Research Article

Deep transfer learning benchmark for plastic waste classification

Anthony Ashwin Peter Chazhoor[1], Edmond S. L. Ho[1], Bin Gao[2], Wai Lok Woo[1]

[1]Department of Computer and Information Sciences, Northumbria University, Newcastle upon Tyne NE1 8ST, UK.
[2]School of Automation Engineering, University of Electronic Science and Technology of China, Chengdu 610000, Sichuan, China.

Correspondence to: Prof. Wai Lok Woo, Department of Computer and Information Sciences, Northumbria University, Ellison Place, Newcastle upon Tyne NE1 8ST, UK. E-mail: wailok.woo@northumbria.ac.uk

Abstract

Millions of people throughout the world have been harmed by plastic pollution. There are microscopic pieces of plastic in the food we eat, the water we drink, and even the air we breathe. Every year, the average human consumes 74,000 microplastics, which has a significant impact on their health. This pollution must be addressed before it has a significant negative influence on the population. This research benchmarks six state-of-the-art convolutional neural network models pre-trained on the ImageNet Dataset. The models Resnet-50, ResNeXt, MobileNet_v2, DenseNet, SchuffleNet and AlexNet were tested and evaluated on the WaDaBa plastic dataset, to classify plastic types based on their resin codes by integrating the power of transfer learning. The accuracy and training time for each model has been compared in this research. Due to the imbalance in the data, the under-sampling approach has been used. The ResNeXt model attains the highest accuracy in fourteen minutes.

Keywords: Plastic, transfer learning, recycling, waste, classification

1. INTRODUCTION

Plastic finds itself in everyday human activities. The mass production of plastic was introduced in 1907 by Leo Baekeland, proved to be a boon to humankind[1]. Over the years, plastic has increasingly become an everyday necessity for humanity. The population explosion has a critical part in increasing domestic plastic usage[2]. Lightweight plastics have a crucial role in the transportation industry. Their usage in space

exploration gives enormous leverage over heavy and expensive alternatives[3]. The packaging industry widely uses plastics after the e-commerce revolution because they are lightweight, cheap, and abundant. In 2015, the packing sector produced 141 million metric tons of garbage, accounting for 97 percent of all waste produced concerning the total consumption in the packaging sector[4]. Discarded polyethylene terephthalate (PETE) bottles are a common source of household waste. In 2021, global waste plastic bottle consumption will surpass 500 billion as estimated[2].

The increasing use of plastics and their wastage negatively affect the global economy. This surge in consumption and the low degradability of plastic have resulted in massive plastic accumulation in the environment, which has harmed ecosystems and human health[5]. This has resulted in countries formulating strict policies for plastics and even banning some types of single-use plastics. Plastics are non-biodegradable and considerably take a longer time to degrade. Reusing and recycling are viable ways to stop contaminating the environment with plastic pollution[6]. Plastic wastes can be retrieved after entering the municipal treatment plants or before it. However, the plastic waste from the municipal treatment plants is usually contaminated and ends up in landfills or incineration centers. The plastic waste collected outside of such plants is relatively cleaner and can be reused or recycled. Recovered plastics from such wastes have varied types of plastic, making it extremely difficult to identify and sort different kinds of plastics.

By integrating transfer learning, the Dataset needs only a limited number of input images to acquire high accuracy, and it also accelerates the training of neural networks, consequently improving the classification of multiple classes in a dataset[7]. Balancing the number of images in each class compensates for the class imbalance problem. This research contributes towards benchmarking of pre-trained models and concluding that the ResNeXt model achieves the highest accuracy on the WaDaBa dataset from the list of pre-trained models specified in this paper.

1.1. Literature review
Seven different varieties of plastics exist in the modern day. They are classified as Polyethylene terephthalate (PET or PETE), high-density polyethylene (HDPE), polyvinyl chloride (PVC or Vinyl), low-density polyethylene (LDPE), polypropylene (PP), polystyrene (PS or Styrofoam) and Others, which does not belong to any of the above types, has been shown in Figure 1[3].

1.1.1. Traditional sorting techniques
Initially, segregation of wastes and separation of different types of plastics were done manually. However, this results in increased labor costs and time consumption[6]. Traditional macro sorting of plastics was performed with the aid of sensors which included near-infrared spectrometers[8,9], x-ray transmission sensor, Fourier transformed Infrared Technique[10], laser aided identification, and marker identification by identifying the resin type[11]. However, these approaches are limited to recognizing just particular types of plastics and are costly due to the large equipment required. The intricacy of mechanical sorting and its maintenance, as well as the high initial investment, are the drawbacks of traditional sorting methods.

1.1.2. Modern sorting techniques
Deep learning has made classification easier, more efficient, and cost-effective, with less human intervention. The deep learning approach was enhanced by convolutional neural networks (CNN)[12]. CNNs are excellent for object classification and detection[13]. After the model has been trained on the data, the plastics may be sorted into the appropriate classes with the assistance of CNN. They do, however, require a huge quantity of training data, which might be difficult to get at times. When the input data is small, the problem of overfitting develops, resulting in inaccurate classifications[14]. Transfer learning reduces the

Figure 1. Types of plastic, its resin code and everyday examples of plastics. PETE: Polyethylene terephthalate; HDPE: high-density polyethylene; PVC: polyvinyl chloride; LDPE: low-density polyethylene; PP: polypropylene, PS: polystyrene.

training time of a CNN by pre-training the model using benchmark datasets such as ImageNet.

Bobulski *et al.*[15] proposed an end-to-end system with a micro-computer embedded with the vision to sort the PETE types of plastics in the WaDaBa dataset. The authors introduced data augmentation, which reduced the number of parameters but exponentially increased the number of samples, increasing the training time. Bobulski *et al.*[16] also proposed to classify distinct plastic categories based on a gradient feature vector. Agarwal *et al.*[17] presented Siamese and triplet loss neural networks to classify the WaDaBa dataset and succeeded with very high accuracy. However, this method requires a significant amount of time for training the neural networks. Chazhoor *et al.*[18] Anthony utilised transfer learning to compare the three most often used architectures (ResNeXt, Resnet-50-50 and AlexNet) on the WaDaBa dataset to select the optimal model; however, the K-fold cross validation technique was not applied; as a result, testing accuracy would vary widely.

The aim of the paper is to provide researchers with benchmark accuracies and the average time required to train on the WaDaBa dataset using the latest CNN models utilising cross-validation to categorise a range of plastics into their appropriate resin types. An unbiased and concrete set of parameters has been set to evaluate the Dataset to compare the models fairly[19]. This benchmark work will assist in gaining an impartial view of numerous recent CNN models applied to the WaDaBa dataset, establishing a baseline for future research. The models used in this paper are AlexNet[20], Resnet-50[21], ResNeXt[22], SqueezeNet[23], MobileNet_v2[24] and DenseNet[25].

2. METHODS

2.1. Dataset

The WaDaBa dataset is a sophisticated collection that contains images of common plastics used in society. The dataset includes seven distinct varieties of plastic. Images show several forms of plastics on a platform under two lighting conditions: an LED bulb and a fluorescent lamp and is displayed in Figure 2. Table 1 shows the distribution of the 4000 images in the dataset according to their classes. As there are no images in the PVC and PE-LD classes, both the classes have been excluded from the deep learning models. Deep learning models are trained on five class types with images in the current work i.e., PETE, PE-HD, PP, PS, and Other. The deep learning models are set up in such a way that each output matches one of the five class categories. When the images for PVC and PE-LD are released, these classes can be included in the models. The dataset's classes are imbalanced, with the last class holding just 40 images and the PETE class consisting of 2000 images. The dataset is freely accessible to the public[15].

2.2. Transfer learning

A large amount of data is needed to get optimum accuracy in a neural network. Data needs to be trained for hours on a powerful Graphical Processing Unit (GPU) to get the results. With the advent of transfer learning[26], there has been a significant change in the learning processes in deep neural networks. The model which has been already trained on a large dataset like ImageNet[27], known as the pre-trained model, enhances the transfer learning process. The transfer learning process works by freezing[28] the initially hidden layers of the model and fine-tuning the final layers of the models. The layer's frozen state indicates that it will not be trained. As a result, its weights will remain unchanged. As the data set used in this research is relatively small with a limited number of images in each class, transfer learning best suits this research. The pre-trained models used in the research are further explained in the subsection.

2.2.1. AlexNet

AlexNet is a neural network with three convolutional layers and two fully connected layers, and it was introduced in 2012 by Alex Krizhevesky. AlexNet increases learning capacity by increasing network depth and using multi-parameter tuning techniques. AlexNet uses ReLU to add non-linearity and dropout to decrease the overfitting of data. CNN-based applications gained popularity following AlexNet's excellent performance on the ImageNet dataset in 2012[23]. The architecture of AlexNet is shown in Figure 3.

2.2.2. Resnet-50

Residual networks (Resnet-50) are convolutional neural networks with skip connections with an extremely deep convolution and 11 million parameters. A skip connection after each block solves the vanishing gradient problem. The skip connection skips some layers in the network. With batch normalization and ReLU activation, two 3 × 3 convolutions are used in each block to achieve the desired result[21]. The architecture of Resnet-50-50 is displayed in Figure 4.

2.2.3. ResNeXt

Proposed by Facebook and ranking second in ILSVRC 2016, ResNeXt uses the repeating layer strategy of Resnet-5050, and it appends the split-transform-merge method[22]. The magnitude of a set of transformations is known as cardinality. Cardinality provides a novel approach to modifying model capacity by increasing the number of separate routes. Having width and depth as critical characteristics, ResNeXt adds on Cardinality as a new dimension. Increasing cardinality is a practical approach to enhance the accuracy of the model[22]. The architecture of ResNeXt is shown in Figure 5.

Table 1. The number of images corresponding to each class in the WaDaBa dataset[15]

Resin code	Class type	Number of images
1	PETE	2200
2	PE-HD	600
3	PVC	0
4	PE-LD	0
5	PP	640
6	PS	520
7	Other	40

PETE: Polyethylene terephthalate; PVC: polyvinyl chloride; PP: polypropylene, PS: polystyrene; PE-HD: high-density polyethylene; PE-LD: low-density polyethylene.

Figure 2. Examples of different types of plastics from the WaDaBa dataset in Figure 1. (A) Class 1 representing PETE (polyethylene terephthalate); (B) Class 2 representing HDPE (high-density polyethylene); (C) Class 5 representing PP (polypropylene); (D) Class 6 representing PS (polystyrene) ; (E) Class 7 representing Others[15].
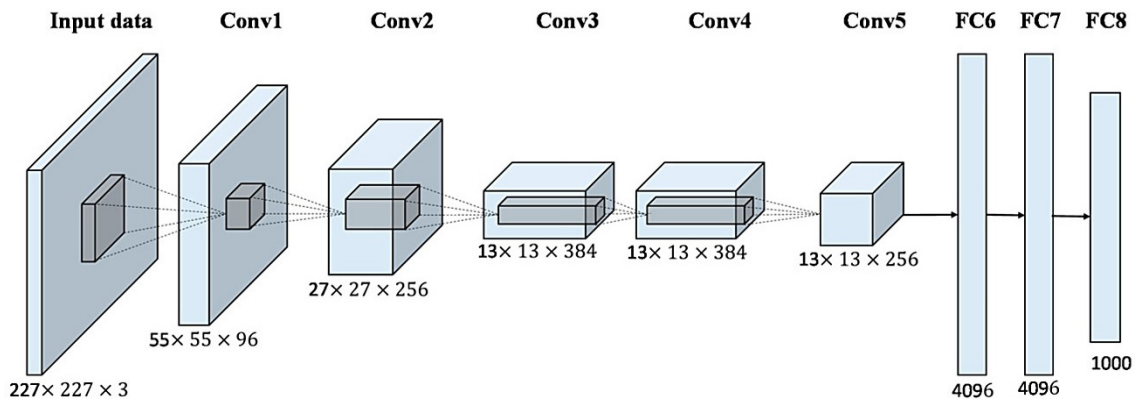

Figure 3. The architecture of AlexNet, having five convolutional layers and three fully connected layers. This figure is quoted with permission from Han *et al.*[29].
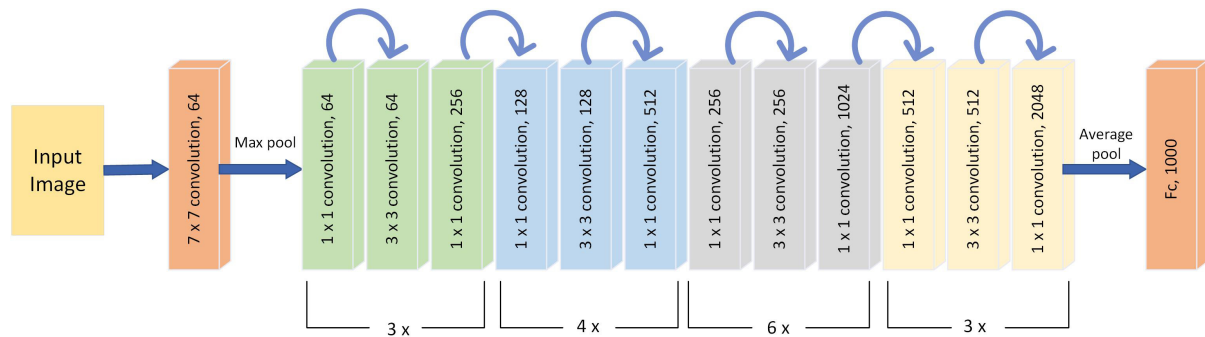

Figure 4. Architecture of Resnet-50-50. This figure is quoted with permission from Talo *et al.*[30].
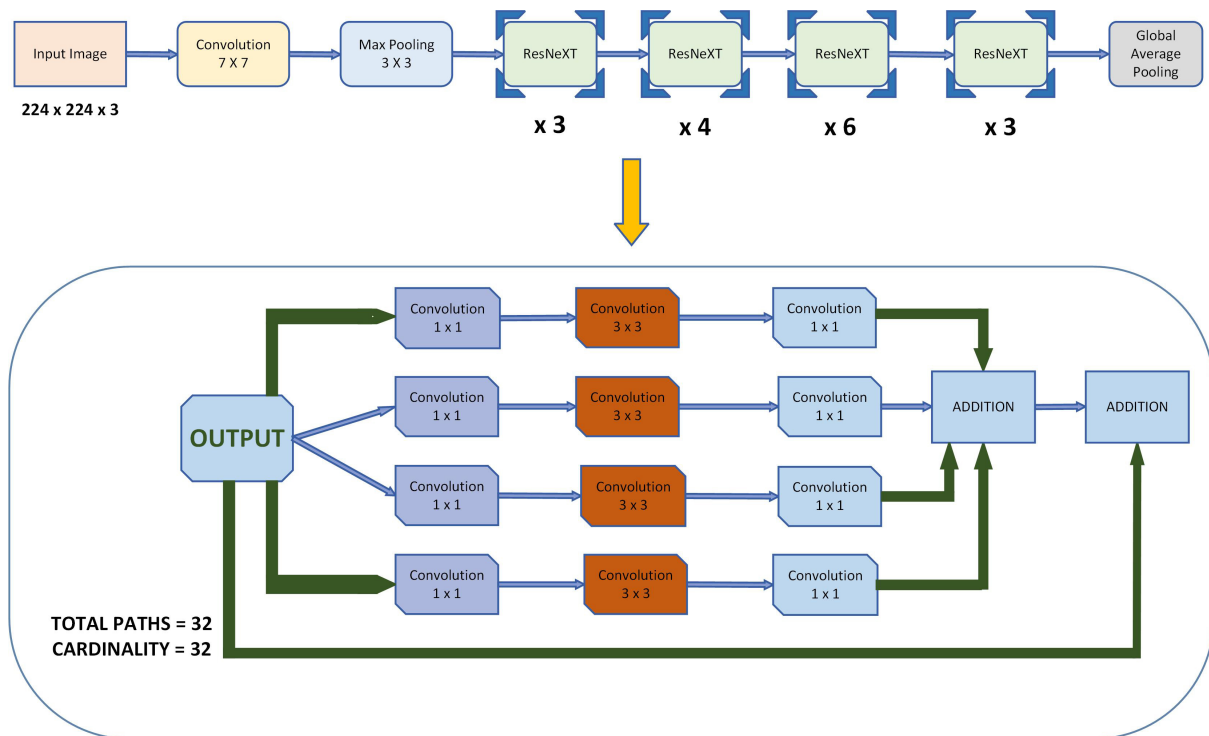
Figure 5. Architecture of ResNeXt. (Figure is redrawn and quoted from Go *et al.*[31])

2.2.4. MobileNet_v2

MobileNet_v2 is a CNN architecture built on an inverted residual structure, shortcut connections between narrow bottleneck layers to improve the mobile and embedded vision systems. A Bottleneck Residual Block is a type of residual block that creates a bottleneck using 1 × 1 convolutions. The number of parameters and matrix multiplications can be reduced by using a bottleneck. The goal is to make residual blocks as small as possible so that depth may be increased, and the parameters can be reduced. The model uses ReLU as the activation function. The architecture comprises a 32-filter convolutional layer at the top, followed by 19 bottleneck layers[24]. The architecture of MobileNet_v2 is shown in Figure 6.

2.2.5. DenseNet

Using a feed-forward system, DenseNet connects each layer to every other layer. Layers are created using feature maps from all previous levels, and their feature maps are utilized in all future layers to create new layers. They solve the vanishing-gradient problem and improve feature propagation and reuse while reducing the number of parameters significantly. The architecture of DenseNet is shown in Figure 7.

2.2.6. SqueezeNet

SqueezeNet is a small CNN that shrinks the network by reducing parameters while maintaining adequate accuracy. An entirely new building block has been introduced in the form of SqueezeNet's Fire module. A Fire module consists of a squeeze convolution layer containing only a 1 × 1 filter, which feeds into an expand layer having a combination of 1 × 1 and 3 × 3 convolution filters. Starting with an independent convolution layer, SqueezeNet then moves to 8 Fire modules before concluding with a final convolution layer. The architecture of SqueezeNet is shown in Figure 8.
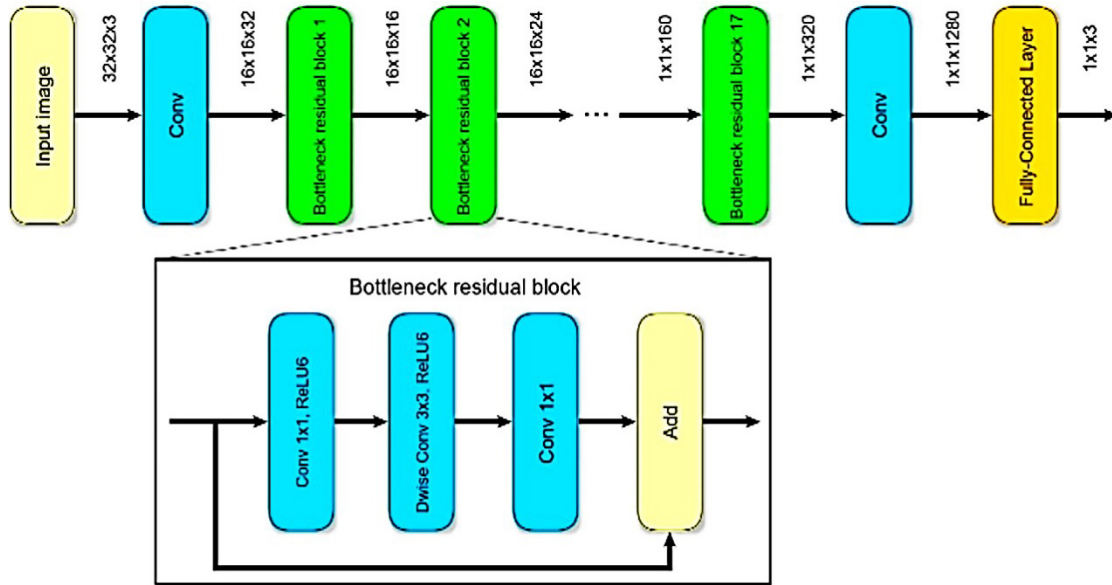
Figure 6. The architecture of MobileNet_v2. This figure is quoted with permission from Seidaliyeva *et al.*[32]
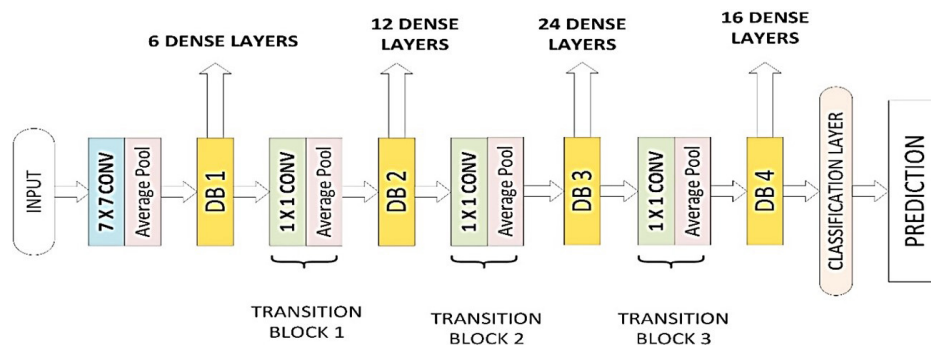
Figure 7. The architecture of DenseNet. This figure is quoted with permission from Huang *et al.*[25].

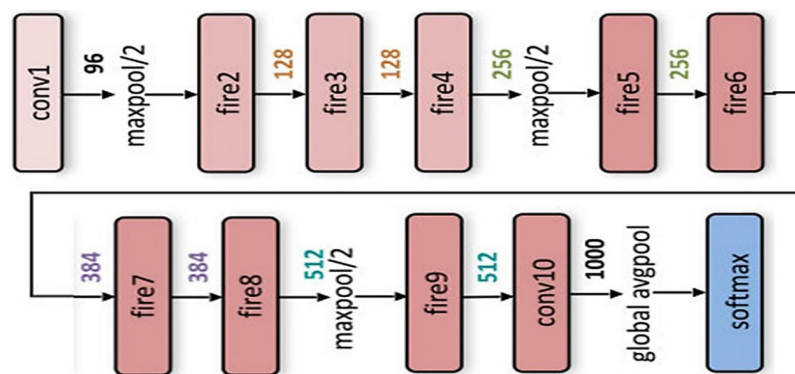**Figure 8.** The architecture of SqueezeNet. This figure is quoted with permission from Nguyen *et al.*[33].

2.3. Experimental settings and the experiment

All the experiments were run on Ubuntu Linux operating system. The models were trained on Intel i7, 3.60 GHz, 32 GB ram and the graphical processing unit used was the Nvidia GeForce RTX 2080 Super. The

deep learning framework used in this research is PyTorch[34]. The images from the WaDaBa dataset are input to the pre-trained models after performing under-sampling in the dataset. The batch size chosen for this experiment is 4 such that the GPU doesn't run out of memory while processing. The learning rate is 0.001 and is decayed by a factor of 0.1 every seven epochs. Decaying the learning rate aids the network's convergence to a local minimum and also enhances the learning of complicated patterns[35]. Cross-Entropy loss is utilized for training, accompanied by a momentum of 0.9, which is widely used in the machine learning and neural network communities[36]. The Stochastic Gradient Descent (SGD) optimizer[37], a gradient descent technique that is extensively employed in training deep learning models, is used. The training is done using a five-fold cross-validation technique, and the result is generated, along with graphs showing the number of epochs *vs.* accuracy and number of epochs *vs.* loss. On the WaDaBa dataset, each model was subjected to twenty epochs.

Before being forwarded on to the training, the data was normalized. These approaches, which were applied to the data, included random horizontal flipping and centre cropping.

The size of the input picture is 224 × 224 pixels [Figure 9].

2.3.1. Imbalance in the dataset
The number of images for each class in the dataset is uneven. The first class (PETE) contains 2200 photos, while the last class (Others) contains only 40. Due to the size and cost of certain forms of plastic, obtaining datasets is quite tricky. Because of the class imbalance, the under-sampling strategy was used. Images were split into training and validation sets, eighty percent for the training and twenty percent for the testing purposes.

2.3.2. K-fold cross-validation
The 5-fold cross-validation was considered for all the tests to validate the benchmark models[38]. The data was tested on the six models and the training loss and accuracy, validation loss and accuracy and the training time was recorded for 20 epochs with identical model parameters. The resultant average data was tabulated, and the corresponding graphs were plotted for visual representation. The flow chart of the experimental process is displayed in Figure 8.

3. RESULTS
3.1 Accuracy, loss, area under curve and receiver operating characteristic curve
The metrics used to benchmark the models on the WaDaBa dataset are accuracy and loss. The accuracy corresponds to the correctness of the value[39]. It measures the value to the actual value. Loss is a prediction of how erroneous the predictions of a neural network are, and the loss is calculated with the help of a loss function[40]. The area under curve (AUC) measures the classifier's ability to differentiate between classes and summarize the receiver operating characteristic (ROC) curve. ROC plots the performance of a classification model's overall accuracy. The curve plots the True Positive Rate against the False Positive Rate.

Table 2 clearly shows that the ResNeXt architecture achieves the maximum accuracy of 87.44 percent in an average time of thirteen minutes and eleven seconds. When implemented in smaller and portable devices, smaller networks such as MobileNet_v2, SqueezeNet, and DenseNet offer equivalent accuracy. AlexNet trains the model in the shortest period but with the lowest accuracy. In comparison to the other models, DenseNet takes the longest to train. With a classification accuracy of 97.6 percent, ResNeXt comes out as the top model for reliably classifying PE-HD. When compared to other models, MobileNet_v2 classifies PS with more accuracy. Also, from Table 2, we can see that PP has the least classification accuracy for all the

Table 2. The mean and class wise accuracies of the models pretrained on the ImageNet dataset, along with the time taken for training for 20 epochs. The standard deviation indicates the average deviation in accuracy across the five-folds in the respective model along with the total number of parameters for each model

	AlexNet	Resnet-50	ResNeXt	MoblineNet_v2	DenseNet	SqueezeNet
Mean accuracy (%)	80.08	85.54	**87.44**	87.35	85.58	82.59
PETE (%)	84.8	85	85	85	**88.8**	84.4
PE-HD (%)	85.0	95.4	**97.6**	94.2	95.6	91.4
PP (%)	67.2	68.6	74	**74.8**	66.4	66.8
PS (%)	80.2	86.0	83.2	**89.6**	85.4	82.2
Other (%)	100	100	100	100	100	97.5
Time (min)	11.8	12.05	13.11	12.06	17.33	12.01
Std. deviation σ (%)	7.5	4.9	5.4	6.0	5.3	1.7
No. of parameters (in million)	57	23	22	2	6	0.7

PETE: Polyethylene terephthalate; PP: polypropylene, PS: polystyrene.
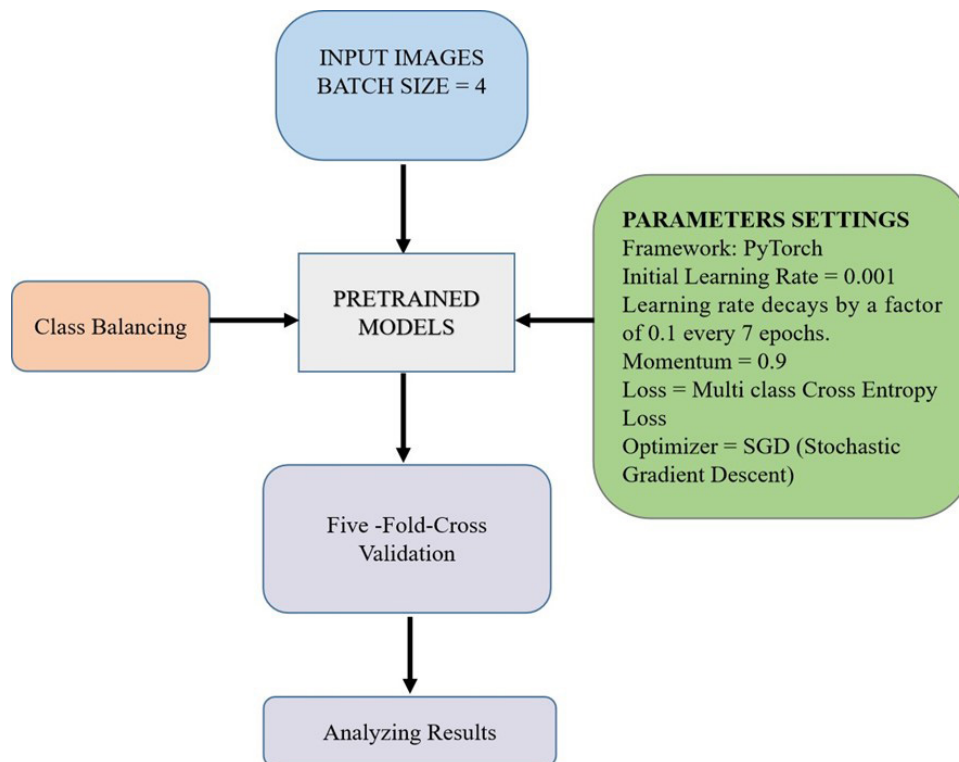

Figure 9. Flowchart summarizing the experiment.

models. In Table 2, the standard deviation, σ, is displayed, which is a measure of how far values deviate from the mean. The standard deviation is given by the following unbiased estimation:

$$\sigma = \sqrt{\frac{\sum(x_i - \bar{x})^2}{n-1}}$$

x_i = accuracy at the ith epoch

$\bar{x}$ = mean of the accuracies

n = total number of epochs (e.g., 20)

4. DISCUSSION

In the results section from Table 2, we can observe that ResNeXt architecture performs better than all the other architectures discussed in this paper. MobileNet_v2 architecture falls behind ResNeXt architecture with 0.1 % accuracy. Considering the time factor, MobileNet_v2 trains faster than ResNext by a minute's advantage. When the data is considerably large, the difference in time factor will increase, giving the MobileNet_v2 architecture dominance.

The validation loss of AlexNet architecture from Table 3 and SqueezeNet architecture from Table 4 does not significantly drop compared to other models used in the research and from the graph, it can be observed from Figure 10 and Figure 11 that there is a diverging gap between its accuracy loss and validation loss curves for both models. Fewer images in the Dataset and multiple classes cause this effect on the AlexNet architecture. Similar results can be observed for SqueezeNet from Table 4 and Figure 11, which have a similar architecture to AlexNet. Table 5 and Figure 12 represent the training and validation accuracies and loss values and their corresponding graphs for the pre-trained Resnet-50 model. From Table 6 and Figure 13, we can observe the training and validation accuracy and loss values and their plots for ResNeXt architecture. Similarly, from Table 7 and Figure 14, the accuracies and their graphs for MobileNet_v2 can be observed. The DenseNet architecture represented in Table 8 and Figure 15 takes the longest time to train and has a good accuracy score of 85.58%, which is comparable to the Resnet-50 architecture, having an accuracy of 85.54%. The five-fold cross-validation approach tests every data point in the dataset and helps improve the overall accuracy.

Figure 16 shows the AUC and ROC for all the models in this paper. The SqueezeNet and AlexNet architecture display the lowest AUC score. MobileNet_v2, Resnet-50, ResNext and DenseNet have a comparable AUC score. From the ROC curve, it can be inferred that the models can correctly distinguish between the types of plastics in the Dataset. ResNeXt architecture achieves the largest AUC.

5. CONCLUSION

When we compare our findings to previous studies in the field, we find that including transfer learning reduces total training time significantly. It will be simple to train the existing model and attain improved accuracy in a short amount of time if the WaDaBa dataset is enlarged in the future. This paper has benchmarked six state-of-the-art models on the WaDaBa plastic dataset by integrating deep transfer learning. This work will be laid out as a baseline work for future developments on the WaDaBa dataset. The paper focuses on supervised learning for plastic waste classification. Unsupervised learning procedures are one area where the article has placed less focus. The latter might be beneficial for pre-training or enhancing the supervised classification models using pre-trained feature selection. Pattern decomposition methods[41] like nonnegative matrix factorization[42] and ensemble joint sparse low rank matrix decomposition[43] are

Table 3. The mean training and validation accuracies and losses for AlexNet architecture for 20 epochs

Epoch	Mean_AlexNet			
	Training accuracy	**Validation accuracy**	**Training loss**	**Validation loss**
1	0.5815	0.57302	1.00228	1.1308
2	0.6675	0.64806	0.80658	1.09448
3	0.7177	0.5804	0.69244	1.1246
4	0.73384	0.64656	0.6721	1.01474
5	0.77882	0.67598	0.55144	0.9506
6	0.78652	0.66568	0.51194	1.04706
7	0.79548	0.7093	0.50188	0.84044
8	0.84654	0.7696	0.36054	0.82302
9	0.87302	0.7642	0.30162	0.89168
10	0.87962	0.77646	0.28896	0.90384
11	0.87458	0.77746	0.29108	0.92258
12	0.88206	0.78874	0.28282	0.8886
13	0.88462	0.78236	0.26542	0.99196
14	0.88192	0.78532	0.26406	0.99434
15	0.89248	0.78972	0.25636	0.98168
16	0.89126	0.78972	0.2576	0.98266
17	0.88914	0.79118	0.25864	0.95596
18	0.897	0.79608	0.24166	0.95004
19	0.89344	0.79706	0.24634	0.9735
20	0.89602	0.79414	0.24826	0.98582

Table 4. The mean training and validation accuracies and losses for SqueezeNet architecture for 20 epochs

Epoch	Mean SqueezeNet			
	Training accuracy	**Validation accuracy**	**Training loss**	**Validation loss**
1	0.47992	0.7281	1.02608	1.32476
2	0.64688	0.7437	0.78012	0.96076
3	0.7134	0.718	0.68612	1.05972
4	0.74428	0.67796	0.6426	1.14184
5	0.76116	0.7003	0.5903	0.81164
6	0.79006	0.70916	0.53186	0.88014
7	0.81026	0.65862	0.51222	0.89182
8	0.85586	0.69658	0.42766	0.81594
9	0.87364	0.70138	0.3871	0.89832
10	0.87874	0.70724	0.37834	0.99886
11	0.88684	0.6838	0.3752	0.9401
12	0.89062	0.69988	0.36256	0.93402
13	0.89798	0.69218	0.3465	0.94986
14	0.88878	0.7183	0.36842	0.8951
15	0.89504	0.70776	0.35906	0.97796
16	0.89798	0.70376	0.35146	1.0066
17	0.89896	0.70712	0.35242	0.99574
18	0.90166	0.70396	0.34732	1.00284
19	0.90422	0.70202	0.34508	1.01182
20	0.90238	0.70606	0.34562	0.9707

Table 5. The mean training and validation accuracies and losses for Resnet-50 architecture for 20 epochs

Epoch	Mean Resnet-50 values			
	Training accuracy	**Validation accuracy**	**Training loss**	**Validation loss**
1	0.5515	0.6706	1.12794	1.04068
2	0.69346	0.70782	0.81024	0.96718
3	0.7455	0.7691	0.66772	0.86036
4	0.77918	0.76568	0.5758	0.82058
5	0.80062	0.77648	0.52012	0.66052
6	0.8256	0.75932	0.44886	0.85278
7	0.83992	0.74364	0.42794	1.16314
8	0.87704	0.82598	0.32214	0.60218
9	0.89198	0.82254	0.2835	0.6571
10	0.90986	0.82942	0.24506	0.62152
11	0.90324	0.83382	0.2566	0.58042
12	0.91498	0.83234	0.23156	0.63032
13	0.91182	0.81626	0.23618	0.6429
14	0.91476	0.83726	0.23086	0.65462
15	0.9151	0.83484	0.2235	0.6636
16	0.91464	0.82894	0.22348	0.70444
17	0.91684	0.8343	0.21748	0.65494
18	0.91684	0.83776	0.21546	0.6189
19	0.91708	0.83482	0.22578	0.68982
20	0.91352	0.83922	0.22412	0.61236

Table 6. The mean training and validation accuracies and losses for ResNeXt architecture for 20 epochs

Epoch	Mean ResNeXt values			
	Training accuracy	**Validation accuracy**	**Training loss**	**Validation loss**
1	0.57454	0.71078	1.09714	0.97576
2	0.69518	0.74312	0.8304	0.87308
3	0.752	0.67498	0.66784	1.3998
4	0.79228	0.76764	0.57174	0.93114
5	0.81336	0.78234	0.52164	0.7225
6	0.83306	0.83136	0.4542	0.70478
7	0.84494	0.81374	0.42144	0.7807
8	0.88366	0.8564	0.30548	0.5644
9	0.89836	0.85442	0.28038	0.64594
10	0.90642	0.85294	0.26156	0.62974
11	0.90826	0.85834	0.2503	0.65006
12	0.9145	0.85	0.2385	0.6518
13	0.9084	0.84118	0.2411	0.64972
14	0.91084	0.8544	0.24424	0.59668
15	0.91316	0.85246	0.2417	0.55656
16	0.92564	0.84854	0.2097	0.58186
17	0.91156	0.85882	0.23282	0.58778
18	0.916	0.85688	0.22358	0.63122
19	0.91598	0.84658	0.223	0.62936
20	0.92014	0.85246	0.21606	0.65276

Table 7. The mean training and validation accuracies and losses for MobileNet_v2 architecture for 20 epochs

Epoch	Mean MobileNet_v2			
	Training accuracy	Validation accuracy	Training loss	Validation loss
1	0.55528	0.66322	1.12416	0.97572
2	0.64264	0.71714	0.94286	0.79604
3	0.6871	0.77108	0.806	0.77816
4	0.72912	0.7392	0.70786	0.89686
5	0.75566	0.74462	0.6542	0.8389
6	0.7858	0.78334	0.57576	0.75382
7	0.78846	0.7799	0.54498	0.86344
8	0.8392	0.83332	0.4141	0.62084
9	0.85942	0.8495	0.36976	0.57796
10	0.8649	0.85296	0.35118	0.57304
11	0.87458	0.84954	0.33336	0.57328
12	0.87606	0.85734	0.32184	0.5281
13	0.8768	0.86618	0.3207	0.50986
14	0.88106	0.84902	0.31194	0.545
15	0.88464	0.85344	0.30746	0.53638
16	0.88756	0.86178	0.2966	0.5141
17	0.88804	0.8613	0.30038	0.50172
18	0.88342	0.8608	0.30566	0.52828
19	0.88512	0.85688	0.30972	0.53054
20	0.8822	0.86176	0.31576	0.50632

Table 8. The mean training and validation accuracies and losses for DenseNet architecture for 20 epochs

Epoch	Mean DenseNet			
	Training accuracy	Validation accuracy	Training loss	Validation loss
1	0.55724	0.6446	1.0884	1.04494
2	0.68426	0.73088	0.81858	0.74552
3	0.7488	0.72302	0.6718	1.14064
4	0.76168	0.75196	0.64602	0.90288
5	0.7874	0.79118	0.5675	0.69646
6	0.81936	0.76862	0.50594	0.85718
7	0.82216	0.77744	0.48568	0.76844
8	0.87188	0.79952	0.36034	0.66998
9	0.87814	0.83136	0.31836	0.51186
10	0.8911	0.80736	0.30766	0.5814
11	0.8954	0.82354	0.28282	0.58526
12	0.90164	0.83874	0.27306	0.59644
13	0.89908	0.8392	0.2748	0.5592
14	0.9019	0.84118	0.27446	0.57224
15	0.90704	0.83578	0.25116	0.5755
16	0.9096	0.84366	0.24786	0.5398
17	0.90582	0.84216	0.24938	0.5301
18	0.9063	0.84316	0.26094	0.60658
19	0.91196	0.8299	0.24698	0.57962
20	0.9079	0.84364	0.24388	0.52476

Accuracy Curves for AlexNet

Loss curves for AlexNet
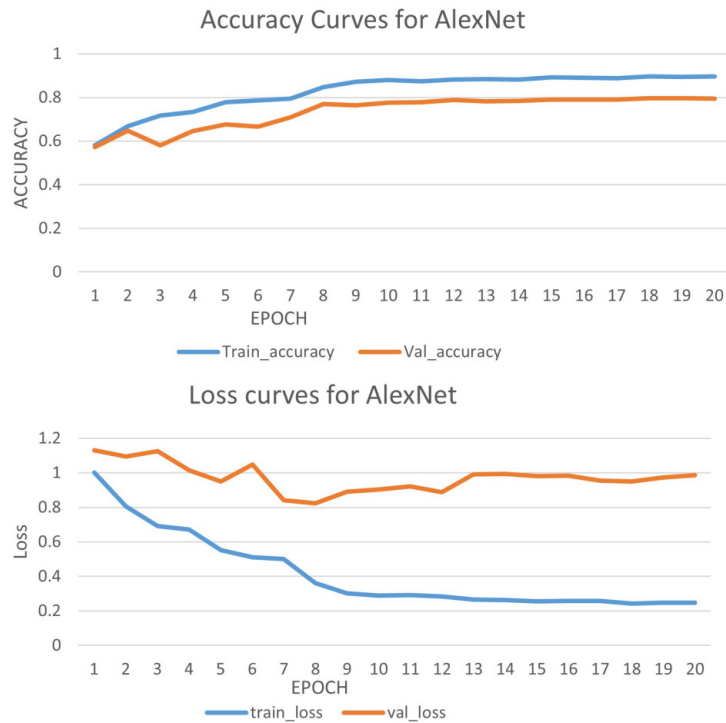
Figure 10. Accuracy and loss curves for AlexNet architecture.

Accuracy Curves for SqueezeNet

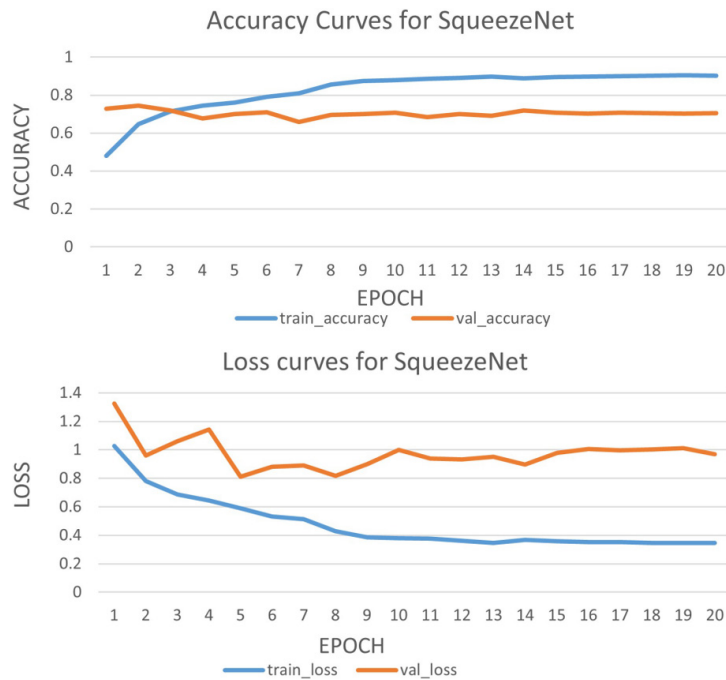### Loss curves for SqueezeNet

Figure 11. Accuracy and loss curves for SqueezeNet architecture.

examples of unsupervised learning strategies. Higher order decomposition approaches, such as low-rank tensor decomposition[44,45] and hierarchical sparse tensor decomposition[46], can result in improved performance. This would be the future path of study to improve plastic waste classification.
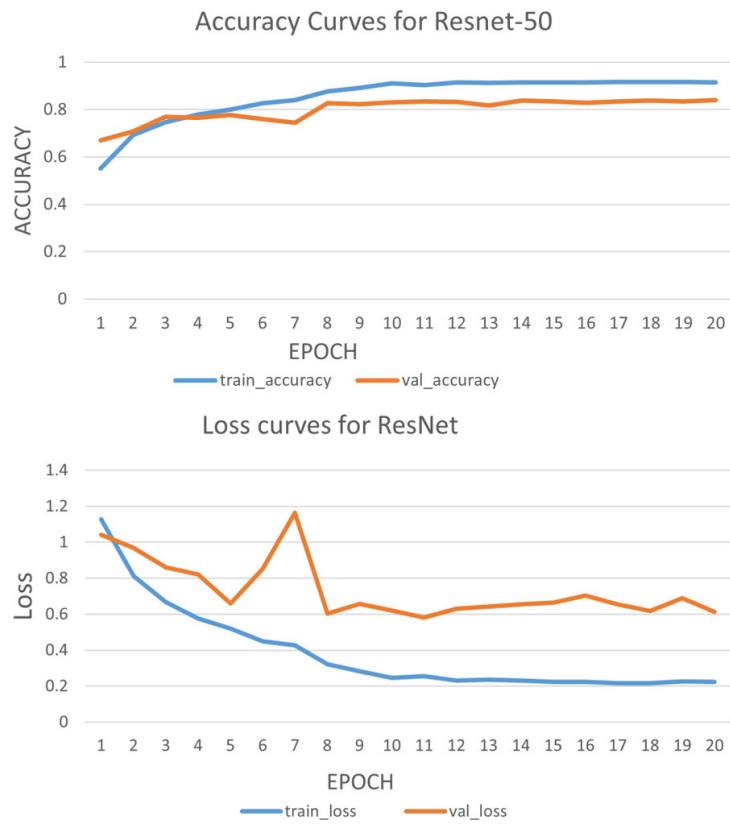
Accuracy Curves for Resnet-50

Loss curves for ResNet

Figure 12. Accuracy and loss curves for Resnet-50 architecture.

Accuracy Curves for ResNext
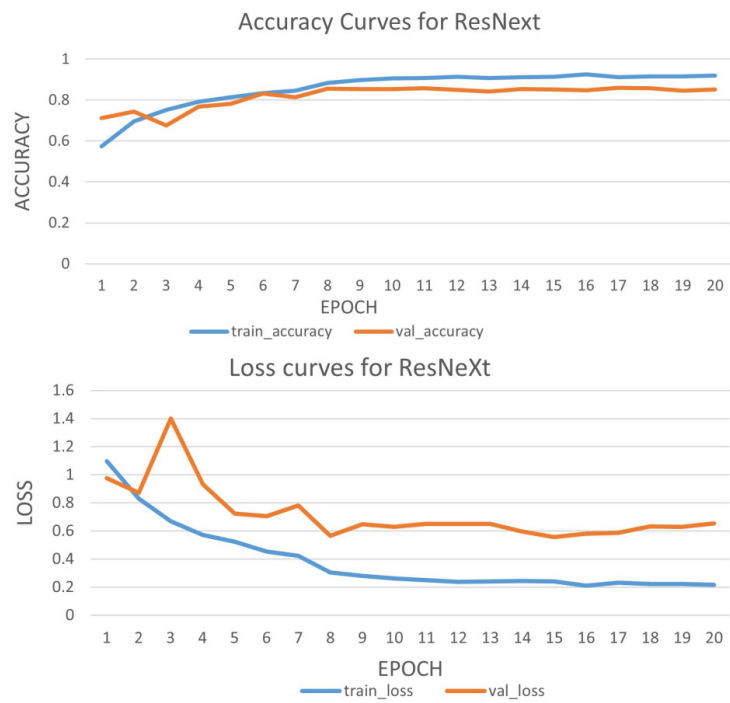

Loss curves for ResNeXt

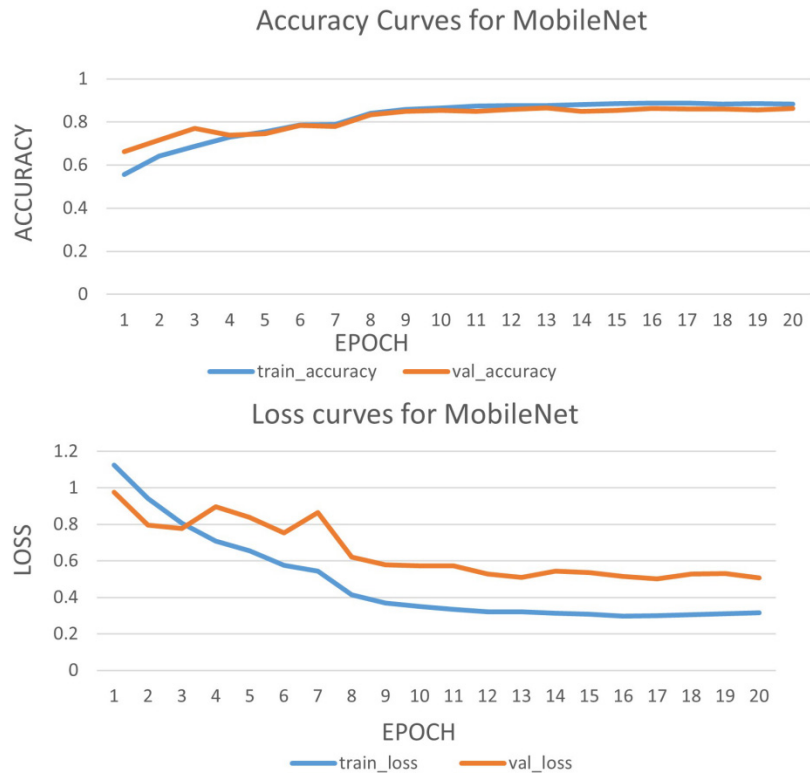**Figure 13.** Accuracy and loss curves for ResNeXt architecture.

Figure 14. Accuracy and loss curves for MobileNet_v2 architecture.
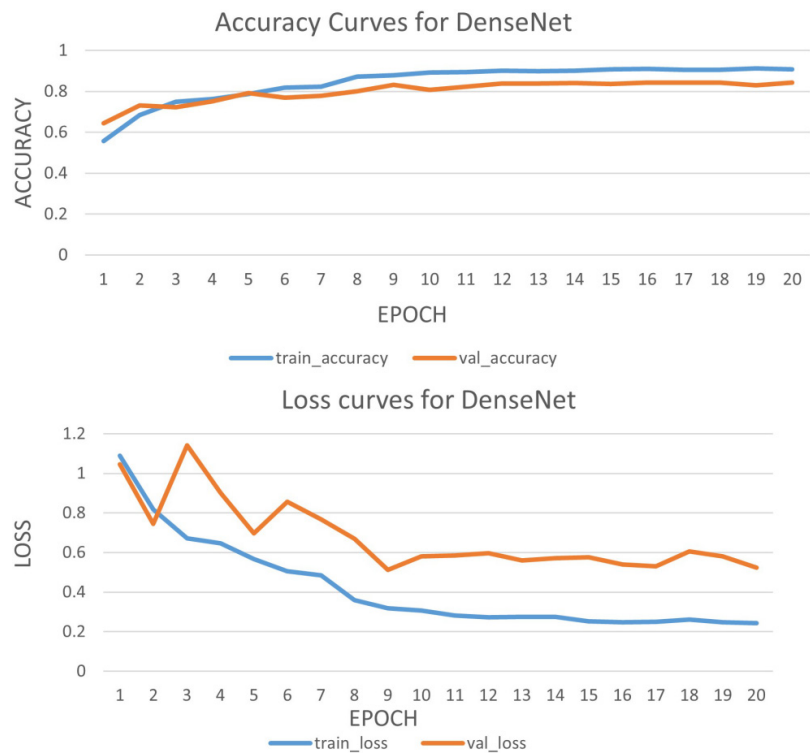

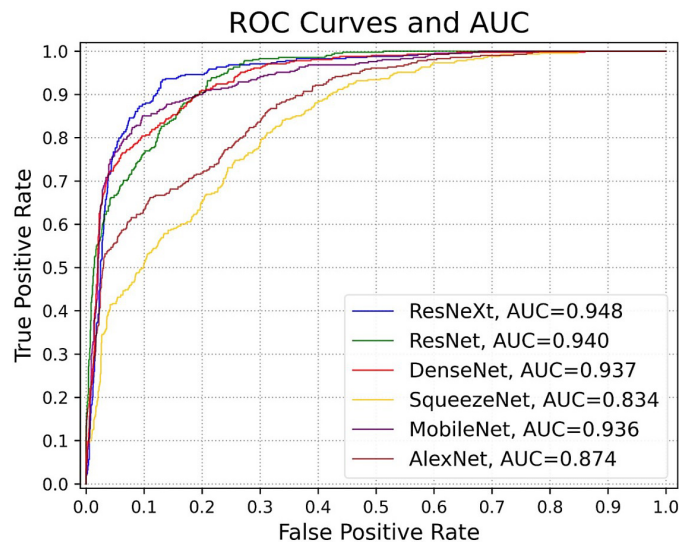**Figure 15.** Accuracy and loss curves for DenseNet architecture.

Figure 16. Area under curve and receiver operating characteristic for Resnet-50, ResNeXt, DenseNet, SqueezeNet, MobileNet_v2 and AlexNet models. AUC: Area under curve; ROC: receiver operating characteristic.

DECLARATIONS

Authors' contributions

Investigated the research area, reviewed and summarized the literature, wrote and edited the original draft: Chazhoor AAP

Managed the research activity planning and execution, contributed to the development of ideas according to the research aims: Ho ESL

Performed critical review, commentary and revision, funding acquisition: Gao B

Managed the research activity planning and execution, contributed to the development of ideas according to the research aims, funding acquisition, provided administrative: Woo WL

Availability of data and materials

The data can be found at http://wadaba.pcz.pl/. Emailing the creator by signing a consent form will give password access to the data[15]. The code has been uploaded to GitHub and the link is: https://github.com/ashys2012/plastic_wadaba/tree/main.

Conflicts of interest

All authors declared that there are no conflicts of interest.

Ethical approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Copyright

© The Author(s) 2022.

REFERENCES

1. Hiraga K, Taniguchi I, Yoshida S, Kimura Y, Oda K. Biodegradation of waste PET: a sustainable solution for dealing with plastic pollution. *EMBO Rep* 2019;20:e49365. DOI PubMed PMC
2. Alqattaf A. Plastic waste management: global facts, challenges and solutions. 2020 Second International Sustainability and Resilience Conference: Technology and Innovation in Building Designs(51154). 2020 Nov 11-12; Sakheer, Bahrain. IEEE; 2020. p. 1-7. DOI
3. Klemeš JJ, Fan YV. Plastic replacements: win or loss? 2020 5th International Conference on Smart and Sustainable Technologies (SpliTech). 2020 Sep 23-26; Split, Croatia. IEEE; 2020. p. 1-6. DOI
4. Backstrom J, Kumar N. Advancing the circular economy of plastics through eCommerce. Available from: https://hdl.handle.net/1721.1/130968 [Last accessed on 24 Jan 2022].
5. Joshi C, Browning S, Seay J. Combating plastic waste via Trash to Tank. *Nat Rev Earth Environ* 2020;1:142-142. DOI
6. Siddique R, Khatib J, Kaur I. Use of recycled plastic in concrete: a review. *Waste Manag* 2008;28:1835-52. DOI PubMed
7. Jiao W, Wang Q, Cheng Y, Zhang Y. End-to-end prediction of weld penetration: a deep learning and transfer learning based method. *J Manuf Process* 2021;63:191-7. DOI
8. Duan Q, Li J. Classification of common household plastic wastes combining multiple methods based on near-infrared spectroscopy. *ACS EST Eng* 2021;1:1065-73. DOI
9. Masoumi H, Safavi SM, Khani Z. Identification and classification of plastic resins using near infrared reflectance. *Int J Mech Ind Eng* 2012;6:213-20. DOI
10. Veerasingam S, Ranjani M, Venkatachalapathy R, et al. Contributions of Fourier transform infrared spectroscopy in microplastic pollution research: a review. *Crit Rev Environ Sci Technol* 2021;51:2681-743. DOI
11. Bruno EA. Automated sorting of plastics for recycling. Available from: https://www.semanticscholar.org/paper/Automated-Sorting-of-Plastics-for-Recycling-Edward-Bruno/e6e5110c06f67171409bab3b38f742db6dc110fc [Last accessed on 24 Jan 2022].
12. Alzubaidi L, Zhang J, Humaidi AJ, et al. Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. *J Big Data* 2021;8:53. DOI PubMed PMC
13. Albawi S, Mohammed TA, Al-Zawi S. Understanding of a convolutional neural network. 2017 International Conference on Engineering and Technology (ICET). 2017 Aug 21-23; Antalya, Turkey. IEEE;2017. p. 1-6. DOI
14. Xie L, Wang J, Wei Z, Wang M, Tian Q. Disturblabel: regularizing CNN on the loss layer. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016 Jun 27-30; Las Vegas, NV, USA. IEEE; 2016. p. 4753-62. DOI
15. Bobulski J, Piatkowski J. PET waste classification method and plastic waste DataBase - WaDaBa. In: Choraś M, Choraś RS, editors. Image processing and communications challenges 9. Cham: Springer International Publishing; 2018. p. 57-64. DOI
16. Bobulski J, Kubanek M. Waste classification system using image processing and convolutional neural networks. In: Rojas I, Joya G, Catala A, editors. Advances in computational intelligence. Cham: Springer International Publishing; 2019. p. 350-61. DOI
17. Agarwal S, Gudi R, Saxena P. One-Shot learning based classification for segregation of plastic waste. 2020 Digital Image Computing: Techniques and Applications (DICTA). 2020 Nov 29-2020 Dec 2; Melbourne, Australia. IEEE; 2020. p. 1-3. DOI
18. Chazhoor AAP, Zhu M, Ho ES, Gao B, Woo WL. Intelligent classification of different types of plastics using deep transfer learning. Available from: https://researchportal.northumbria.ac.uk/ws/portalfiles/portal/55869518/ROBOVIS_2021_33_CR.pdf [Last accessed on 24 Jan 2022].
19. Guo Y, Zhang L, Hu Y, He X, Gao J. MS-Celeb-1M: a dataset and benchmark for large-scale face recognition. In: Leibe B, Matas J, Sebe N, Welling M, editors. Computer Vision - ECCV 2016. Cham: Springer International Publishing; 2016. p. 87-102. DOI
20. Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. *Adv Neural Inf Process Syst* 2012;25:1097-105. DOI
21. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016 Jun 27-30; Las Vegas, NV, USA. IEEE; 2016. p. 770-8. DOI
22. Xie S, Girshick R, Dollár P, Tu Z, He K. Aggregated residual transformations for deep neural networks. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017 Jul 21-26; Honolulu, HI, USA. IEEE; 2017. p. 5987-95. DOI
23. Iandola FN, Han S, Moskewicz MW, Ashraf K, Dally WJ, Keutzer K. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size. Available from: https://arxiv.org/abs/1602.07360 [Last accessed on 24 Jan 2022].
24. Sandler M, Howard A, Zhu M, Zhmoginov A, Chen L-C. Mobilenetv2: Inverted residuals and linear bottlenecks. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018 Jun 18-23; Salt Lake City, UT, USA. IEEE; 2018. p. 4510-20. DOI
25. Huang G, Liu Z, Van Der Maaten L, Weinberger KQ. Densely connected convolutional networks. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017 Jul 21-26; Honolulu, HI, USA. IEEE; 2017. p. 2261-9. DOI
26. Tan C, Sun F, Kong T, Zhang W, Yang C, Liu C. A survey on deep transfer learning. In: Kůrková V, Manolopoulos Y, Hammer B, Iliadis L, Maglogiannis I, editors. Artificial neural networks and machine learning - ICANN 2018. Cham: Springer International Publishing; 2018. p. 270-9. DOI
27. Deng J, Dong W, Socher R, Li L-J, Li K, Fei-Fei L. Imagenet: a large-scale hierarchical image database. 2009 IEEE Conference on Computer Vision and Pattern Recognition. 2009 Jun 20-25; Miami, FL, USA. IEEE; 2009. p. 248-55. DOI
28. Brock A, Lim T, Ritchie JM, Weston N. Freezeout: accelerate training by progressively freezing layers. Available from: https://arxiv.org/abs/1706.04983 [Last accessed on 24 Jan 2022].

29.   Han X, Zhong Y, Cao L, Zhang L. Pre-trained AlexNet Architecture with pyramid pooling and supervision for high spatial resolution remote sensing image scene classification. *Remote Sensing* 2017;9:848. DOI

30.   Talo M. Convolutional neural networks for multi-class histopathology image classification. 2019. Available from: https://arxiv.org/ftp/arxiv/papers/1903/1903.10035.pdf [Last accessed on 24 Jan 2022].

31.   Go JH, Jan T, Mohanty M, Patel OP, Puthal D, Prasad M. Visualization approach for malware classification with ResNeXt. 2020 IEEE Congress on Evolutionary Computation (CEC). 2020 Jul 19-24; Glasgow, UK. IEEE; 2020. p. 1-7. DOI

32.   Seidaliyeva U, Akhmetov D, Ilipbayeva L, Matson ET. Real-time and accurate drone detection in a video with a static background. *Sensors (Basel)* 2020;20:3856. DOI PubMed PMC

33.   Nguyen THB, Park E, Cui X, Nguyen VH, Kim H. fPADnet: small and efficient convolutional neural network for presentation attack detection. *Sensors (Basel)* 2018;18:2532. DOI PubMed PMC

34.   Paszke A, Gross S, Chintala S, et al. Automatic differentiation in pytorch. Available from: https://openreview.net/pdf?id=BJJsrmfCZ [Last accessed on 24 Jan 2022].

35.   You K, Long M, Wang J, Jordan MI. How does learning rate decay help modern neural networks? Available from: https://arxiv.org/abs/1908.01878 [Last accessed on 24 Jan 2022].

36.   Li X, Chang D, Tian T, Cao J. Large-margin regularized Softmax cross-entropy loss. *IEEE Access* 2019;7:19572-8. DOI

37.   Ketkar N. Stochastic gradient descent. Deep learning with Python. Springer; 2017. p. 113-32. DOI

38.   Mukherjee H, Ghosh S, Dhar A, Obaidullah SM, Santosh KC, Roy K. Shallow convolutional neural network for COVID-19 outbreak screening using chest X-rays. *Cognit Comput* 2021. DOI PubMed PMC

39.   Selvik JT, Abrahamsen EB. On the meaning of accuracy and precision in a risk analysis context. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability* 2017;231:91-100. DOI

40.   Singh A, Príncipe JC. A loss function for classification based on a robust similarity metric. The 2010 International Joint Conference on Neural Networks (IJCNN). 2010 Jul 18-23; Barcelona, Spain. IEEE; 2010. p. 1-6. DOI

41.   Gao B, Bai L, Woo WL, Tian G. Thermography pattern analysis and separation. *Appl Phys Lett* 2014;104:251902. DOI

42.   Gao B, Zhang H, Woo WL, Tian GY, Bai L, Yin A. Smooth nonnegative matrix factorization for defect detection using microwave nondestructive testing and evaluation. *IEEE Trans Instrum Meas* 2014;63:923-34. DOI

43.   Ahmed J, Gao B, Woo WL, Zhu Y. Ensemble joint sparse low-rank matrix decomposition for thermography diagnosis system. *IEEE Trans Ind Electron* 2021;68:2648-58. DOI

44.   Song J, Gao B, Woo W, Tian G. Ensemble tensor decomposition for infrared thermography cracks detection system. *Infrared Physics & Technology* 2020;105:103203. DOI

45.   Ahmed J, Gao B, Woo WL. Sparse low-rank tensor decomposition for metal defect detection using thermographic imaging diagnostics. *IEEE Trans Ind Inf* 2021;17:1810-20. DOI

46.   Wu T, Gao B, Woo WL. Hierarchical low-rank and sparse tensor micro defects decomposition by electromagnetic thermography imaging system. *Philos Trans A Math Phys Eng Sci* 2020;378:20190584. DOI PubMed PMC