2023 9th International Conference on Control, Decision and Information Technologies (CoDIT) | 979-8-3503-1140-2/23/$31.00 ©2023 IEEE | DOI: 10.1109/CODIT58514.2023.10284298

Garbage classification based on fine-tuned state-of-the-art models

Ramil Shukurov

Abstract — **The world faces one of the crucial issues - waste management, which needs to be tackled urgently to minimize environmental pollution. The main challenge is to classify the massive amount of waste reliably and accurately. Many existing deep learning models employed to achieve accurate results still require to be improved as their performance varies on a different dataset. This research proposes an algorithm based on fine-tuned convolutional neural networks (GoogleNet, ResNet, DenseNet, ResNeXt, EfficientNet) using two distinct optimization algorithms (SGD momentum, Adam) to classify an open-source dataset into twelve classes. Experimental results reveal that the fine-tuned ResNeXt neural network model achieves high model accuracy, 95 %, with a small number of epochs. The experiment indicates that our achievement outperforms several counterpart methods.**

***Index Terms*-garbage, deep learning, multi-label classification, environment**

I. Introduction

Waste generation is soaring at a shocking rate with the acceleration of economic growth, and their disposal is becoming a core issue all over the world. At this stage, China, one of the most populated countries globally, adds approximately 1.1 billion tons of domestic waste annually [1], while the United States produces more than 600,000 metric tons of solid waste [2]. Another most populated country, India, is believed to produce ten million tons of waste only in metropolitan cities [3]. According to the world bank, annual global waste production is about four billion tons of waste [4]. The figure is expected to increase significantly in the upcoming 25 years.

Disposing of garbage timely and effectively may reduce environmental pollution that strongly impedes the sustainable development of a country and precariously endangers individuals' health [5]. Landfilling, incineration, and recycling are distinct management methods to deal with various categories of waste**.** Especially in China, incineration and landfilling are common methods to process garbage, but they have severe drawbacks. For instance, incineration may lead to air pollution and imperil soil resources, while landfilling may contaminate air and surface water [6]. In this instance, recycling appears to be the most effective way of dealing with contamination, even though only 10 % of massive solid waste products can be recycled [7]. A fundamental step for recycling, waste classification, should be a mandatory measurement to protect land, and water resources, and minimize air contamination. Therefore, countries should shift from voluntary to compulsory to make garbage classification, as China implemented this action in 2019 [8]. The traditional way of classifying waste is a manual selection which is inefficient and a waste of time, whereas automatic garbage classification is more powerful and less hazardous to human health and the natural environment.

The research objective is to enhance the accuracy of garbage classification using five well-known convolutional neural networks. To achieve this objective, comparative experiments were conducted on the application, leading to the following contributions:

- This research compares the performance of traditional deep-learning models: GoogleNet, ResNet50, ResNeXt50, DenseNet121, and EfficientNetB0 under a different number of epochs and optimization algorithms.
- The performance of five models is assessed using performance metrics: accuracy, precision, recall, and F1-score.
- This research helps to distinguish which model could yield better waste classification using the fine-tuned technique.

II. Related Work

Comprehensive research has been conducted using Machine Learning (ML) and Deep learning (DL) techniques in the classification of waste, but the primary factor is the precise classification to reduce the operating cost of waste treatment. A proper selection of DL model parameters and their optimization method plays a significant role in the prediction accuracy [9]. Many researchers have used these techniques to classify waste into several categories. They have deployed pre-trained models to get the best result of multi-label classification. In this section, we review related work in this field.

Zian et al. expanded an open-source dataset of more than 2,800 images that Gary Thung and Mindy Yang created to classify pictures into seven categories by scrapping various e-waste images over the internet [10]. They preprocessed the dataset to avoid overfitting deep learning models. Using the transfer learning approach, Xception, DenseNet121, Resnet-50, MobilenetV2, and EffiecienNetB7 models were applied to categorize images. According to the experimental results, they achieved the best accuracy by employing the DenseNet model with 121 layers and obtained an accuracy rate of 93.3 %.

Ramil Shukurov is with the Process Automation Engineering Department, Baku Higher Oil School, Baku, Azerbaijan (e-mail: ramil.shukurov@bhos.edu.az).

Mobilenet demonstrated an incredible performance following the DenseNet121 model with slightly less accuracy, 93 %. The other three models: namely Resnet50, Xception, and EfficientNetB7, achieved 92 %, 92.5 %, and 87 % accuracy.

Chinassylov et al. used a dataset containing approximately 22.5k images to categorize them into two classes, organic and recyclable waste [11]. They used the PCA algorithm in training to reduce the size of features and speed up the SVM classifier execution time. The experiment resulted in an accuracy of 88 %, 86 % on the test, and a cross-validation set accordingly.

Siva Kumar and others applied the VGG16 model to classify kitchen waste into organic, non-recyclable, and recyclable waste [12]. Experimental results showed that the accuracy of the model reached 84 % over an imbalanced dataset.

Yuke Chen et al. conducted research using the Inception V3 deep learning model in urban waste classification [6]. They used an open dataset from Kaggle that includes twelve labels and applied data augmentation techniques, such as random flip, random rotation, and random zoom, to preprocess input images to obtain better performance. Image classification is executed by the combination of five fully connected layers and two batch normalization layers after extracting core features using a pre-trained Inception V3 model. Freezing all layers of the pre-trained model and training the model resulted in high classification accuracy of 93 %.

Li Cao and Wei Xiang also applied Inception V3 for classifying images into four labels: recyclable waste, household food waste, hazardous waste, and residual waste [13]. Training of the model is conducted using varying learning rate factor that is achieved by the exponential decay method with the Adam optimization algorithm. As a result of the experiment, test accuracy accounted for 93.2 % after more than 2,000 iterations of the training process.

Sarah Frost et al. enlarged the dataset that G. Thung and M. Yang collected for TrashNet, by adding images of compostable content and named it ComposNet [2]. The ComposNet dataset includes seven labels, unlike TrashNet consisting of five classes of recyclable content and trash. The application of pre-trained MobileNet architecture along with four layers trained on the ComposNet dataset resulted in a model accuracy of 77.3 %, slightly higher than Thung and Yang's achievement.

Nasim Shah et al. also referred to G. Thung and M. Yang's dataset and increased the number of images in this dataset to 4,000 by collecting them from various sources [14]. They implemented Yolo v3 to classify the images into four categories labeled glass, paper, metal, and plastic, with a model accuracy of 85%.

Sidharth R et al. designed a ConvNet model to classify the TrashNet dataset into four classes [3]. The test accuracy reached about 76 % by training the model with 100 epochs.

Olugboja Adedeji and Zenghui applied a pre-trained ResNet50 model to extract features from images from Wang G. Thung and M. Yang's dataset. They applied Support Vector Machine (SVM) to classify objects into four classes and obtained a model accuracy of 87 % [4].

Wang Hao successfully classified objects into four classes using VGG16 architecture composed with ReLU activation function and BN layer with an accuracy rate of 75.6 % [1].

Rismiyati et al. made a thorough comparative analysis of three different deep-learning models in the classification of TrashNet images into six categories [15]. Xception, ResNet50, and VGG16 models were trained using two various numbers of epochs. As a result, the highest model accuracy rate of 88 % was achieved by Xception architecture, whereas VGG16 and ResNet50 models showed slightly less performance with 85% and 84% accuracy rates accordingly. Increasing the number of epochs did not affect model accuracy, except the ResNet50 model showed a slight rise.

Ashik Mohammed Sal et al. also applied a pre-trained Xception model to differentiate images into five categories [16]. They initially modified the FC layer of the model and trained it with a few epochs, followed by a fine-tuning method by freezing some parts of the model. Test accuracy accounted for 92 % on an open-source dataset.

Yujie He et al. focused on CNN models with two different classifiers [8]. They experimented with ResNet50, VGG11, and AlexNet deep learning architectures on the TrashNet dataset and concluded that the AlexNet model performed better than others. They explored the AlexNet model with diverse options, including dropout, learning rate decay, and data augmentation, and evaluated with two distinct classifiers, Softmax and SVM. As a result of the experiment, the highest accuracy they obtained was about 80 % with the SVM classifier among all options. Their research was based on training the dataset from scratch rather than using the transfer learning technique.

III. MATERIALS AND METHODS

We referred to an open-source dataset on Kaggle created by Mostafa Mohamed [17]. The dataset contains more than 15,000 images in *.jpeg* format associated with twelve classes: battery, biological, brown, green, and white glass, clothes, metal, paper, cardboard, biological images, plastic, shoes, and trash. We split the dataset into three sets for training, validating, and testing to obtain an accurate result. Training, validation, and testing set account for 80, 10, and 10 percent of all pictures in the dataset accordingly. Our experiment was carried out specifically using Python programming language with the Pytorch framework. We diminished the number of images from all categories since we lack hardware resources, and Fig. 1 shows the latest picture quantity for three sets.

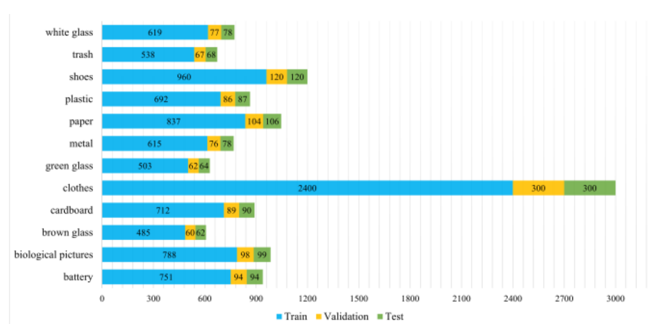

Figure 1. Number of images

Images in the dataset have conflicting sizes, and we performed pre-processing to make all images in unique heights and widths for fitting into state-of-the-art models, so all images in the training, validation, and testing set are reshaped to a size of 200x200 in *.jpeg* format.

In this research, we applied specific deep-learning models to classify the dataset into twelve labels. GoogleNet, ResNet50, ResNeXt50, DenseNet121, and EfficientNet_B0 pre-trained neural networks were employed by re-training them using two distinct optimization algorithms. GoogleNet uses the inception structure that extracts features from a variety of scales on the picture through several convolutional kernels and concatenates them to achieve a better representation of the picture at the end [18]. The initial structure increases the depth and width of the network, and at the end of the architecture, the fully connected layer is replaced with the average pooling layer to eliminate the vanishing of the gradient. ResNet model, however, introduces residual networks that contain some skip connections followed by batch normalization. ResNeXt model is a complex version of the ResNet model in which consecutive layers in each block are replaced with a set of branches of parallel layers [19]. Inspired by the basic idea of the ResNet model, DenseNet neural network model establishes dense connections between all the previous layers and subsequent layers to achieve better performance with low computation cost and fewer parameters [18]. The EfficientNet model combines several scaling methods under the compound scaling method: network width, depth, and image resolution to make the model balance the accuracy and speed [18, 19].

Stochastic Gradient Descent (SGD) with momentum, shortly SGD momentum, and Adam optimization algorithms are used for tuning deep learning architectures. The steepest-descent method is the most common method for parameter learning and can sometimes behave unexpectedly when finite step size is considered, it does not always point in the best direction to update parameters [20]. In specific cases, minor differences in sensitivity to various features can cause oscillations that reduce effective step size in the correct direction. Therefore, momentum-based techniques are aimed to smooth oscillations in the last few steps to move smoothly in the right direction, given in (1) and (2).

$$\overline{\mathrm{V}} \Leftarrow \beta\overline{\mathrm{V}} - \alpha\frac{\partial \mathrm{L}}{\partial \overline{\mathrm{W}}} \tag{1}$$

$$\overline{\mathrm{W}} \Leftarrow \overline{\mathrm{W}} + \overline{\mathrm{V}} \tag{2}$$

where $\beta \in (0,1)$ - smoothing parameter, α – learning rate, $\bar{V}$ - velocity, $\bar{W}$ – parameter vector, L - loss function (defined over mini-batch of instances)

Though the momentum-based approach accelerates the learning process, it often causes a slight overshoot in the direction where velocity is picked up [19].

Adam optimizer incorporates most of the advantages of other optimization algorithms. It uses a similar idea to AdaGrad and RMSProp. For example, it updates the value as RMSProp, shown in (3).

$$\mathrm{A_i} \Leftarrow \rho\mathrm{A_i} + (1-\rho)\left(\frac{\partial \mathrm{L}}{\partial \mathrm{w_i}}\right),\ \forall \mathrm{i} \tag{3}$$

where A_i - the exponentially averaged value of an *i*th parameter w_i, ρ – decay parameter

At the same time, it maintains an exponentially smoothed value of the first-order gradient given below,

$$\mathrm{F_i} \Leftarrow \rho_f\mathrm{F_i} + (1-\rho_f)\left(\frac{\partial \mathrm{L}}{\partial \mathrm{w_i}}\right),\ \forall \mathrm{i} \tag{4}$$

where α_t – learning rate in the *t*th iteration.

We evaluated the performance of the five neural network models both graphically and numerically. Numerical evaluation includes four metric types based on values: TP, TN, FP, FN.

Accuracy, shown in (6), is a ratio of the number of correct predictions to the number of total predictions.

$$\mathrm{Accuracy} = \frac{\mathrm{TP+TN}}{\mathrm{FP+FN+TP+TN}} \tag{6}$$

Precision is a ratio of the number of correct positive predictions to the number of total positive predictions given in (7).

$$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{FP+TP}} \tag{7}$$

A recall is a ratio of the number of correct positive predictions to the number of all positive observations given in (8).

$$\mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP+FN}} \tag{8}$$

F1-score combines both precision and recall and takes FP and FN into account. In case of having an uneven class distribution, it is effective to compute this performance measurement. Equation (9) represents how f1-score is calculated.

$$\mathrm{F1-score} = 2 * \frac{\mathrm{Precision*Recall}}{\mathrm{Precision+Recall}} \tag{9}$$

IV. RESULTS AND DISCUSSION

Initially, we modified fully connected layers of GoogleNet, ResNet50, DenseNet121, EfficientNet_B0, and ResNeXt50 neural networks: added dropout to avoid overfitting problems and changed the number of neurons in the output layer, then separately trained them. The training process of each model is accomplished separately using SGD momentum followed by Adam optimizer. Model accuracy barely reached 90 % once freezing all layers in a model except the fully connected layer. Hence, we unfreeze all layers to obtain better model accuracy during the training process.

In this section, we compare and analyze the results of each model following research objectives. Fig. 2 separately examines the change in training accuracy and validation accuracy over the number of epochs in the case of training the model with the SGD momentum optimizer.

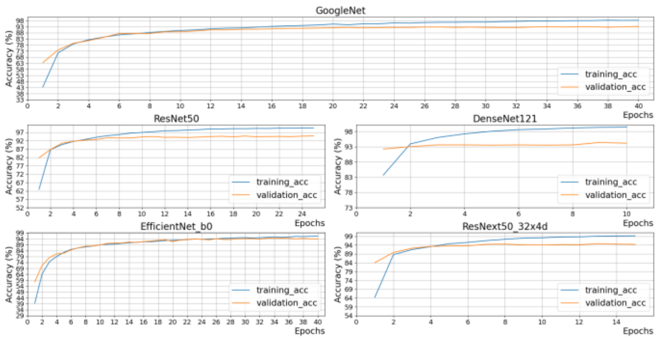

Figure 2. Change in accuracy with SGD Momentum optimizer.

It is obvious that each model was trained with a different number of epochs. The least number of epochs corresponds to the model of DenseNet121, whereas GoogleNet and EfficientNet architectures require 40 epochs for fine-tuning. Training the model with ResNeXt50 and ResNet50 utilized 15 and 25 epochs, respectively. The validation accuracy of all pre-trained models reached about 95 %, except GoogleNet, which stayed at around 93 %, while training accuracy is more than 95 % in all cases. Fig. 3 compares the change in training loss with the change in validation loss over the number of epochs.

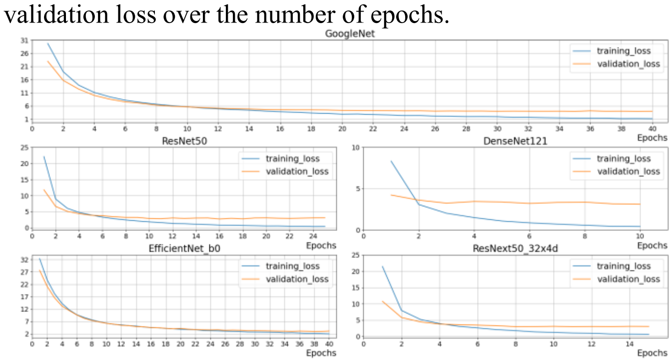

Figure 3. Change in loss with SGD Momentum optimizer.

We can state that there is barely overfitting as the gap between training and validation loss is not widening. The identical experiment we carried out is training the model with the Adam optimizer to observe the difference from the performance of the previous optimizer. In Fig. 4, we plot the change in training and validation accuracy over the epochs in the case of training the model with the Adam optimizer. The noticeable difference between Fig. 4 and Fig. 2 is the number of epochs. Training the model with the Adam optimizer used a smaller number of epochs with similar validation accuracy in comparison with the model trained with SGD momentum, except for the DenseNet121 where it remained stable. EfficientNet and GoogleNet still use more epochs to reach 95 % validation accuracy. On the contrary, it took just 6 epochs to train both ResNet and ResNeXt neural networks.

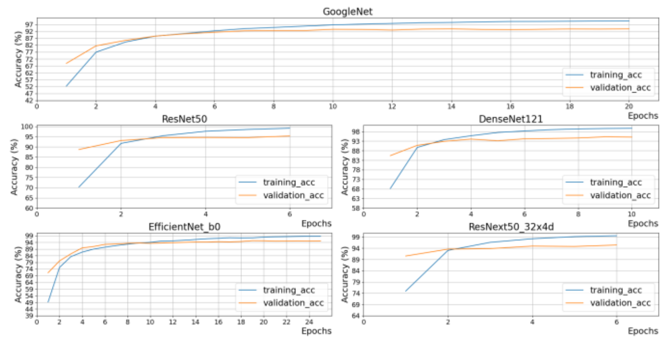

Figure 4. Change in accuracy with Adam optimizer.

The change in loss over the epochs with the Adam optimizer is depicted in Fig. 5.

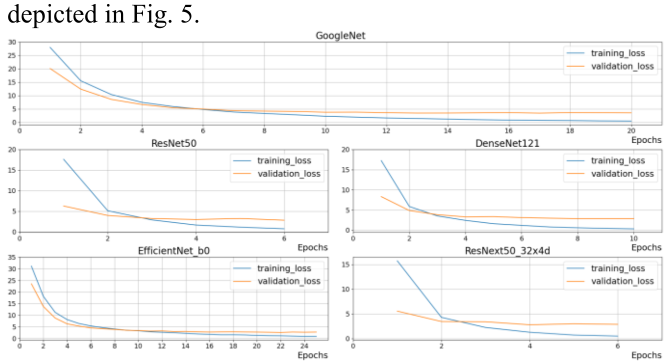

Figure 5. Change in loss with Adam optimizer.

The investigation continued with comparative analyses of model accuracy when an unknown set – test dataset is fed to the model. The stacked column chart in Fig. 6 compares the accuracy of each fine-tuned model with two optimizers.

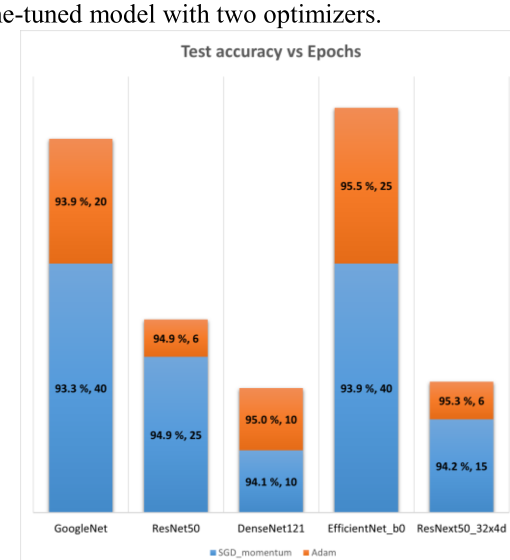

Figure 6. Comparison of test accuracy with SGD momentum and Adam optimizer.

The test accuracy with the SGD momentum optimizer is the highest, about 95 %, in the ResNet50 model using 25 epochs.

ResNeXt50 and DenseNet121 show almost identical performance with a slight difference in the number of epochs, where model accuracy is about 94 %. The performance of models is getting better with the Adam optimizer, where the number of epochs drastically declines. The most accurate model is ResNeXt50 with 6 epochs followed by EfficientNet_B0 with 25 epochs. The classification report for each architecture trained using the Adam optimizer, based on equations in (6-9), is depicted in Table I.

TABLE I. Classification report on the test dataset.

Model	GoogleNet			ResNet50			DenseNet121			EfficientNet_B0			ResNeXt50		
Metrics / labels	*Precision*	*Recall*	*F1-score*	*Precision*	*Recall*	*F1-score*	*Precision*	*Recall*	*F1-score*	*Precision*	*Recall*	*F1-score*	*Precision*	*Recall*	*F1-score*
battery	0.89	0.87	0.88	0.92	0.88	0.90	0.92	0.87	0.90	0.91	0.92	0.91	0.93	0.85	0.89
biological pictures	0.94	0.97	0.96	0.90	1.00	0.95	0.98	0.99	0.98	0.95	0.98	0.97	0.96	0.98	0.97
brown glass	0.98	0.97	0.98	0.98	1.00	0.99	0.98	0.97	0.98	0.98	0.97	0.98	0.97	0.98	0.98
cardboard	0.98	0.93	0.95	0.97	0.93	0.95	0.94	0.92	0.93	0.98	0.97	0.97	0.99	0.97	0.98
clothes	1.00	0.99	1.00	1.00	0.99	0.99	0.99	0.99	0.99	1.00	0.99	0.99	1.00	0.99	0.99
green glass	0.94	0.94	0.94	1.00	0.94	0.97	0.98	0.98	0.98	0.97	0.98	0.98	0.97	0.98	0.98
metal	0.82	0.82	0.82	0.82	0.88	0.85	0.83	0.87	0.95	0.85	0.88	0.87	0.88	0.87	0.88
paper	0.95	0.98	0.97	0.96	0.97	0.97	0.95	0.98	0.96	1.00	0.96	0.98	0.98	0.99	0.99
plastic	0.88	0.83	0.85	0.92	0.89	0.90	0.87	0.90	0.88	0.94	0.87	0.90	0.89	0.84	0.86
shoes	0.96	0.94	0.95	0.97	0.92	0.94	0.97	0.96	0.97	0.97	0.94	0.96	0.97	0.96	0.97
trash	0.96	0.99	0.97	0.99	1.00	0.99	0.99	0.97	0.98	0.99	0.97	0.98	0.97	0.99	0.98
white glass	0.81	0.90	0.85	0.88	0.90	0.89	0.90	0.88	0.89	0.83	0.94	0.88	0.82	0.97	0.89
Accuracy	93.9%			94.9%			95.0%			95.5%			95.3%		

Many researchers endeavored to achieve higher accuracy of garbage classification with various methods. The table below illustrates a comparison of obtained results on the test dataset.

TABLE II. Summary of results

Model name	Test accuracy	Classes	Reference
VGG16	75.6 %	4	[1]
ConvNet	76 %	4	[3]
MobileNet	77.3 %	7	[2]
Alexnet with SVM	80 %	5	[8]
VGG16	84 %	3	[12]
Yolo v3	85 %	4	[14]
ResNet50	87 %	4	[4]
Xception	88 %	6	[15]
Xception	92 %	5	[16]
Inception V3	93 %	12	[6]
Inception V3	93.2 %	4	[13]
DenseNet121	93.3 %	7	[10]
ResNeXt50	**95 %**	**12**	**Our simulation**

Model accuracy achieved in research [1] is the lowest, 75.6 %, while classification into seven categories of 93.3 % accuracy belongs to the research [10], according to the table. In this research, however, we obtained a more accurate classification result for twelve classes using ResNeXt50 architecture; the model accuracy accounts for approximately 95 %, slightly higher than the results obtained by [6, 10, 13].

V. Conclusion

In this research, we compared five neural network models to recognize waste images and classify them into twelve categories. We tuned the parameters of each model using two optimization algorithms. From the overall results and assessments, we consider that using transfer learning and fine-tuning technique give satisfactory outcomes. We also observed how the proper selection of the optimization algorithm significantly affects the result. The highest model accuracy we obtained is 95 % with the ResNeXt model optimized by the Adam algorithm.

This research can be extended with experiments to slightly enhance the accuracy and to be employed as a backbone to detect litter in the environment using unmanned aerial vehicles.

References

[1] H. Wang, “Garbage Recognition and Classification System Based on Convolutional Neural Network VGG16,” *2020 3rd International Conference on Advanced Electronic Materials, Computers and Software Engineering (AEMCSE)*, Apr. 2020, doi: 10.1109/aemcse50948.2020.00061.

[2] S. Frost, B. Tor, R. Agrawal, and A. G. Forbes, “CompostNet: An Image Classifier for Meal Waste,” *IEEE Xplore*, Oct. 01, 2019. https://ieeexplore.ieee.org/document/9033130

[3] S. R, R. P, V. S, K. R, and G. M, “Deep Learning based Smart Garbage Classifier for Effective Waste Management,” *2020 5th International Conference on Communication and Electronics Systems (ICCES)*, Jun. 2020, doi: 10.1109/icces48766.2020.9137938.

[4] O. Adedeji and Z. Wang, “Intelligent Waste Classification System Using Deep Learning Convolutional Neural Network,” *Procedia Manufacturing*, vol. 35, pp. 607–612, 2019, doi: 10.1016/j.promfg.2019.05.086.

[5] G. Huang, J. He, Z. Xu, and G. Huang, “A combination model based on transfer learning for waste classification,” *Concurrency and Computation: Practice and Experience*, vol. 32, no. 19, Apr. 2020, doi: 10.1002/cpe.5751.

[6] Y. Chen, W. Han, J. Jin, H. Wang, Q. Xing, and Y. Zhang, “Clean Our City: An Automatic Urban Garbage Classification Algorithm Using Computer Vision and Transfer Learning Technologies,” *Journal of Physics: Conference Series*, vol. 1994, no. 1, p. 12-22, Aug. 2021, doi: 10.1088/1742-6596/1994/1/012022.

[7] M. Malik *et al.*, “Waste Classification for Sustainable Development Using Image Recognition with Deep Learning Neural Network Models,” *Sustainability*, vol. 14, no. 12, p. 7222, Jun. 2022, doi: 10.3390/su14127222.

[8] Y. He, Q. Gu, and M. Shi, “Trash Classification Using Convolutional Neural Networks Project Category: Computer Vision.” [Online]. Available: http://cs230.stanford.edu/projects_spring_2020/reports/38847029.pdf

[9] V. Yaloveha, A. Podorozhniak, and H. Kuchuk, “Convolutional neural network hyperparameter optimization applied to land cover classification,” *RADIOELECTRONIC AND COMPUTER SYSTEMS*, no. 1, pp. 115–128, Feb. 2022, doi: 10.32620/reks.2022.1.09.

[10] Z. M. A. Amin, K. N. Sami, and R. Hassan, “An Approach of Classifying Waste Using Transfer Learning Method,” *International Journal on Perceptive and Cognitive Computing*, vol. 7, no. 1, pp. 41–52, Jul. 2021, [Online]. Available: https://journals.iium.edu.my/kict/index.php/IJPCC/article/view/213

[11] D. Chinassylov, A. Kozhamseitova, M. Kalen, and R. Omirgaliyev, “APPLICATION INFORMATION MODELING AND MACHINE LEARNING ALGORITHM FOR CLASSIFICATION OF WASTE USING SUPPORT VECTOR MACHINE,” *Scientific Journal of Astana IT University*, no. 8, pp. 45–53, Dec. 2021, doi: 10.37943/aitu.2021.52.74.005.

[12] D. A. P. S. Kumar, K. BuelaEvanzalina, and C. K, “AN EFFICIENT CLASSIFICATION OF KITCHEN WASTE USING DEEP LEARNING TECHNIQUES,” *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, vol. 12, no. 14, pp. 5751–5762, Dec. 2021, [Online]. Available: https://turcomat.org/index.php/turkbilmat/article/view/11782/8611

[13] L. Cao and W. Xiang, “Application of Convolutional Neural Network Based on Transfer Learning for Garbage Classification,” *2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC)*, Jun. 2020, doi: 10.1109/itoec49072.2020.9141699.

[14] N. Shah, L. Panigrahi, A. Patel, S. Tiwari, M. University, and K. Somaiya, “Classification and Segregation of Garbage for Recyclability Process,” *International Journal of Science and Research (IJSR)*, 2018.

[15] 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)*, Nov. 2020, doi: 10.1109/icicos51170.2020.9299017.

[16] A. M. Sali, H. Thampy, I. S. Vadakedam, and S. S. Pillai, “Waste Classification using Convolutional Neural Network on Edge Devices,” *www.semanticscholar.org*, 2020. https://www.semanticscholar.org/paper/Waste-Classification-using-Convolutional-Neural-on-Sali-Thampy/a57ed7fae1e82c78828966db118973d38954d97c

[17] “Garbage Classification (12 classes),” *www.kaggle.com*. https://www.kaggle.com/datasets/mostafaabla/garbage-classification

[18] Y. Yang *et al.*, “A comparative analysis of eleven neural networks architectures for small datasets of lung images of COVID-19 patients toward improved clinical decisions,” *Computers in Biology and Medicine*, vol. 139, p. 104887, Dec. 2021, doi: 10.1016/j.compbiomed.2021.104887.

[19] M. Tan and Q. Le, “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks,” *proceedings.mlr.press*, May 24, 2019. https://proceedings.mlr.press/v97/tan19a.html

[20] C. C. Aggarwal, *Neural Networks and Deep Learning*. Cham: Springer International Publishing, 2018. doi: 10.1007/978-3-319-94463-0.