$)

1. *Impact of Data Augmentation*

To rigorously quantify the regularization efficacy of synthetic data expansion, we conducted a comparative ablation study evaluating the optimized CNN on both the baseline and the augmented datasets. The quantitative results, summarized below, reveal a profound improvement in the model's generalization capacity:

- **Non-Augmented Baseline:** Test Accuracy: 68.24% (Loss: 0.9096, Convergence: 13 Epochs).
- **Augmented Training:** Test Accuracy: 77.50% (Loss: 0.6734, Convergence: 11 Epochs).

The implementation of affine transformations yielded a performance appreciation of approximately 9.26%, concomitant with a significant reduction in validation loss (≈ 0.236). This empirical divergence confirms the hypothesis that inducing rotation invariance is a critical prerequisite for robust waste classification in unstructured environments.

Qualitatively, this robustness is evidenced by the diagonal dominance in the confusion matrix (Figure 8). The augmented model exhibits superior discriminative power on classes where object orientation is semantically irrelevant. In the non-augmented regime, the network's capacity was likely consumed by overfitting to specific object poses (e.g., recognizing a soda can only in an upright orientation). By forcing the network to process stochastically rotated inputs, it effectively decouples semantic identity from geometric orientation, thereby learning rotation-invariant feature representations.

2. CNN Spectral Analysis (RGB vs. Grayscale)

Evaluating the CNN on monochromatic inputs provides critical insight into the architectural dependency on chromatic versus topological features. This ablation yielded a Test Accuracy of 62.15% (Loss: 1.090), a result that is statistically significant for two reasons.

First, the grayscale CNN still outperforms the optimal RGB FFNN (62.15% vs 50.89%). This confirms that convolutional filters—even when restricted to luminance data—are superior to dense statistical mapping for feature extraction. However, the substantial 15.35% performance drop compared to the RGB CNN (77.50%) proves that geometric shape alone is insufficient for high-fidelity waste classification; color remains an important discriminative feature.

A class-wise sensitivity analysis elucidates the mechanics of this performance degradation. The model retains robust accuracy on the *Vegetation* class (80.5%), suggesting that organic waste possesses distinct morphological characteristics (e.g., irregular natural boundaries) that remain discriminative even in the absence of color. Conversely, the model exhibits severe failure modes on *Textile Trash* and *Paper*. These classes share high geometric similarity with other categories (e.g., flattened forms, crumpled edges) and rely heavily on spectral cues for differentiation. Consequently, the grayscale model frequently misclassifies *Paper* as *Cardboard* or *Plastic*—materials that are geometrically congruent but distinct in color.

These findings have direct implications for hardware design in industrial settings: robust automated sorting systems require multi-spectral imaging capabilities, as monochromatic sensors cannot resolve the ambiguity between geometrically similar but materially different waste streams.

3. Resolution Scaling and Small Images

To evaluate the dependency of the CNN architecture on high-frequency spatial details, we conducted an experimental trial using downsampled 64×64 pixel inputs. Contrary to the intuitive assumption that higher resolution yields superior performance, this configuration achieved a Test Accuracy of 76.34% (Loss: 0.7382), performing only slightly worse than the 128×128 model.

This result empirically supports the hypothesis that our waste classification relies primarily on gross geometric morphology and spectral distribution rather than fine-grained texture. By reducing the resolution, we effectively introduce a noise-suppression mechanism. This prevents the model from overfitting to irrelevant high-frequency artifacts—such as text on a wrapper or surface grain—while preserving the global object structure essential for classification. Furthermore, this reduction increases the effective data density relative to the feature space volume, allowing the CNN to generalize more efficiently from the limited dataset ($N = 4,752$).

Class-specific performance metrics validate this theory. The most significant improvements were observed in classes characterized by amorphous shapes and distinct color profiles, rather than rigid edges. Specifically, accuracy for *Food Organics* rose from 80.5% (at 128×128) to 86.6%, and *Cardboard* improved from 80.4% to 82.6%. We postulate that for *Food Organics*, which is defined by a heterogeneous mix of colors (browns, greens, yellows), downsampling acts as a form of spectral averaging. This forces the model to focus on dominant color signatures rather than being distracted by complex, irrelevant surface textures. Repeated trials confirmed the statistical consistency of this improvement, suggesting that for specific material types, lower resolution serves as an effective regularization strategy when data is scarce.

4. Qualitative Error Analysis and Failure Modes

Despite the overall convergence of the CNN architecture, specific failure modes persist that highlight the challenges of waste sorting.

The most significant ambiguity occurs between *Plastic* and *Metal*. Unlike the FFNN, which defaulted to these classes due to statistical prevalence, the CNN's confusion appears to be structural. Qualitative examination identifies "Topological Mimicry" as the principal confounder.

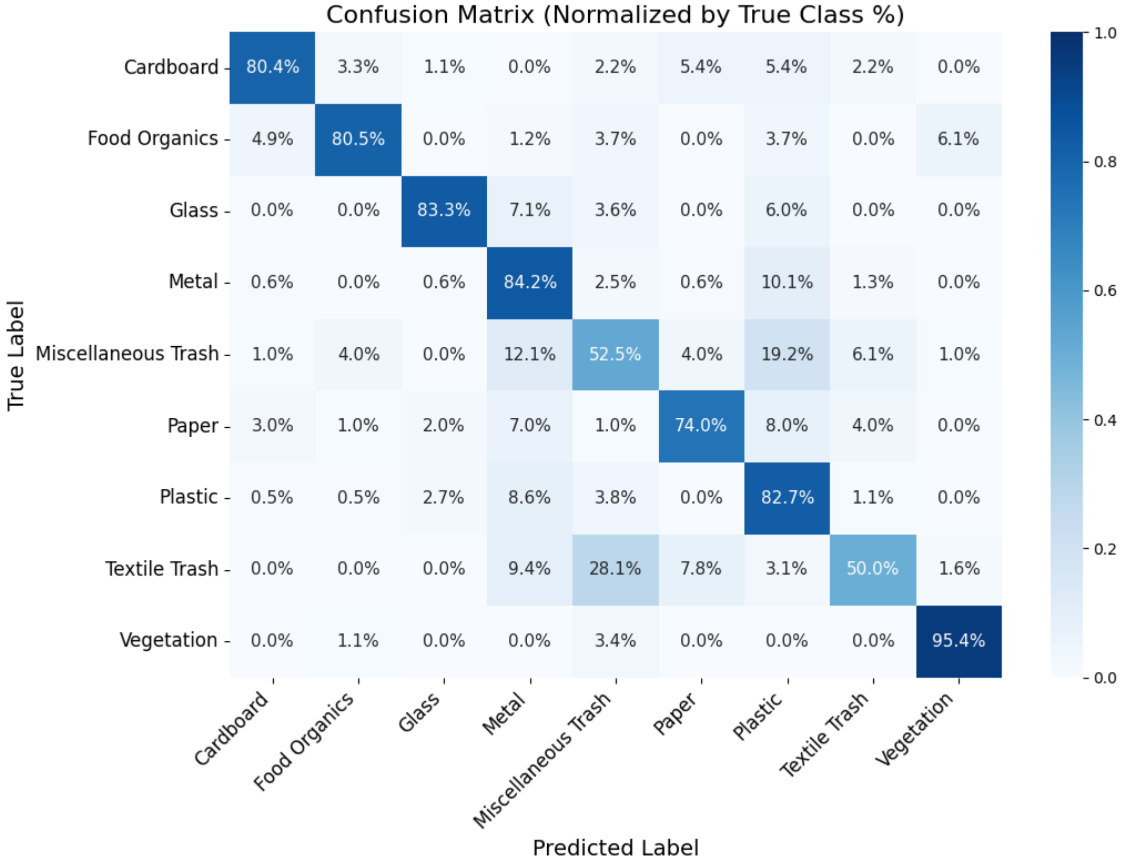

Figure 8. Confusion Matrix for the Augmented CNN. The pronounced diagonal dominance illustrates high classification fidelity, while off-diagonal elements highlight persisting texture-based ambiguities.

As illustrated in Figure 9, a crushed plastic bottle and a crushed metal can exhibit nearly identical geometric deformations. Both materials fail under stress by buckling, creating a complex network of sharp folds, cracks, and structural creases. Although the materials may possess distinct color profiles (e.g., translucent plastic vs. opaque painted metal), the network appears to prioritize these high-contrast structural features—the ”cracks” and ”folds”—over the spectral information. The convolutional filters successfully detect these high-frequency edges, but because the topology of ”crushed waste” is consistent across material classes, the model misinterprets the plastic deformation as metallic buckling.

Furthermore, the *Miscellaneous Trash* category remains a persistent bottleneck (30.3% accuracy). This represents an intrinsic limitation of supervised learning when applied to ”catch-all” categories defined by exclusion. Because this class encompasses disjointed objects—ranging from ceramics to electronics—it lacks a coherent centroid in the feature space. The resulting high intra-class variance prevents the network from learning a compact, discriminative representation, leading to undefined decision boundaries against the other eight distinct classes.

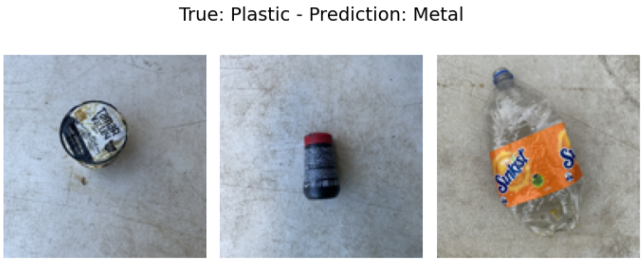

Figure 9. Analysis of False Positives. The model misclassifies a translucent plastic bottle as metal (Right). This error illustrates the reliance on specular reflection cues; the geometric shape of plastic mimics the crushed metallic surfaces, overriding the spectral features.

D. Comparison of Methods

Table III summarizes the performance of the tested architectures. The Convolutional Neural Network outperforms the optimized FFNN by over 21 percentage points (77.50% vs 50.89%). This performance disparity validates the fundamental hypothesis of Computer Vision that spatial topology, rather than global statistical variance, is the primary carrier of semantic information in unstructured image data.

Table III. Summary of Test Set Performance across architectures.

Model Configuration	Accuracy	Loss
FFNN (RGB, Raw Pixels)	$\approx 19\%$	> 2.0
FFNN (RGB, PCA 95%)	50.89%	1.5363
CNN (Grayscale)	62.15%	1.0900
CNN (RGB, No Aug)	68.24%	0.9096
CNN (RGB)	**77.50%**	**0.6734**
CNN (RGB, 64 × 64)	76.34%	0.7382

A comparative analysis of class-specific fidelity highlights the mechanism of this superiority. In the *Vegetation* category, the CNN achieves $\approx 95\%$ accuracy compared to the FFNN's 64.4%. Since both models utilized RGB data, this gap is not spectral but topological. This distinction is qualitatively analyzed in Figure 10, which contrasts the prediction pipelines for a specific paper sample. While the CNN successfully extracts the high-frequency edge geometry to correctly identify the object, the FFNN—limited by PCA compression—blurs this distinct edge geometry, resulting in a misclassification as Plastic. The CNN successfully extracts the high-frequency textural boundaries of leaves that PCA discards as noise. Similarly, the dramatic improvement in *Food Organics* (82.9% CNN vs 51.2% FFNN) illustrates the advantage of Max-Pooling over PCA. While PCA flattens the heterogeneous colors of organic waste into a global average, pooling operations preserve local chromatic clusters, allowing the CNN to identify the mix of colors characteristic of food waste.

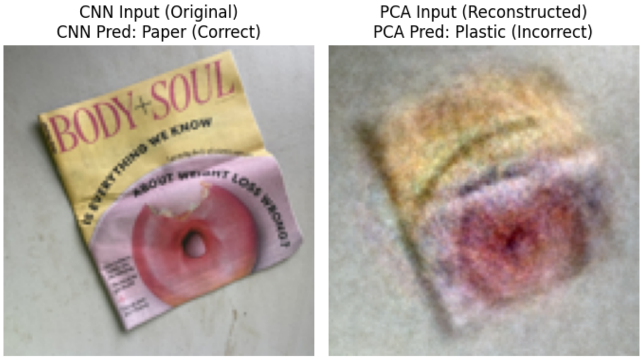
CNN Input (Original)
CNN Pred: Paper (Correct)
PCA Input (Reconstructed)
PCA Pred: Plastic (Incorrect)

Figure 10. Comparative Prediction Analysis. The CNN correctly identifies the paper sample by leveraging spatial edge geometry. In contrast, the FFNN (PCA) might misclassify the object as the reduction is blurring the distinct straight edges.

Ultimately, the computational trade-off heavily favors the CNN. While the dense FFNN with PCA is computationally cheaper per epoch, its performance ceiling is structurally limited by the loss of spatial correlation.

IV. FUTURE WORK

While our results demonstrate that the CNN architecture provides a robust baseline for waste segregation, several avenues remain for optimizing deployment in real-world infrastructure.

First, the hyperparameter optimization in this study was conducted primarily on the original dataset due to computational constraints. Given that the augmented dataset introduces significant geometric variance (rotations and flips), a secondary search on this expanded domain could yield an architecture better tuned to the rotation-invariant feature space.

Second, the current system classifies single, precropped images. To function effectively in a recycling plant, the pipeline must transition to an Object Detection framework (e.g., YOLO or R-CNN) capable of localizing and classifying multiple items simultaneously from a continuous video feed of a conveyor belt.

Finally, generalization remains a challenge. Waste packaging varies significantly by region. To minimize domain shift, future iterations should incorporate a data expansion strategy that specifically targets local packaging designs, ensuring the model is calibrated to the specific waste stream of the deployment region.

V. CONCLUSION

One of the primary constraints of this project was limited computational resources, which necessitated a careful balance between input resolution, model complexity, and training efficiency. This constraint framed the study as a rigorous evaluation of feature engineering, specifically questioning whether dimensionality reduction via PCA could compete with the higher computational cost of hierarchical feature extraction found in CNNs.

Our evaluation demonstrates that while Principal Component Analysis successfully compressed the input space to allow the dense FFNN to converge, it did so by discarding the high-frequency ”texture” of the data. With a best-case accuracy of 50.89%, the FFNN functioned largely as a color-based heuristic classifier. It failed to differentiate materials with similar chromatic profiles, such as *Textile* versus *Metal*, rendering it unsuitable for industrial applications where contamination rates must be minimized.

In contrast, the Convolutional Neural Network proved that spatial structure—edges, shapes, and textures—is the primary carrier of information in waste classification. Even when restricted to grayscale inputs, the CNN outperformed the color-based FFNN, indicating that geometric structure supersedes spectral information for feature extraction. The final optimized CNN, utilizing data augmentation to achieve rotation invariance, achieved a test accuracy of 77.50%.

While this result falls short of the 89.19% benchmark established by Single et al. [1] using the InceptionV3 architecture, it is important to contextualize this gap within the study's constraints. The state-of-the-art benchmark relied on transfer learning and high-fidelity inputs (524 × 524 pixels), whereas our custom CNN was trained from scratch on heavily downsampled inputs (128 × 128). Despite retaining only a fraction of the original pixel information, our model's ability to achieve within 12% of the state-of-the-art accuracy validates the robustness of convolutional filtering. It suggests that while higher resolution is necessary to resolve the finest ambiguities, the fundamental topological features required for broad classification are preserved even at lower resolutions.

Ultimately, despite the higher computational latency, we conclude that CNN architectures represent the only viable pathway for automated waste segregation. The performance cost of ignoring spatial topology—as evidenced by the failure of the PCA-based FFNN—is too high for practical deployment.

[1] S. Single, S. Iranmanesh, and R. Raad, *RealWaste: A Novel Real-Life Data Set for Landfill Waste Classification Using Deep Learning* (2023).

[2] S. Single, S. Iranmanesh, and R. Raad, *RealWaste* (2023), DOI: `10.24432/C5SS4G`.

[3] A. Maćkiewicz and W. Ratajczak, *Principal components analysis (PCA)* (1993).

[4] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (MIT Press, 2016) `http://www.deeplearningbook.org`.

[5] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, *Gradient-based learning applied to document recognition* (1998).

[6] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," (2017), arXiv:1412.6980 [cs.LG].

[7] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, *Dropout: a simple way to prevent neural networks from overfitting* (2014).

[8] A. N. Torgersen, *Project 3 - FYS-STK4155* (2025).

[9] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, *Array programming with NumPy* (2020).

[10] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems* (2015), software available from tensorflow.org.

[11] F. Chollet *et al.*, *Keras* (2015).

[12] T. O'Malley, E. Bursztein, J. Long, F. Chollet, H. Jin, L. Invernizzi, *et al.*, *Keras Tuner* (2019).

[13] J. D. Hunter, *Matplotlib: A 2D graphics environment* (2007).