Cureus Journal of Computer Science

Open Access Original Article

Eco-Evolution: Transforming Waste Management Through Convolutional Neural Network Parameter Evolution With Optimizers

Received 04/03/2025
Review began 04/10/2025
Review ended 06/24/2025
Published 06/27/2025

DOI: https://doi.org/10.7759/s44389-025-04505-4

Mohsin Ali Shah [1] , Majeed Jan [1] , Ansar Ullah [1]

1. Department of Computer Systems Engineering, University of Engineering and Technology, Peshawar, PAK

Corresponding author: Mohsin Ali Shah, mohsinalishah13@gmail.com

Abstract

Recycling waste materials is critical to reducing the environmental effects of increased waste production. Proper waste separation is critical for speeding recycling procedures. This research uses advanced optimization algorithms to handle the difficulty of classifying waste materials. The complexity of waste material composition and diversity pose a significant challenge for accurate classification, requiring advanced classification algorithms. Convolutional neural networks (CNNs) have developed as effective methods for image classification, such as waste material sorting. However, adjusting CNN parameters to optimize classification accuracy remains a key area of study. This paper suggests using genetic algorithms (GA) and particle swarm optimization (PSO) to improve the performance of CNNs in waste material classification. The goal is to improve classification accuracy while minimizing loss by fine-tuning CNN's dense layer, which includes the number of dense layers, number of neurons, and activation functions. A dataset containing images of 12 waste material types, such as food, plastic, cardboard, glass, metal, paper, and others, is used for experimental assessment. Based on the results of this research finding, it can be concluded that GA and PSO have succeeded in optimizing and increasing the accuracy of the test to 85.32% and 85.42%, respectively, compared to the baseline of 76.78%. PSO also converged more rapidly and completed the optimization process only about 10% faster and with the lower final loss (0.545 against 0.579), whereas in the only final generation, the closer ties of losses were seen by GA. The proposed technique can enhance waste management process through better sorting and recycling of the waste materials. Future studies will be conducted on hybrid metaheuristic strategies and continue optimization to the convolutional layers in classifying waste.

Categories: AI applications, Algorithm Analysis, Artificial Intelligence and Machine Learning in the Cloud
Keywords: waste classification, genetic algorithm (ga), particle swarm optimization (pso), convolutional neural network (cnn), optimization techniques, artificial intelligence

Introduction

Recycling waste materials is necessary because of the decreasing natural resources [1]. Recycling of waste materials is a building block for a sustainable society. If the waste materials are not recycled in a proper way, they can create many problems, like air pollution, water pollution, and various diseases in living organisms. The present waste material separation and recycling processes require authorities to sort garbage by hand and use various filters to separate out objects. The motivation of this work is to find an optimized automatic technique for sorting waste materials. This will help to make recycling plants more efficient and help reduce the adverse effects of waste materials. Sorting the waste materials manually takes a lot of time and is not 100% accurate. Manual sorting can increase the cost of recycling and have a harmful effect on the health of the waste sorting workers [2]. This will not only have positive environmental effects but will also have a constructive effect on the economic growth of a country [3,4]. In addition, the application of an advanced waste classification system can save a lot of us from environmental risks simply by minimizing the improper disposal and increasing the recycling rates. Organizations that are devoted to protecting the environment like World Wide Fund For Nature acknowledge the fact that the effective waste management is vital for the protection of natural ecosystems and public health [5,6]. There are intellectual and environmentalist groups whose members are active in their local areas, advancing their objectives. Automation of garbage classification is very necessary nowadays due to the increasing garbage production in the world. The average waste generated by a person per day is 0.58 kg [7]. With the current rapid growth in global garbage production, solid waste will reach up to 2.2 billion tons per year by the year 2025, and the cost of disposing of the waste will be approximately 375.5 billion dollars [8]. The increasing garbage production has a bad effect on the environment. The utilization of garbage materials and the optimization of the garbage disposal process are urgent issues that are currently a major concern for countries around the world. There are two main challenges in the current recycling of waste materials: the garbage is sorted manually by workers, which is not efficient, and classifying the garbage material manually is not accurate. It is an important task to solve these two problems in the recycling process of waste materials. Convolutional neural networks (CNNs) have emerged as excellent tools for automating trash categorization tasks, owing to their ability to extract intricate features from waste material images.

How to cite this article
Ali Shah M, Jan M, Ullah A (June 27, 2025) Eco-Evolution: Transforming Waste Management Through Convolutional Neural Network Parameter Evolution With Optimizers. Cureus J Comput Sci 2 : es44389-025-04505-4. DOI https://doi.org/10.7759/s44389-025-04505-4

However, tuning CNN parameters for optimal classification accuracy remains a complicated and difficult task. Traditional optimization approaches may be insufficient for successfully navigating the high-dimensional parameter space. Metaheuristic optimization approaches provide interesting opportunities for improving CNN performance in garbage classification problems. Genetic algorithm (GA) and particle swarm optimization (PSO) are two metaheuristic algorithms that are well known for their capacity to effectively explore and exploit solution spaces in search of near optimal solutions [9,10]. GA mimics natural selection by developing candidate solutions over multiple generations, whereas PSO replicates swarming particle behavior to iteratively enhance solution quality. This work suggests using GA and PSO to optimize CNN's dense layer for garbage classification, with the goal of improving classification accuracy and streamlining waste management operations. By utilizing the strengths of these metaheuristic algorithms, we seek to enhance the efficiency of garbage classification systems, thus contributing to sustainable waste management techniques.

Garbage disposal has become a serious problem in recent years, owing to the increased use of single-use items [5]. This encompasses everything from bottled water to throw-away coffee cups, packing foam to medical waste, light bulbs to plastic bags. Several ways to categorize garbage have been developed utilizing computer vision systems. Juan Carlos et al. [11] demonstrated an automated garbage sorting system intended for Colombian high schools. The system consists of three modules: image capture with a webcam in a regulated lighting box; image processing using grayscale image correlation against a database of trash photos; and a robotic system that opens the relevant waste bin based on categorization. The method achieved testing accuracy of 75%, 72%, and 87% for plastic bottles, soda cans, and cardboard boxes, respectively. In comparison research, Sakr et al. [12] compared CNN using AlexNet to support vector machines (SVM) using a bag of features technique for waste image categorization. SVM outperformed CNN (83%), with an accuracy of 94.8%. The SVM model was installed on a Raspberry Pi 3 to do image-based waste sorting. Nnamoko et al. [13] suggested a unique five-layer CNN architecture for classifying garbage images into recyclable and organic categories. Utilizing an enhanced version of Sakr's waste products categorization dataset for training, the more compact 80 × 45 resolution model demonstrated 80.88% testing accuracy, surpassing the performance of the larger 225 × 264 model (76.19%), despite its lighter weight and shorter training duration. Chhabra et al. [14] proposed a unique CNN architecture that incorporates leaky ReLU activation, batch normalization, max pooling, and dropout for categorizing images into food waste and general waste categories. On the given dataset, their architecture achieved an F1-score of 87.12%, an accuracy of 88.54%, and a recall of 87.62%. The EfficientNet-B0 CNN model, which was pre-trained on ImageNet, was suggested to be used by Malik et al. [15] to categorize municipal solid waste images into 34 groups. An accuracy of 81.2% was attained while requiring four times fewer computer operations than the bigger EfficientNet-B3 model through the fine-tuning of a region-specific waste image dataset. Using a frozen state for pre-trained layers and a waste dataset to train new layers, the approach included transfer learning. Altikat et al. [1] classified images of plastic, glass, paper, and organic garbage using deep convolutional neural networks (DCNN) with four- and five-layer architectures, respectively. The five-layer DCNN attained a classification accuracy of 70%, whereas the four-layer DCNN scored 61.67%. Song et al. [8] developed an algorithm, which was inspired by Inception-V4 and ResNet network. The developed algorithm can detect four types of waste material with an accuracy of 94.38%. Singh et al. [2] developed a deep learning algorithm to classify waste as biodegradable and non-biodegradable. Sudha et al. [16] used SVM and CNN to classify images with garbage and images which did not contain any garbage material. CNN was optimized using GA to classify six types of waste materials, and an accuracy of 99.6% was achieved on the TrashNet dataset [3]. Zhang et al. [17] achieved an accuracy of 95.87% on the TrashNet dataset using CNN. Zhang et al. [18] proposed DenseNet169 garbage image classification deep learning model based on transfer learning, and an accuracy of 82% was achieved. A deep learning model was trained to achieve an accuracy of 87% for classifying four types of waste materials, namely plastic, metal, glass, and paper [19]. Inception-v3 was used for the classification of six types of waste, and an accuracy of 92.5% was achieved [20]. Smart dustbin was able to differentiate six types of waste materials with an accuracy of 92% using CNN [21]. In order to automatically sort the garbage generate in metropolitan areas, Chu et al. [22] presented a multilayer hybrid deep learning system. By analyzing various fertilizing phase images, Xue et al. [23] offered CNN as a means of achieving quick fertilizer analysis. When Aral et al. [24] examined many deep learning models for categorizing the TrashNet dataset, they examined DenseNet, Inception ResNet, MobileNet, and Xception structures. Nowakowski and Pamula [25] put forward a novel approach to categorizing and identifying electronic waste. Sousa et al. [26] proposed a faster region-based CNN that is hierarchical and can recognize and categorize trash in food trays. ResNet-based intelligent trash categorization was proposed by Adedeji and Wang [27]. Using preprocessing and classification, Mazloumian et al. [28] suggested DNN for the categorization of food waste. In a study by Vyas et al. [29], the combination of image augmentation techniques with a ResNeXt-50 architecture was shown to achieve validation accuracies greater than 99% on waste datasets, indicating that deep learning methods are scalable on real-world waste sorting scenarios. Caballero et al. [30] used CNNs to predict recyclable objects from complex waste images and achieved top-1 accuracies of up to 99% on bespoke datasets, using model quantization and other techniques to enable deployment on resource-constrained devices. Yu and Grammenos [31] employed transfer learning paired with a systematic tuning of the training parameters including learning rate schedulers and layer freezing to boost image processing for waste classification increasing test accuracy from 91.21% to 95.40% via data augmentation. The end of human intervention in waste sorting was proposed by Shah and Kamat [32] who proposed to use CNNs to distinguish between organic and recyclable waste having an accuracy of 94.9%. Nasir and Aziz Al-Talib [33] critically compared the various AI techniques for waste classification and also pointed out the challenges existing in using limited training datasets, with the need for strategies of augmented data. Ranjbar et al. [34] suggested a deep learning method for the recognition and sorting of waste plastic bottles based on RGB and infrared image inputs. They employed RevCol as the backbone network, with a lightweight decoupling head, and used the WIoU v3 loss function to increase both detection accuracy and efficiency. Likewise, Xie et al. [35] introduced a deep learning system to machine-learn the resin type of construction and demolition plastic waste from RGB images. By leveraging transfer learning using pre-trained CNN and Vision Transformer models, ResNet, RegNet, and Swin Transformer, and by distilling knowledge from the Swin Transformer into MobileNetV3, this approach improves the performance of lightweight models. Collectively, these contributions illustrate how deep learning-based waste classification systems can rapidly evolve. In addition to providing innovative solutions for sorting municipal and electronic waste, they offer an important basis for smart city applications aimed at improving recycling efficiency and environmental sustainability.

In this section, we also discuss the components used in our proposed method. In particular, we discuss the basics of the CNN, GA, and PSO, three building blocks used in this study. In the next section, we will discuss the proposed method for evolving the CNN through GA and PSO for waste material classification.

Convolutional neural network

CNN is one of the most popular types of neural networks. It can achieve high accuracy in problems involving image data. CNN is composed of layers, as shown in Figure *1*, such as the convolutional layer, pooling layer, nonlinearity layer, and dense layer [36]. It helps reduce the number of parameters in the neural network. The convolutional layer is the second layer in the CNN, following the input layer, which is typically an image or video. One of the most important components of the convolutional layer is the kernel [37]. The dot product of the kernel and the input pixels is computed to produce feature maps. The convolutional layer helps in detecting lines, edges, curves, and colors. Features in the images are detected by multiplying the kernel values with the pixels of the image from left to right.

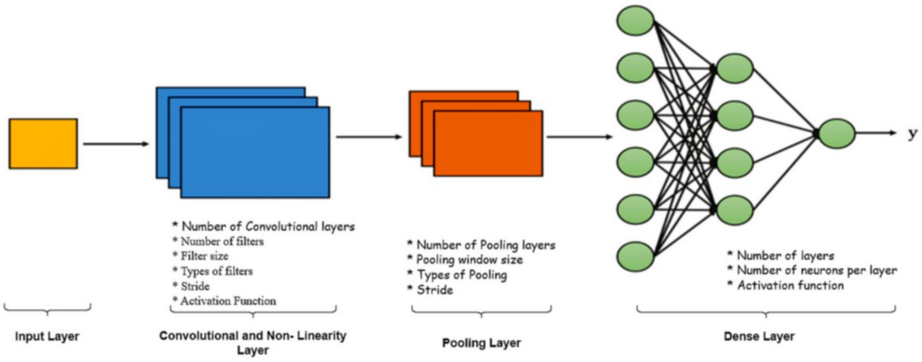

FIGURE 1: Layers in convolutional neural network

The third layer of CNN is the non-linearity layer. The main component of this layer is the activation function. The function of the activation function in the non-linearity layer is the same as its function in other neural networks. Its main purpose is to normalize the image. One of the most popular activation functions used by CNN is ReLU. The fourth layer in CNN is the pooling layer. The pooling layer is used to reduce the number of parameters, so it takes less time to train CNN, and the problem of overfitting is also solved to some extent. On the feature map, the pooling operation is applied. Usually, a 2×2 filter is used for the pooling operation. Mean, max, and min are some types of pooling operations. The final layer of CNN is known as the dense layer. It is called a dense layer because the neurons in each layer are fully connected to all the neurons in the next layer. The task of this layer is to classify the images. The main components of this layer are layers, neurons, and activation functions. Output layer neurons are the same as the number of classes in the problem to be solved.

Genetic algorithm

GA is an algorithm used for finding the exact solution or a close-to-exact solution for optimization and search problems [38]. It is a population-based optimization algorithm. In these types of algorithms, new solutions are generated from the current solutions in order to find better results for a given problem. Each population in the GA has a number of solutions called individuals. Each individual consists of a chromosome, and the chromosome consists of genes. There is a fitness value associated with each individual, which is computed through a fitness function. GAs can have multiple generations in order to find the optimal solution for the optimization problem. The first generation of GA is populated randomly. Individuals having good fitness values produce a good solution. In the selection stage, the individuals having the best fitness value among the population are selected for the crossover stage to produce better individuals so better results can be achieved, and for diversity, some of the individuals with low fitness scores are also selected. In the crossover stage, two parents are selected from the selection stage, and an offspring is produced with a mixture of features from both parents. In the mutation stage, some of the children are randomly selected, and their features are randomly mutated in order to produce offspring with better features. However, it is not guaranteed that the mutated features are always better than the features that are inherited by the parents in the crossover stage.

Particle swarm optimization algorithm

PSO was proposed by Kennedy and Eberhart in 1995. The PSO algorithm is inspired by the social behavior of some animal groups. It can be more difficult for animals to find food alone than when they are searching for food in groups. PSO is an optimization algorithm that can solve complex problems with a few lines of computer code [39]. PSO is computationally less expensive than other optimization algorithms in terms of memory and speed. In PSO, agents (also called particles) are the main components of a swarm. The agents move in the search space in order to find an optimal solution. The best position achieved by each agent is known as their personal best. The best value achieved by the neighboring agents of any agent is known as the global best [10]. Some of the parameters of particle swarm optimization are agents, velocity, cognitive constant, random numbers, social constant, position of the agent, personal best of the agent, and global best. The algorithm of PSO can be simulated for several iterations until an optimal solution is found [40]. There are built-in functions for the PSO algorithm in many programming languages, like Python, Java, and C++. PSO is usually used for minimization problems. There are many neighborhood topologies for PSO, such as star topology, ring topology, and wheel topology [41].

Materials And Methods

In this section, the dataset and techniques used for evolving CNNs through optimization algorithms such as GA and PSO are discussed.

Waste dataset

In this research, the waste image dataset [42] used for training, validation, and testing of the neural network comprises 12 categories of waste materials: food, plastic, cardboard, green glass, transparent glass, brown glass, metallic waste, paper, general waste, batteries, garments, and footwear. The dataset contains a total of 16,148 images, which were divided into 85% for training (13,725 images), 10% for validation (1,615 images), and 5% for testing (808 images). The images were resized to 220 × 220 pixels. To prevent overfitting, data augmentation techniques, including random rotations, flips, scaling, and brightness jitter, were applied to the training set. Each CNN architecture, whether baseline or optimized, was trained using the augmented training data (85%), with hyperparameters tuned on the validation set (10%). Final accuracy and loss metrics were reported on the test data, which was held out using 5% of the data.

Structure of deep CNN

The constant parameters used in the CNN are listed in Table *1*.

Parameters	Values
Convolutional layers	6
Activation function of convolutional layer	ReLU
Pooling layers	6
Step size	2
Optimizer	Adam
Loss function	Categorical cross-entropy
Leaning rate	0.0001

TABLE 1: Constant parameters of convolutional neural network

ReLU, Rectified Linear Unit

The CNN used in this study consists of six convolutional layers and six pooling layers. A step size of 2 is applied in both the convolutional and pooling layers. Techniques such as dropout and batch normalization are employed to mitigate overfitting, while data augmentation is used to expand the dataset. The Adam optimizer, recognized for its effectiveness in training deep neural networks, is utilized with a learning rate set to 0.0001. Categorical cross-entropy, chosen for its suitability in problems with multiple output classes, is employed as the loss function.

Baseline CNN architecture

To measure the advantage of optimization, we first took a baseline CNN with no optimization. It included six blocks of Conv2D, Batch Normalization, ReLU, and MaxPooling (2 × 2), and two fully connected layers with 512 neurons (ReLU) and Dropout (0.5). The output layer had 12 neurons with softmax activation. Its compilation uses the Adam optimizer (learning rate 0.0001) and categorical cross-entropy loss, and it was trained on 30 epochs.

Evolving CNN's dense layer with GA and PSO

In this study, GA and PSO are used to find a deep neural network with a minimum loss score for classifying waste materials. The number of dense layers and the parameters in the dense layer, which are the number of neurons and activation functions, are modified through these optimizers to find an optimal neural network. The search space is shown in Table *2*. The modified parameters are the same in all dense layers if optimizers modify the number of layers to more than one layer.

Parameters	Search space
Layers	1, 2, 3, 4
Number of Neurons	64, 128, 256, 512, 768, 1,024
Activation functions	ReLU, ELU, Tanh, Sigmoid

TABLE 2: Search space

ReLU, Rectified Linear Unit; ELU, Exponential Linear Unit; Tanh, Hyperbolic Tangent

The GA used for evolving the CNN consists of these steps:

1. We created five neural networks using the parameters listed in Table *2*. They were the individuals in the population for the first generation, as shown in Figure *2*.

2. The loss of each network is calculated on the test dataset after training the model in the first generation. Classification loss is used as a fitness function.

3. For the next generation, the networks in the first generation are sorted by their loss score. Networks with low loss scores are retained, and some of the high-loss networks are also retained for maintaining diversity.

4. Crossovers are done between the neural networks retained in Step 3 to produce children.

5. Some of the children were mutated randomly to produce better neural networks.

6. The GA had four generations, with each population having five neural networks. So in total, 20 neural networks were trained.

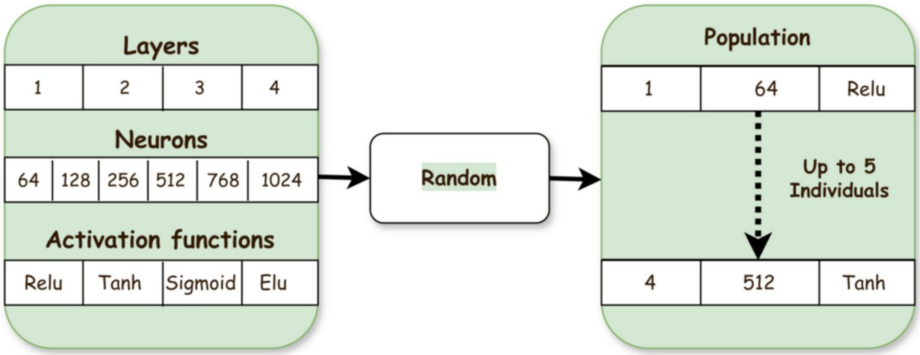

FIGURE 2: First generation

ReLU, Rectified Linear Unit; ELU, Exponential Linear Unit; Tanh, Hyperbolic Tangent

PySwarms, which is a Python library for PSO, was used for evolving CNN with PSO. There were five agents generated for this study, and the PSO algorithm was run for four iterations. The steps to evolve CNN by PSO are explained as follows:

1. The population of agents is generated for the PSO algorithm. The parameters of PSO are set, such as the number of iterations, number of particles, social constant, cognitive constant, inertial weight, and two random numbers.

2. Then, the CNN is trained with the parameters obtained by the PSO algorithm.

3. The loss value of CNN is used as the objective function of PSO.

4. Then, the objective function of the PSO algorithm is evaluated.

5. The parameters of PSO are updated. Each agent updates its position and velocity, which depends on its best position and the best position in the whole swarm.

6. The best solution is selected that gives the lowest loss value among all networks.

7. The search space for particles used is presented in Table *3*.

Particle coordinates	Parameters	Search space range
X1	Number of layer	1, 2, 3, 4
X2	Number of neurons	64, 128, 256, 512, 768, 1,024
X3	Activation function	ReLU, Tanh, Sigmoid, ELU

TABLE 3: Search space of particles

ReLU, Rectified Linear Unit; ELU, Exponential Linear Unit; Tanh, Hyperbolic Tangent

Results

Each model was trained for 30 epochs. The dense layer was optimized using GA and PSO, which considered the number of dense layers, neurons, and activation functions. GA operated through four generations, each consisting of five individuals, whereas PSO ran for four iterations, each containing five particles. This approach generated 20 neural networks with different dense layer configurations in the CNN for garbage categorization. All models (baseline, GA optimized, and PSO optimized) were trained using an NVIDIA K80 GPU.

Model	Test accuracy (%)	Test loss
Baseline CNN	76.78	0.9359
GA-Optimized CNN	85.32	0.5790
PSO-Optimized CNN	85.42	0.5450

TABLE 4: Accuracy and loss comparison: Baseline vs. GA optimized vs. PSO optimized

CNN, Convolutional Neural Network; GA, Genetic Algorithm; PSO, Particle Swarm Optimization

Table 4 illustrates that both GA and PSO significantly improve performance over the non-optimized baseline, increasing accuracy from 76.78% to approximately 85.4%. GA achieves an accuracy of 85.32% with a test loss of 0.5790, while PSO achieves 85.42% accuracy with a test loss of 0.5450, an improvement of about 8.6 percentage points. These results indicate that dense layer optimization is highly effective and supports further analysis of convergence behavior and computational overhead.

Generations no.	Best-result individuals	Parameters			Accuracy	Loss
		Layers	No. of neurons	Activation function		
1	1	1	512	ReLU	0.8271	0.7009
2	3	1	768	ReLU	0.8209	0.5987
3	4	1	512	ELU	0.8532	0.5790
4	3	4	64	ReLU	0.8423	0.6250

TABLE 5: Results of genetic algorithm

ReLU, Rectified Linear Unit; ELU, Exponential Linear Unit

Iterations no.	Best-result particles	Parameters			Accuracy	Loss
		Layers	No. of neurons	Activation function		
1	4	2	1,024	Sigmoid	0.8542	0.5514
2	1	2	512	Sigmoid	0.8475	0.5450
3	1	2	768	Tanh	0.8508	0.5477
4	5	2	768	ELU	0.8443	0.5662

TABLE 6: Results of particle swarm optimization algorithm

ELU, Exponential Linear Unit; Tanh, Hyperbolic Tangent

The results are further summarized in Tables 5 and 6, which present the best-performing individuals from each GA generation and the best-performing particles from each PSO iteration, respectively, along with their model configurations, accuracy, and loss. GA achieved a maximum accuracy of 85.322% (Generation 3) with a loss of 0.5790. The generation accuracies for GA ranged from 82.092% to 85.322%, and the losses ranged from 0.5790 to 0.7009 across the four generations. In comparison, the best accuracy achieved by PSO was 85.42% (Iteration 1) with a loss of 0.5514. The accuracies of PSO's particles ranged from 84.43% to 85.42%, and the losses ranged from 0.5514 to 0.5662. Overall, PSO produced more high-performing models per iteration than GA.

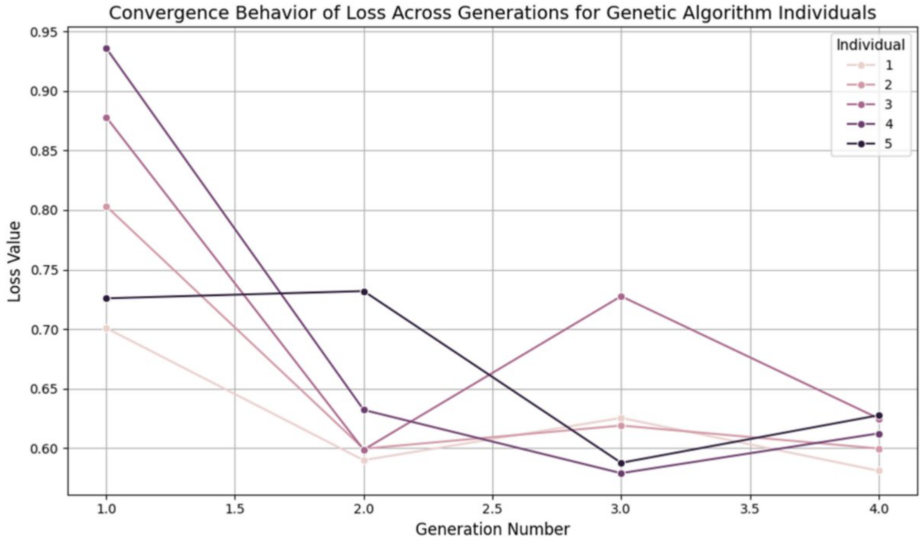

FIGURE 3: Genetic algorithm loss convergence

FIGURE 4: PSO loss convergence

PSO, Particle Swarm Optimization

Figure 3 indicates that in Generation 1, individuals in GA start at high and spread loss values (i.e., 0.7009-0.9359), representing slow initiation and a large gap among solutions. At Generation 4, the optimal GA loss further decreases to 0.5810, but the range is very wide (up to 0.6276), which also shows that GA is prone to searching an extensive solution area. Figure 4 displays the behavior of PSO. At Iteration 2, its best loss becomes 0.5450, and at Iteration 3, it reaches the overall best of 0.5385. The formation of the losses by PSO at Iteration 4 extends between 0.5662 and 0.6865. It is interesting to note that three of the five PSO particles (0.5662, 0.5717, and 0.5817) have their final solutions less than the GA best (0.5810), and two particles have their solutions greater than the worst solution of GA (0.6865 and 0.6300). This shows that PSO not only converges quicker and lands at minimal minima but also has far more solutions that surpass the best solution in GA, even though some runs are somewhat slower.

These observations indicate that PSO not only converges more quickly and consistently reaches lower minima but also produces more high-quality solutions than GA, despite some variability in performance across iterations.

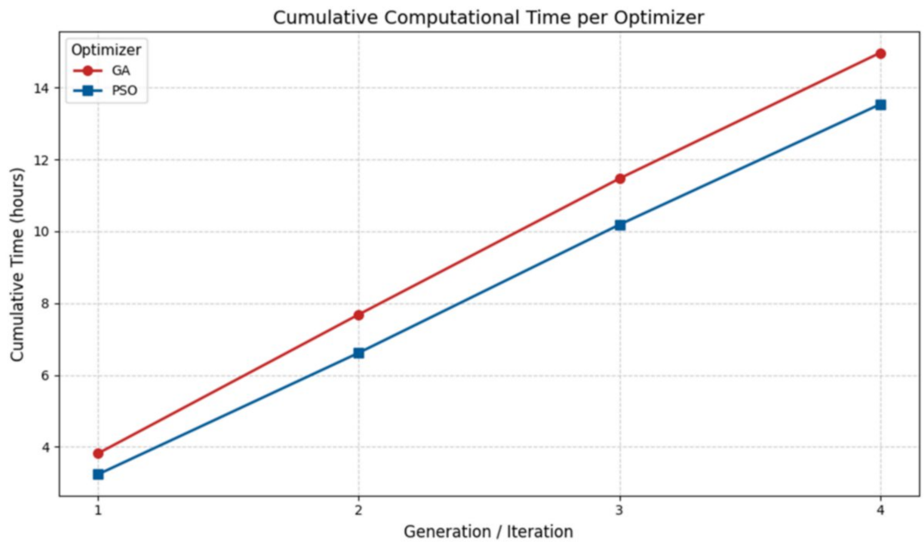

FIGURE 5: Compute time comparison

GA, Genetic Algorithm; PSO, Particle Swarm Optimization

Figure *5* is a comparison of total computation time. PSO takes about 13.5 hours to run its four iterations, and GA requires 14.58 hours, which is almost 15 hours to run the same number of individuals, a time difference of about 10%. Overall, PSO is significantly faster in converging and frequently finds many better solutions in less time than GA, which delivers more uniform though higher losses because of the larger exploration.

Discussion

The experimental findings clearly showed that both PSO and GA brought significant improvements to the waste classification accuracy over the non-optimized baseline model. Both optimization techniques performed well above 84% on test accuracy, with PSO achieving the highest test accuracy of 85.42%, compared to GA's 85.32%. PSO also resulted in lower values of loss between iterations (best loss of 0.5450) than GA (best loss of 0.5790). Both GA and PSO showed a progressive decrease in the values of loss across generations and iterations, and by the last stage, reliable convergence was observed. In the fourth generation, it was interesting to note that GA had a close final cluster of loss values, with most individuals clustering around 0.580-0.630, in comparison to PSO, which was slightly more dispersed in the fourth iteration, with loss values ranging from 0.566 to 0.686. This showed that PSO achieved smaller peak losses and better convergence at the initial stages. GA, in turn, had a more consistently good population in its final generation, indicating its local-search enhancement ability in later iterations.

A notable point of observation is that PSO reliably found high-performing configurations with less variation when compared to GA. The evolutionary characteristic of GA supports varied solutions but is more susceptible to variations in performance, owing to the risk of getting stuck in local optima. PSO, conversely, features the swarm intelligence mechanism that ensures a fast convergence and comparably high accuracy with minimized loss fluctuation. It is, however, worth mentioning that GA has a larger exploration space, which means that it has the potential to search through more candidate solutions, which may prove useful in problems that need diversity in architecture.

Moreover, one of the constraints that were applied to this study is that every dense layer of a specific CNN will have the same number of neurons and activation functions. Although this simplification also led to quicker convergence in both optimization approaches, it could have limited the discovery of more diverse architectures. It was noted that such combinations of hyperparameters as a single dense layer with ELU activation and 64 neurons in the case of GA, and two dense layers with Sigmoid activation and 1024 neurons in PSO, were observed in models that performed well. These findings indicate that there is a need to investigate more dynamic level-wise optimization of hyperparameters in the future to possibly attain even greater classification performance. Additionally, the computational study showed that PSO needed a time of about 13.5 hours to complete its four optimization iterations, whereas GA needed a time of about 15 hours to reproduce four generations, implying faster convergence of PSO under the same hardware specifications (NVIDIA K80 GPU). This also proves the usefulness of PSO in situations that involve less computation power or when fast convergence of a model is important.

However, this work only considered tuning the dense layer hyperparameters, but future work will explore hyperparameter optimization on convolutional-layer parameters like kernel size, learning rate, dropout rate, and optimizer settings. Further, the usage of hybrid metaheuristic methods, where the global searching abilities of the GA are implemented together with the fast convergence properties of the PSO, will offer a proactive trend in the further enhancement of the CNN-based waste classification systems.

Conclusions

For waste classification from 12 categories, GA and PSO were applied to optimize the number of dense layers and their parameters in CNNs. The results of both techniques were competitive in that GA achieved an accuracy of 85.32%, and PSO overtook it, achieving a maximum accuracy of 85.42%. PSO was also more stable in its performance, as it recorded lower test losses (0.5450 as compared to 0.5790 by GA) and faster times of convergence. As comparative analysis shows, PSO, due to the presence of the faster convergence and more reliable loss values, is a suitable candidate in a condition where there is a limit on computational resources or stability of the model is of key concern. However, the wide search capability that GA has could enable it to find other architectures that may give valuable and alternative solutions to difficult tasks. In addition, optimization of the same number of neurons and activation functions on each dense layer enhanced convergence, but this limitation might have reduced the exploration space and consequently architectural variety.

These findings also affirm the usefulness of the evolutionary algorithms in optimizing the CNN hyperparameters in waste management. Computational reasons made it necessary to keep the number of training epochs constant (30) across all experiments. Future research could thus explore increased training time on stronger GPUs, examine the tuning of convolutional-layer parameters, and explore hybrid metaheuristic methods or layer-wise flexible tuning techniques. These enhancements can lead to more improvements in classification accuracy and model generalization in waste classification and other computer vision tasks.

Additional Information

Author Contributions

All authors have reviewed the final version to be published and agreed to be accountable for all aspects of the work.

Concept and design: Mohsin Ali Shah

Acquisition, analysis, or interpretation of data: Mohsin Ali Shah, Majeed Jan, Ansar Ullah

Drafting of the manuscript: Mohsin Ali Shah, Majeed Jan, Ansar Ullah

Critical review of the manuscript for important intellectual content: Majeed Jan

Disclosures

Human subjects: All authors have confirmed that this study did not involve human participants or tissue. **Animal subjects:** All authors have confirmed that this study did not involve animal subjects or tissue. **Conflicts of interest:** In compliance with the ICMJE uniform disclosure form, all authors declare the following: **Payment/services info:** All authors have declared that no financial support was received from any organization for the submitted work. **Financial relationships:** All authors have declared that they have no financial relationships at present or within the previous three years with any organizations that might have an interest in the submitted work. **Other relationships:** All authors have declared that there are no other relationships or activities that could appear to have influenced the submitted work.

References

1. Altikat A, Gulbe A, Altikat S: Intelligent solid waste classification using deep convolutional neural networks. International Journal of Environmental Science and Technology. 2021, 19:1285-1292. 10.1007/s13762-021-03179-4
2. Singh S, Mehta KS, Bhattacharya N, Prasad J, Lakshmi SK, Subramaniam KV: Identifying uncollected garbage in urban areas using crowdsourcing and machine learning. 2017 IEEE Region 10 Symposium (TENSYMP). 2017, 1-5. 10.1109/TENCONSpring.2017.8070078
3. Mao W-L, Chen W-C, Wang C-T, Lin Y-H: Recycling waste classification using optimized convolutional neural network. Resources, Conservation and Recycling. 2021, 164:105132. 10.1016/j.resconrec.2020.105132
4. Kanaparthi SK, Reddy CK, Sharma TVSR, Reddy KAK, Reddy NS, Vishnu A: Classification of waste in natural environments. MATEC Web of Conferences. 2024, 392:01167. 10.1051/matecconf/202439201167
5. Mohammed MA, Abdulhasan MJ, Kumar NM, et al.: Automated waste-sorting and recycling classification using artificial neural network and features fusion: a digital-enabled circular economy vision for smart cities. Multimedia Tools and Applications. 2023, 82:39617-39632. 10.1007/s11042-021-11537-0
6. Rahman AU, Saeed M, Mohammed MA, Abdulkareem KH, Nedoma J, Martinek R: Fppsv-NHSS: Fuzzy parameterized possibility single valued neutrosophic hypersoft set to site selection for solid waste management.

Applied Soft Computing. 2023, 140:110273. 10.1016/j.asoc.2023.110273

7. Darban Astane AR, Hajilo M: Factors affecting the rural domestic waste generation. Global Journal of Environmental Science and Management. 2017, 3:417-426. 10.22034/gjesm.2017.03.04.007
8. Song F, Zhang Y, Zhang J: Optimization of CNN-based garbage classification model. CSAE '20: Proceedings of the 4th International Conference on Computer Science and Application Engineering. 2020, 107. 10.1145/3424978.3425089
9. Katoch S, Chauhan SS, Kumar V: A review on genetic algorithm: past, present, and future. Multimedia Tools and Applications. 2021, 80:8091-8126. 10.1007/s11042-020-10139-6
10. Gad AG: Particle swarm optimization algorithm and its applications: A systematic review. Archives of Computational Methods in Engineering. 2022, 29:2531-2561. 10.1007/s11831-021-09694-4
11. Juan Carlos BL, Alejandro R, Manuel E: Automatic waste classification using computer vision as an application in Colombian high schools. 6th Latin-American Conference on Networked and Electronic Media (LACNEM 2015). 2015, 1-5. 10.1049/ic.2015.0316
12. Sakr GE, Mokbel M, Darwich A, Khneisser MN, Hadi A: Comparing deep learning and support vector machines for autonomous waste sorting. 2016 IEEE International Multidisciplinary Conference on Engineering Technology (IMCET). 2016, 207-212. 10.1109/IMCET.2016.7777453
13. Nnamoko N, Barrowclough J, Procter J: Solid waste image classification using deep convolutional neural network. Infrastructures. 2022, 7:47. 10.3390/infrastructures7040047
14. Chhabra M, Sharan B, Gupta K, Astya R: Waste classification using improved CNN architecture. Proceedings of the Advancement in Electronics & Communication Engineering. 2022, 10.2139/ssrn.4157549
15. Malik M, Sharma S, Uddin M, Chen C-L, Wu C-M, Soni P, Chaudhary S: Waste classification for sustainable development using image recognition with deep learning neural network models. Sustainability. 2022, 14:7222. 10.3390/su14127222
16. Sudha S, Vidhyalakshmi M, Pavithra K, Sangeetha K, Swaathi V: An automatic classification method for environment: Friendly waste segregation using deep learning. 2016 IEEE Technological Innovations in ICT for Agriculture and Rural Development (TIAR). 2016, 65-70. 10.1109/TIAR.2016.7801215
17. Zhang Q, Zhang X, Mu X, Wang Z, Tian R, Wang X, Liu X: Recyclable waste image recognition based on deep learning. Resources, Conservation and Recycling. 2021, 171:105636. 10.1016/j.resconrec.2021.105636
18. Zhang Q, Yang Q, Zhang X, Bao Q, Su J, Liu X: Waste image classification based on transfer learning and convolutional neural network. Waste Management. 2021, 135:150-157. 10.1016/j.wasman.2021.08.038
19. Gyawali D, Regmi A, Shakya A, Gautam A, Shrestha S: Comparative analysis of multiple deep CNN models for waste classification. 10.48550/arXiv.2004.02168
20. Azis FA, Suhaimi H, Abas E: Waste classification using convolutional neural network. ITCC '20: Proceedings of the 2020 2nd International Conference on Information Technology and Computer Communications. 2020, 9-13. 10.1145/3417473.3417474
21. Ziouzios D, Dasygenis M: A smart recycling bin for waste classification. 2019 Panhellenic Conference on Electronics & Telecommunications (PACET). 2019, 1-4. 10.1109/PACET48583.2019.8956270
22. Chu Y, Huang C, Xie X, Tan B, Kamal S, Xiong X: Multilayer hybrid deep-learning method for waste classification and recycling. Computational Intelligence and Neuroscience. 2018, 2018:1-9. 10.1155/2018/5060857
23. Xue W, Hu X, Wei Z, Mei X, Chen X, Xu Y: A fast and easy method for predicting agricultural waste compost maturity by image-based deep learning. Bioresource Technology. 2019, 290:121761. 10.1016/j.biortech.2019.121761
24. Aral RA, Keskin ŞR, Kaya M, Hacıömeroğlu M: Classification of TrashNet dataset based on deep learning models. 2018 IEEE International Conference on Big Data (Big Data). 2018, 2058-2062. 10.1109/BigData.2018.8622212
25. Nowakowski P, Pamula T: Application of deep learning object classifier to improve e-waste collection planning. Waste Management. 2020, 109:1-9. 10.1016/j.wasman.2020.04.041
26. Sousa J, Rebelo A, Cardoso JS: Automation of waste sorting with deep learning. 2019 XV Workshop de Visão Computacional (WVC). 2019, 43-48. 10.1109/WVC.2019.8876924
27. Adedeji O, Wang Z: Intelligent waste classification system using deep learning convolutional neural network. Procedia Manufacturing. 2019, 35:607-612. 10.1016/j.promfg.2019.05.086
28. Mazloumian A, Rosenthal M, Gelke H: Deep learning for classifying food waste. 10.48550/arXiv.2002.03786
29. Vyas C, Rathod K, Makvana K, Nimavat A, Kandoriya K: Smart waste sorting using deep learning and image augmentation. African Journal of Biomedical Research. 2024, 27:5777-5785. 10.53555/ajbr.v27i3s.2886
30. Caballero J, Vergara F, Miranda R, Serracín J: Inference of recyclable objects with convolutional neural networks. 10.48550/ARXIV.2104.00868
31. Yu Y, Grammenos R: Towards artificially intelligent recycling Improving image processing for waste classification. 10.48550/ARXIV.2108.06274
32. Shah J, Kamat S: A method for waste segregation using convolutional neural networks. 2022 Second International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT). 2022, 1-5. 10.1109/ICAECT54875.2022.9807969
33. Nasir I, Aziz Al-Talib GA: Waste classification using artificial intelligence techniques: Literature review. Technium Romanian Journal of Applied Sciences and Technology. 2023, 5:49-59. 10.47577/technium.v5i.8345
34. Ranjbar I, Ventikos Y, Arashpour M: Deep learning-based construction and demolition plastic waste classification by resin type using RGB images. Resources Conservation and Recycling. 2025, 212:107937. 10.1016/j.resconrec.2024.107937
35. Xie S, Wu H, Mao W, Chu X, Meng Y, Yang X: Study on efficient recognition and accurate localization method of waste plastic bottles based on deep learning. Ecological Informatics. 2025, 86:103020. 10.1016/j.ecoinf.2025.103020
36. Albawi S, Mohammed TA, Al-Zawi S: Understanding of a convolutional neural network. 2017 International Conference on Engineering and Technology (ICET). 2017, 1-5. 10.1109/ICEngTechnol.2017.8308186
37. O'Shea K, Nash R: An introduction to convolutional neural networks. 10.48550/arXiv.1511.08458
38. Kumar M, Husain M, Upreti N, Gupta D: Genetic algorithm: Review and application. International Journal of Information Technology and Knowledge Management. 2010, 2:451-454.

39. Kennedy J, Eberhart R: Particle swarm optimization. Proceedings of ICNN'95 - International Conference on Neural Networks. 1995, 4:1942-1948. 10.1109/ICNN.1995.488968
40. Vis JK: Particle Swarm Optimizer for Finding Robust Optima. LIACS, Holland; 2009.
41. https://en.wikipedia.org/wiki/Particle_swarm_optimization.
42. Garbage Classification (12 classes). (2021). Accessed: January 24: https://www.kaggle.com/datasets/mostafaabla/garbage-classification.