Waste Classification Using Fine-Tune ResNet50V2: A Deep Learning Approach

Ngo An, Nguyen Bao Han, Tran Hoang Nam, Nguyen Nguyen Phong

ABSTRACT— *Vietnam faces significant challenges in waste management, with rapidly increasing waste generation overwhelming the country's handling capacity. Currently, 70% of waste is disposed of in landfills, often failing to meet environmental standards. A substantial portion of this waste polluting Vietnam's waterways consists of single-use, low-value items such as plastic bags, food containers, and straws, contributing significantly to the country's plastic pollution. To combat this issue, the Vietnamese government has developed a National Action Plan for Management of Marine Plastic Litter by 2030, involving various organizations in campaigns and initiatives to reduce ocean plastic waste. This research proposes developing a real-time application to accurately identify and categorize various types of waste into predefined groups, using the Trashnet classification system. This system aims to streamline waste segregation, reduce manual labor, and enhance classification accuracy. By leveraging ResNet50, a deep convolutional neural network, the system processes a dataset of garbage images. ResNet50's deep architecture and residual learning capabilities allow it to learn complex image features, addressing challenges such as diverse waste appearances, varying lighting, and occlusions. This approach optimizes waste management practices, accelerates the segregation process, and ensures greater precision. Our results demonstrate the system's capability to accurately classify diverse waste images, significantly improving classification accuracy and promoting better resource management.*

Keywords— *Waste classification, Image classification, Deep learning, ResNet50,*

I. INTRODUCTION

Vietnam faces major concerns with waste management as waste generation is increasing rapidly. The country does not have the capacity to effectively handle this waste, with 70% disposed of in landfills where environmental standards are limited. A significant portion of the waste polluting Vietnam's waterways consists of single-use, low-value items like plastic bags, food containers, and straws. These contribute to a large part of the plastic pollution in the country. Effective waste management and recycling are crucial for minimizing environmental pollution and conserving natural resources. Proper recycling processes reduce the strain on landfills and decrease the need for raw material extraction, leading to sustainable resource use. One of the critical components of waste management is the accurate classification of trash, which facilitates recycling and reduces the strain on landfills. Advanced technologies in image classification, leveraging the power of deep learning, offer promising solutions to enhance the accuracy and efficiency of trash classification.

The task of trash classification involves distinguishing between various types of waste materials such as plastics, metals, paper, and glass. This process presents several challenges, including variations in lighting conditions, occlusions, and the diverse appearances of waste items. Traditional methods based on manual sorting are not only labor-intensive but also slow-going. Therefore, there is a pressing need for automated solutions that can accurately classify trash with minimal human intervention.

This report proposes the use of ResNet50, a state-of-the-art convolutional neural network (CNN) architecture, for the task of trash classification. ResNet50, known for its deep architecture and residual learning capabilities, is particularly effective in addressing the vanishing gradient problem, which is common in deep networks. By utilizing residual blocks, ResNet50 can learn intricate features and patterns from images, making it a robust solution for classifying diverse trash items. The network's ability to maintain high accuracy and efficiency even with complex data makes it an ideal candidate for this application.

The primary results of this study demonstrate that the ResNet50-based model significantly improves the accuracy of trash classification compared to traditional methods. The model achieves high precision and recall rates, effectively distinguishing between different types of waste materials. This not only streamlines the recycling process but also contributes to better resource management and environmental sustainability. Additionally, the study explores novel techniques in data augmentation and transfer learning to further enhance the model's performance.

The report is structured as follows: Section 2 reviews related works in image classification. Section 3 is the Method, which details the ResNet50v2 architecture and its modifications. Section 4 discusses the Experiments. Following is the Conclusion, including the results and analysis as well as the summary of findings.

II. RELATED WORKS

Image classification has attracted considerable attention due to its potential to improve waste management. Importance of the waste classification process can lead to reduced environmental impacts. CNN (Convolution Neural Network) used to become the most standard approach when working with classification tasks

A. Modern feature extract base on tradition extract methods

Feature extraction is the core problem of object recognition. It mainly focuses on low-level feature extraction, such as texture, edges, corners, and colors [1]. One of the most classic methods for extracting local texture features is the Local Binary Patterns (LBP) descriptor[2] . The LBP method converts texture into a binary vector by thresholding the neighborhood of each pixel with the center pixel and treating the result as a binary number. This technique is computationally efficient and invariant to rotation and grayscale variations. But, LBP still focuses on extracting physical features like texture without containing much information about high-level object structure.

SIFT can overcome such weaknesses in LBP and has certain advantages in terms of robustness to noise and occlusion. It has been used to represent images in various scenes. Histograms of gradient directions, which provides stability against noise and occlusion. However, the complex calculations for keypoint detection and descriptor generation make the computation expensive and time-consuming. Additionally, the high memory usage makes it difficult to handle large datasets.

The Histogram of Oriented Gradients (HOG) method focuses on edge feature extraction by computing the gradients of the image. HOG extracts features based on the histogram of gradient directions within localized portions of an image, which helps in detecting objects based on their shape and contour. While HOG is robust to variations in geometric and optical deformations, it can be sensitive to noise and may not capture fine-grained details or subtle variations in object appearance.

Although these traditional methods are still effective, it still needs to import some aspects. On one hand, such traditional algorithms do not work well in complex scenarios. On the other hand, it only extracts the low level semantic features of images and ignores the high level semantic features, which leads to the loss of several problem-specific features [3]

B. Related Work

Image classification is continuously diversifying due to the rapid growth of artificial intelligence, with the prominent models being AlexNet [9], VGG [10], Inception [11], and ResNet [12]. The researchers ran multiple trials with these models and attempted to make improvements in order to acquire better outcomes. Garbage classification, being a relatively young subject, lacked a consistent dataset for neural network learning in its early stages.
Shanshan Meng and Wei-Ta Chu describe their result on the common Computer Vision (Shanshan Meng and Wei-Ta Chu 2020) such as HOG+SVM which their highest accuracy (47,25%), Simple CNN(93,75% which 40 epochs), HOG + CNN (93,56% which 40 epochs)... Which ResNet50 make highest accuracy (95.35%)

In others hand, RahMi Arda Aral et al. publish their paper (Aral et al. 2018) such as Densenet121, Densenet169, InceptionResnetV2 (Szegedy et al. 2017) MobileNet, Xception (Chollet 2017) show their result with efficient approach. The best result were found in Densenet121 with fine-tuning technique (95%)

Selim Sürücü1* , İrem Nur Ecemiş2 used ResNet50V2 and has positive accuracy (97,07%). Rismiyati et al. on (Rismiyati et al. 2020) using XceptionNet have 88% accuracy

In addressing the limitations of existing classification models, Xiaoxuan Ma et al [3] propose an augmented trash classification model derived from modifications to the ResNet-50 architecture. The enhancement comprises two primary modifications: firstly, the integration of an attention module within the residual block to refine input feature filtering, coupled with an altered downsampling method to mitigate information loss; secondly, the implementation of horizontal and vertical multi-scale feature fusion within the core network structure to optimize feature utilization. These adjustments facilitate improved filtering and reuse of image features, culminating in a model that not only surpasses the original ResNet-50 by 7.62% on the TrashNet dataset but also exhibits increased robustness, particularly for small datasets with limited samples. The empirical evidence underscores the model's superior classification efficacy.

These advancements underscore ongoing progress in refining image classification models, especially for waste categorization, by leveraging novel architectures and optimization strategies.

III. METHOD

A. Dataset

The dataset used for this study was taken from the Kaggle website[7] by Farzad Nekouei , which is an open-source dataset. These dataset include 2527 images distributed among six distinct categories [7] -Metal, Glass, Paper, Trash, Cardboard and Plastic. Number of each class distribution are: Metal: 410, Glass: 501, Paper: 594, Trash: 137, Cardboard: 403, Plastic: 482 All image dimensions are 512 x 384 pixel and in JPEG format

These datasets show a noticeable imbalance in the distribution of image categories: ‘Trash’ is less numerous than other categories and ‘Paper’ has the most numerous. This Imbalance problem could make biased models predicting the more frequent categories is ‘Paper’. It also makes it hard to predict ‘Trash’ categories because of less frequency. In the real-world scenarios where ‘Trash’ items is the common type of ‘Paper’ items make models not work so well. . It continues to demonstrate to us that accuracy is high but poor in the "Paper" type due to the observed number.

B. Addressing Class Imbalance

$$w_j = \frac{n}{k \times n_j}$$

Fig.1: Formula of ‘Balanced’ class weight
k: number of classes
n: total number of sample
nj: number of sample in class j

To address class imbalance, the Scikit-learn library's "balanced" heuristic derives its core formula from the idea of calculating class weight. This is frequently used in machine learning to give various classes the proper weights based on the dataset's frequency (Fig 1).

$$\text{Loss} = -\sum_{j=1}^{C} y_{ij} \log(p_{ij})$$

Fig.2 Cross-Entropy Loss function

$$\text{Loss} = -\sum_{j=1}^{C} w_j \cdot y_{ij} \log(p_{ij})$$

Fig 2.1: Cross-Entropy Loss Function with Class Weight

where: C is the number of classes, yij is the true label, pij is the predicted probability of the sample i belong to class j, wj is the weight assigned to class j (fig 2,2.1). This adjustment to the loss function helps make learning more optimized and efficient by addressing the class imbalance. Classes with more samples (majority classes) are given lower weights, whereas classes with fewer samples (minority classes) are given higher weights according to this method. Consequently, the model gains the ability to assign greater weight to minority groups, thereby enhancing its overall robustness and performance.

C. Fine-Tune ResNet50V2

a) ResNet50V2

ResNet50V2 architecture was chosen for this study due to its proven effectiveness in reducing training error and achieving high accuracy. It has consistently achieved state-of-the-art results in many benchmark datasets. And it is readily available in the popular learning environment Tensorflow.

ResNet50 is made up of 50 layers that are separated into 5 blocks, each of which has a collection of "residual blocks.". This residual block enables the network to develop more accurate representations of the incoming data by allowing preventative information from earlier layers.

Deep Convolutional Neural Networks start the era of classification problems. Deep networks naturally integrate low/mid/high level features [9] and classifiers in an end-to-end multilayer fashion, and the “levels” of features can be enriched by the number of stacked layers. When stacking more layers into the model, a degradation problem has been exposed: with the network depth increasing, accuracy gets saturated (which might be unsurprising) and then degrades rapidly[8]. Its issues don't come from overfitting, adding more layers to suitably deep models leads to higher training-error[8].

$$\mathbf{y}_l = h(\mathbf{x}_l) + \mathcal{F}(\mathbf{x}_l, \mathcal{W}_l) \quad (1)$$

$$\mathbf{x}_{l+1} = f(\mathbf{y}_l) \quad (2)$$

Fig.4: Residual Block V1

$$\mathbf{x}_L = \mathbf{x}_l + \sum_{i=l}^{L-1} \mathcal{F}(\mathbf{x}_i, \mathcal{W}_i)$$

Fig.5 : Output recursive

where x_l: Input feature to the residual block, h(x_l): identity mapping of h(x_l) = x_l ,F(x_l, W_l): is a residual function, W_l is a set of weights (and bias l-th), f(y_l) the activation function (ReLU), y_l: Output

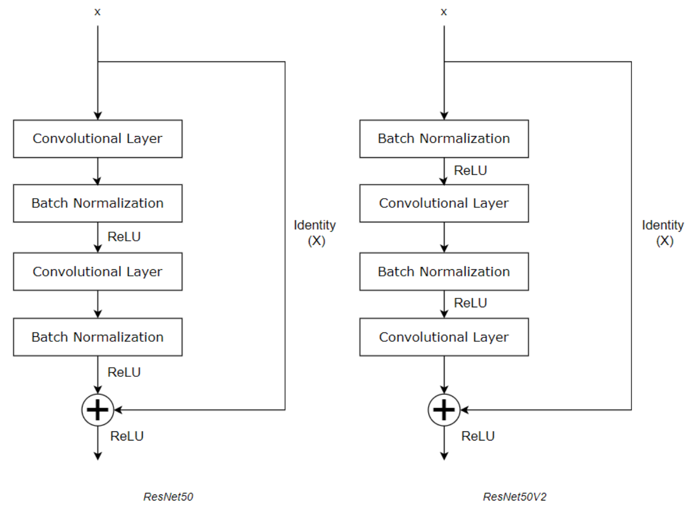


Fig 6. Difference between ResNet50 and Resnet50V2

Key difference between ResNet50 and ResNet50V2 is: ‘pre-activation’. This means ReLU activation is applied before the convolution operation. This means that normalization and activation occur prior to convolution within each residual block[8]. It demonstrates that no matter how deep or shallow a layer is within the network, we can think of the features learned at a deeper layer as the combination of features learned at any shallower layer, along with some additional details captured by what we call the "residual function". This setup makes it easier for the network to learn complex patterns by focusing on the differences between features at different layers[10]

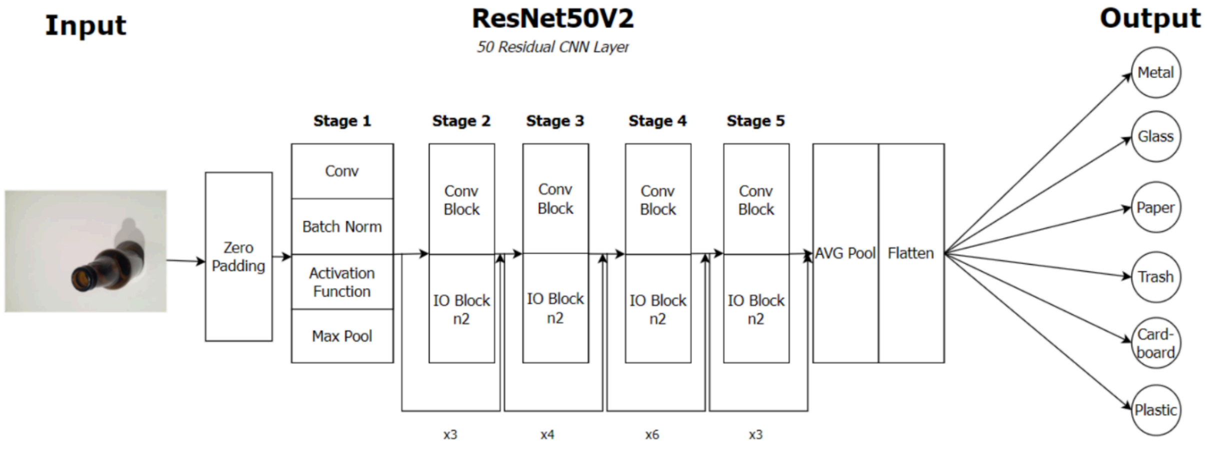


Fig.7: ResNet50V2 Flow

ResNet50v2 has several advantages and disadvantages. It has been proven to reduce training error while achieving high accuracy[10] . Specifically, it reduces training time and effectively handles the vanishing gradient problem, allowing for deeper and more robust networks.[8] .However, the disadvantages are significant. It requires intensive computational requirements[14]

b) Fine-Tune

Fine-tuning technique to improve efficient performance of pre-trained convolutional models, now is ResNet50V2. Fine-tuning works by model compatible with unexplored dataset[23]. First, ResNet50V2 initially trained on a comprehensive dataset, using pre-existing knowledge for efficient learning. The methodology comprises both pre-training and fine-tuning.

First, initially models like ResNet50V2 were pre-trained in a large dataset (ImageNet). ImageNet containing over a million images with a thousand classes, provided a rich dataset feature that helps for visual recognition tasks.

Second, Fine-tuning by adjusting the number of layer freezing or hyper-parameters to make a model suit with a new dataset while existing weight is still retained. Selection of new hyper-parameters such as the number of layers will be frozen.. in order to avoid overfitting on the smaller dataset while yet allowing the model to adapt to the new objective. [24]

IV. EXPERIMENTS

A. Experiment setup.

1. Objective

The goal of this experiment is to understand and evaluate the performance of the ResNet50V2 in garbage classification tasks.

2. Hardware and Software

Hardware: CPU Kaggle Processor, Ram: 29GB, GPU Tesla P100 and GPU Memory (VRAM): 16GB
Software: Python version: 3.10.13 , TensorFlow version: 2.15.0, NumPy version: 1.26.4, scikit-learn version: 1.2.2

B. Procedure

1. Data Preparation

The dataset used for this experiment is Kaggle website[7] by Farzad Nekouei. The directory of the folder is stored in the variable name 'ROOT_PATH'. To facilitate training and validation, the dataset was split into training and validation is 80/20

Fig.8 : Example of dataset

2. Image Augmentation

The experiment with ResNet50V2 employed the best result, we applied Data Augmentation. This technique was implemented by the 'ImageDataGenerator' class from tensorflow keras library. . This class allows for real-time data augmentation during the training process, helping decrease extracting before the training process[16]. In a fine-tuning, we unfroze the last 38 layers of ResNet50V2 to retain the learned features from ImageNet. Our technique's we use is:

- *Rescale = 1./255*: to help faster convergence by convert in common scale during training[14]
- *Rotation Range= 45*, Width Shift Range and Height Shift Range= 0.15, *Zoom Range= 0.15*, Horizontal and Vertical Flip= True
- *Fill mode= nearest*: handle wrong pixel that are affected by transformation by fill nearest pixel into, preserving the overall structure[15]
- *Chanel shift range= 10*: random shifted value in color channels of image from -10 to +10, which can improve model's ability to handle different color distributions in real-world scenarios[11]
- *Brightness range= from 0.9 to 1.1*: random adjust brightness in range [0.9, 1.1]. This help model become more invariant to change of lighting condition
- *Shear Range= 0.05:* shear transformation shape of object in the image allow to 5% horizontally and vertically[1]

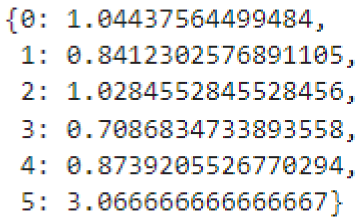
3. Model Architecture

With the top-layer removed, we add ResNet50V2 (size=384,384) into the top layers. This allows us to add some specific classification tasks. The size=(384,384) through it higher computation time and higher memory used, but it returns higher detail. We also tried with size= (224,224) but accuracy is not good. We also add some in end, Included:

- **GlobalAveragePolling2D**: average over all spatial locations to reduce the input tensor's (two-dimensional spatial data) spatial dimensions to a single vector.[18]
- **Batch Normalization Layer**: stabilized and accelerated training process by normalizing the input activation[19]
- **Dense:** is a fully connected layer where each neuron is connected to every neural in the preceding layer. It performs a linear operation on the input data, followed by an activation function (in this case, ReLU activation)[20]

- **Dropout Layer**: is a regularization technique to prevent overfitting by random setting value to zero during training. A dropout rate is 0.5, meaning 50% chance of being dropped out during each training iteration. This helps the network learn more robust features.[21]
- **Softmax Output Layer**: Classified the input images into the predefined categories. The sum of the probabilities all classes equal to 1.0. It uses a probability distribution over the classes—each value denoting the likelihood that the input belongs to that class—to classify the input photos into predetermined categories [22].

4. Address Class Imbalance



```
{0: 1.044375644994844,
 1: 0.8412302576891105,
 2: 1.0284552845528456,
 3: 0.7086834733893558,
 4: 0.8739205526770294,
 5: 3.066666666666667}
```

Fig.9 : Result after apply Sklearn class weight
0: 'cardboard', 1: 'glass', 2: 'metal',
3: 'paper', 4: 'plastic', 5: 'trash'

The 'scikit-learn class-weight' library is used in Address Class Imbalance to calculate class weight. During training, classes are given varying weights in other classes to penalize misclassification in the minority classes more heavily. Because of time, we don't have the opportunity to discover others to handle imbalance classes.

5. Training Process

The model used was the 'Adam' Optimizer with a learning rate of 0.001. The loss function is categorical cross entropy. The metric for evaluation includes accuracy, precision, and recall. The training also used 'ReduceLROnPlateau' adjust learning rate base on validation accuracy, 'EarlyStopping' stop training when validation accuracy not improving, 'ModelCheckpoint' to save best-perform model weight, 'Tensorboard' visualizes training metrics for monitoring.

C. Results

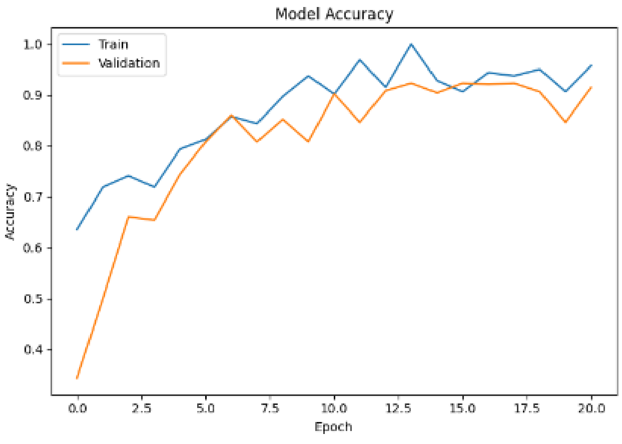


Fig.10: Visualize Model Accuracy

From Epoch 4 to 18, the training accuracy consistently rises while the validation accuracy drops sharply, which is a clear indication of overfitting. During this period, the model becomes increasingly adept at learning the specific patterns and noise within the training data, but it fails to generalize these patterns to the validation data. On average, the training accuracy is significantly higher than the validation accuracy, with training accuracy around 90% and validation accuracy around 80%. This 10% gap further confirms that the model is overfitting, as it performs well on the training data but poorly on unseen data [11]. Additionally, the accuracy graph fluctuates significantly, making it difficult to identify an optimal epoch for training. These fluctuations suggest instability in the training process, which may be due to an overly complex model or inadequate regularization techniques.

The loss graph reveals several critical insights about the model's training process. On average, the training loss is significantly lower than the validation loss, which is a strong indication of overfitting. The model is learning the training data well but fails to generalize to unseen data, leading to higher validation loss [12]. Notably, at epochs 12 and 43, there are dramatic spikes in the validation loss, rising sharply to approximately 1.55 and 1.7, respectively. These spikes suggest moments where the model's performance on the validation set degrades suddenly, possibly due to the model overfitting to certain batches of the training data or fluctuations in the learning process [12]. The overall fluctuation in both losses and these sharp increases in validation loss highlight the need for strategies to stabilize the training process and improve the model's generalization capabilities.

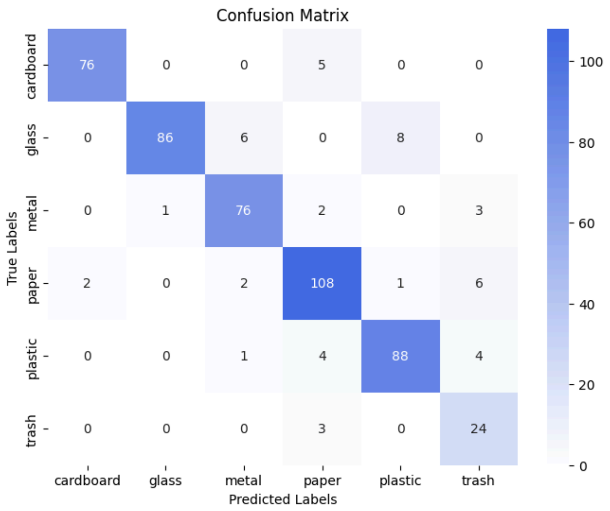


Fig.11: Confusion Matrix

The model's accuracy of 90.51% on validation predictions suggests a relatively high overall performance. However, a deeper analysis of the confusion matrix reveals that certain materials, specifically metal, paper and trash are more likely to be predicted incorrectly. This observation indicates potential areas of weakness in the model's classification abilities, particularly when distinguishing between these materials. Misclassifications in these categories could have significant implications, especially in recycling processes where precise sorting is crucial for effective resource management and environmental sustainability.

V. CONCLUSIONS

In this study, we demonstrated the efficacy of the ResNet50V2 model in classifying synthetic garbage images from the dataset by Farzad Nekouei. While our findings contribute to the understanding of deep learning applications in waste management, the reliance on simulated data limits the generalizability of our results. The pursuit of real-world food waste imagery stands as the paramount direction for future research, promising significant advancements in the accuracy and applicability of classification models. As we move forward, the exploration of data augmentation techniques remains a valuable inquiry to enhance model performance under diverse conditions. Ultimately, the integration of robust machine learning models like ResNet50V2 into waste management systems holds the potential to revolutionize our approach to environmental sustainability.

VI. REFERENCES

[1] Wu, Zhize & Li, Huanyi & Wang, Xiaofeng & Wu, Zijun & Zou, Le & Xu, Lixiang & Tan, Ming. (2022). New Benchmark for Household Garbage Image Recognition. Tsinghua Science and Technology. 27. 793-803. 10.26599/TST.2021.9010072.

[2] T. Ojala, M. Pietikainen and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 7, pp. 971-987, July 2002, doi: 10.1109/TPAMI.2002.1017623.

[3] Ma, X.; Li, Z.; Zhang, L. (2022). An Improved ResNet-50 for Garbage Image Classification

[4] S. Meng and W. -T. Chu, "A Study of Garbage Classification with Convolutional Neural Networks," 2020 Indo – Taiwan 2nd International Conference on Computing, Analytics and Networks (Indo-Taiwan ICAN), Rajpura, India, 2020, pp. 152-157, doi: 10.1109/Indo-TaiwanICAN48429.2020.9181311.

[5] R. A. Aral, Ş. R. Keskin, M. Kaya and M. Hacıömeroğlu, "Classification of TrashNet Dataset Based on Deep Learning Models," 2018 IEEE International Conference on Big Data (Big Data), Seattle, WA, USA, 2018, pp. 2058-2062, doi: 10.1109/BigData.2018.8622212.

[6] Rismiyati, S. N. Endah, Khadijah and I. N. Shiddiq, "Xception Architecture Transfer Learning for Garbage Classification," 2020 4th International Conference on Informatics and Computational Sciences (ICICoS), Semarang, Indonesia, 2020, pp. 1-4, doi: 10.1109/ICICoS51170.2020.9299017.

[7] https://www.kaggle.com/datasets/farzadnekouei/trash-type-image-dataset

[8] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. (2016)

[9] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional neural networks. In ECCV, 2014.

[10] He, K., Zhang, X., Ren, S., Sun, J. (2016). Identity Mappings in Deep Residual Networks. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds) Computer Vision – ECCV 2016. ECCV 2016. Lecture Notes in Computer Science(), vol 9908. Springer, Cham. https://doi.org/10.1007/978-3-319-46493-0_3

[11] Hawkins, D.M., 2004. The problem of overfitting. Journal of chemical information and computer sciences, 44(1), pp.1-12

[12] Klugman, S.A., Panjer, H.H. and Willmot, G.E., 2012. Loss models: from data to decisions (Vol. 715). John Wiley & Sons

[13] A. Rastogi, "ResNet50," Medium, Mar. 14, 2022. https://blog.devgenius. io/resnet50-6b42934db431 (accessed Jul. 31, 2023)

[14] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., ... & Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision, 115(3), 211-252.

[15] Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on Image Data Augmentation for Deep Learning. Journal of Big Data, 6, 60

[16] Perez, L., & Wang, J. (2017). The Effectiveness of Data Augmentation in Image Classification using Deep Learning. arXiv preprint arXiv:1712.04621.

[17] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. Advances in Neural Information Processing Systems (NIPS).

[18] Lin, Min, Qiang Chen, and Shuicheng Yan. "Network In Network." arXiv preprint arXiv:1312.4400 (2013).

[19]Ioffe, Sergey, and Christian Szegedy. "Batch normalization: Accelerating deep network training by reducing internal covariate shift." International Conference on Machine Learning. 2015.

[20] Goodfellow, Ian, et al. "Deep Learning." MIT Press, 2016.

[21] Srivastava, Nitish, et al. "Dropout: A simple way to prevent neural networks from overfitting." The Journal of Machine Learning Research 15.1 (2014): 1929-1958.

[22]Bishop, Christopher M. "Pattern Recognition and Machine Learning." Springer, 2006.

[23] Wang, Yulong & Haoxin, Zhang & Zhang, Guangwei. (2019). cPSO-CNN: An efficient PSO-based algorithm for fine-tuning hyper-parameters of convolutional neural networks. Swarm and Evolutionary Computation. 49. 114-123. 10.1016/j.swevo.2019.06.002.

[24]Sarasaen, Chompunuch et al. "Fine-tuning deep learning model parameters for improved super-resolution of dynamic MRI with prior-knowledge." Artificial intelligence in medicine 121 (2021): 102196 .