

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

# ScrapNet: An Efficient Approach to Trash Classification

**ABHISHEK MASAND<sup>1</sup>, SURYANSH CHAUHAN<sup>1</sup>, MAHESH JANGID<sup>1</sup>, RAJESH KUMAR<sup>2</sup> and SATYABRATA ROY<sup>1</sup>**

<sup>1</sup>Department of Computer Science and Engineering, Manipal University Jaipur, Rajasthan, India

<sup>2</sup>Department of Computer Science, College of Informatics, Bule Hura University, Ethiopia

Corresponding author: Rajesh Kumar (e-mail: rajesh.kumar@bhu.edu.et).

**ABSTRACT** As people have become more aware of their actions and how they affect their surroundings, they have realized the dire state of the environment. As a result, the recycling movement has gained momentum as a measure to save it. Contemporarily, the recycling industry has not seen a major shift and the problems that existed decades earlier persist. Trash classification is at the core of these problems, because if you can't classify it, obviously you can't recycle it. Manual classification often leads to misclassification as humans classify or judge things based on their experiences, knowledge, and not just absolute facts. Additionally, if the waste to be sorted is toxic, being in direct contact may be physically harmful to the people involved. Until a solution is found for this problem, the recycling industry won't be on par with the rise in recycling culture. Thus, this is the problem we have set out to solve; this paper proposes a Deep Learning model based on EfficientNet Architecture that can classify different kinds of trash with an accuracy high enough to make it a viable solution for the industry while using a comparatively lower number of parameters than existing methods. We achieved an accuracy of 98% on the TrashNet dataset, the standard data for Trash classification, and outperformed all existing models. Additionally, as no large dataset with a varied set of trash images was present, we created a new dataset of 8135 images by combining various datasets and standardized them, achieving a classification accuracy of 92.87% with EfficientNet B3.

**INDEX TERMS** EfficientNet, Recycle, ScrapNet, Trash, Trash Classification

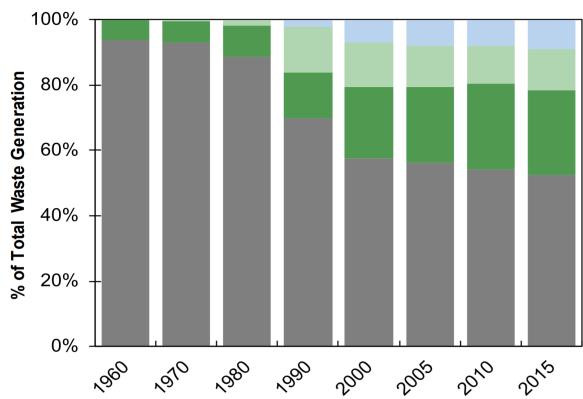
## I. INTRODUCTION

Annually, 2.01 billion tonnes of municipal solid waste is generated, and at least 33 percent, which is a conservative estimate by any means, is not managed correctly. The daily waste generated by a person per day averages 0.74 kilograms but ranges from 0.11 to 4.54 kilograms. Even though high-income countries only account for 16% of the world's population, they account for 34% of the waste. Also, waste collection differs across income levels, reflecting the consumption trends. High-income countries generate more dry waste that is easily recyclable like- paper, plastic, cardboard, metal, glass, etc. On the opposite end of the spectrum, the low-income countries generate more food and green waste and just 20% recyclable waste. There is a clear inverse relationship between organic waste produced and the level of economic development [11].

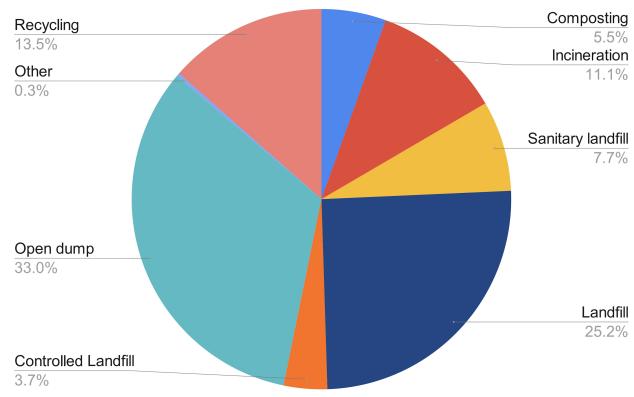
According to [11], Municipal Solid waste is expected to rise to 3.4 billion tonnes annually by 2050. The waste collection followed by recycling, combusting, or landfilling

is crucial for managing this rising threat. Still, only 48% of waste is collected in low-income countries, dropping to 26% outside urban areas. One of the main constituents of waste is plastic. Since the rise of plastic waste in the 1950s, only 9% has been recycled, 12% has been incinerated while leaving 79% of it in the environment. The rapid growth in plastic manufacturing has outpaced the rise of nearly every other man-made material. For comparison, half of all steel produced is used in construction, while half of all plastic manufactured becomes trash in less than a year.

A solution for this excessive waste is the need of the hour, and the long-term solution is recycling. As shown in Figure 1, we see that even though recycling has been on a rising trend, only 25.6% of the Municipal Solid Waste generated in the U.S. in 2015 was recycled [9]. The recycling industry faces several hurdles, which begin right at the source,i.e., the household. People often do not separate their waste which makes it a challenge to separate the recyclable material at the plant. This leads to a majority of the waste in the landfill,



**FIGURE 1.** Waste Management in the US through Recycling, Composting, Combustion with energy Recovery and Land-filling from top to bottom



**FIGURE 2.** How waste is treated globally through multiple processes(percentage distribution)

further aggravating the situation and increasing the pollution.

Even if a substantial part of this waste reaches recycling plants, the plants are neither accurate nor large enough to efficiently recycle the waste. They are heavily dependent upon manual labor that is riddled by problems like a lack of training, information retention, excessive downtime, inability to distinguish materials, etc. and slows down the throughput of the plant [12]. Globally, less than 14% of the total waste is recycled, and almost one-fourth of the waste is thrown in uncontrolled landfills(Figure 2). It is estimated that 1.6 billion tonnes of Carbon dioxide (5% of Global emission) was generated due to disposing of waste in open dumps and uncontrolled landfills. This number is expected to rise to 2.38 billion tonnes by 2050 if nothing changes.

The current COVID-19 pandemic has only worsened the situation with the astronomical rise in medical waste. With the advancement in technology, techniques like Deep Learning have gained enough robustness to be implemented in real-life situations. Many papers have been proposed to solve the problem of waste classification, but the majority of the experiments have been done on the TrashNet dataset. Albeit being the most frequently used waste dataset, it has a small size and variety. The other datasets currently being used to

classify waste were too specific in their use cases to be used in the industry on a large scale. Another drawback of the majority of existing research is that they achieved their highest accuracy through large networks like ResNet, DenseNet, and InceptionNet, and would be harder to implement in smaller devices. As quite evident, there is a lot of scope for improvement in this field. This paper presents a modern approach, one that implements and compares the latest Deep Learning neural networks to aid the struggling recycling industry. We suggest a deep neural network for object detection for multi-object scenes to separate trash and then a different deep neural network that classifies trash based on their material and composition into Seven categories. Finally, we suggest using a similar network to further sub-classify the plastic category as recyclable or non-recyclable based on popular guidelines. Our major contributions through this paper are as following-

- 1) Creation of a new diverse dataset, which is larger than any pre-existing waste dataset, and can be used as the new standard for waste classification.
- 2) Comparison and analysis of various deep learning architectures for item detection of objects on the novel dataset to achieve superior accuracy.
- 3) Modifying the deep neural network used on our data to classify images of TrashNet dataset and achieve the highest accuracy yet.
- 4) Reclassifying plastic objects using deep neural network as recyclable or non-recyclable based on UK recycling guidelines.

The rest of the paper is distributed as following - Section 2 covers the previous related works in waste classification through machine learning and deep learning approaches, Section 3 covers the creation of a new dataset called ScrapNet and its distribution. Section 4 covers the methodology, the architectures, and the experiments performed, while Section 5 discusses the results of these experiments. Section 6 concludes the paper while comparing the accuracy attained on TrashNet.

## II. RELATED WORKS

Even though this field is still quite undeveloped with considerably less research on the use of advanced technology like deep learning, the following studies have been conducted and are relevant to our research:

- Authors Mingxing Tan et. al. [19], proposed a weighted Bi-directional Feature Pyramid Network(BiFPN) along with a compound scaling method to improve efficiency for object detection networks. Based on this, they developed a new family of object detectors called Efficient-Det, which achieve much higher efficiency than existing networks across a wide range of constraints.
- Authors Mindy Yang et al. [24], collected images of single pieces of recycling or garbage and classified them into six classes: glass, paper, metal, plastic, cardboard, and trash. The data was hand-collected and consisted

of 400-500 images for each image class. SVM with SIFT and CNN were implemented and compared for this classification task. They achieved an accuracy of 63% and 22%, respectively.

- In another study, Yinghao Chu et al. [6] proposed a Multi-layer Hybrid deep-learning System (MHS) to sort waste automatically. A high-resolution camera with sensors was used to extract image features and other features. The MHS used a CNN and a multi-layer perceptron (MLP) to combine the image with features and classify wastes as recyclable or not. They achieved an accuracy higher than 90% under two different testing scenarios.
- Authors Olugboja Adedeji et al. [1] proposed an intelligent waste classification system that uses ResNet-50 CNN to extract the features and SVM to classify them into different categories such as glass, metal, paper, etc. The trash image dataset created by [24] was used and, an accuracy of 87% was achieved.
- A more targeted approach was taken by Joao Sousa et al. [17], they explored the wastage of food in food trays and created a new dataset called Labeled waste in the wild. Based on this dataset, they proposed a hierarchical deep learning model for waste detection and classification in food trays. Faster R-CNN was used to create bounding boxes to crop the images, and 7 independent CNN's were used to classify the different objects like cutlery, Box, Bottle, etc. The proposed model achieved an accuracy of 86%.
- In another research, the authors Anh H Vo et al. [20] collected 5904 images belonging to three classes- Organic, Inorganic, and Medical waste to create a new dataset called VN-trash dataset. They developed a deep neural network model for trash classification named DNN-TC that outperformed the existing ResNext model and achieved an accuracy of 94% and 98% for TrashNet and VN-trash dataset, respectively.
- VGG, Inception, and ResNet architectures were implemented by Victoria Ruiz et al. (AIWC) [15]. They compared these deep learning architectures by classifying images of the TrashNet dataset into different categories. They implemented a combined Inception-ResNet model that outperformed others and achieved an accuracy of 88.6%.
- Several deep CNN architectures were implemented and compared by the authors Cenk Bircanoglu et al. [4] to find the best one. They compared several models without pre-trained weights and found that Inception-ResNet, Inception-v4 outperformed other networks with a 90% accuracy. Further, they also implemented transfer learning and fine-tuning of weight parameters using ImageNet. They were able to achieve an accuracy of 95% with DenseNet121.
- Authors Rahmi Arda Aral et al. [3] tested several deep learning models on the TrashNet Dataset. These included the DenseNet121, DenseNet169, MobileNet, In-

**TABLE 1.** Summary of related works.

Authors	Problem	Solution
Mindy Yang et al.	Limited Dataset, low accuracy	Novel model
Yinghao Chu et al.	Dataset of only 50 items	New dataset
Olugboja Adedeji et al.	Limited Dataset	New dataset
Joao Sousa et al.	Limited to Foodtray data	New dataset
Anh H Vo et al.	Limited Dataset	New dataset
Victoria Ruiz et al.	Limited Dataset, low accuracy	Novel model
Cenk Bircanoglu et al.	Limited Dataset	New dataset
Rahmi Arda Aral et al.	Limited Dataset	New dataset

**TABLE 2.** Information about datasets combined to form ScrapNet.

Name	Images	Categories
TrashNet	2527	6
Openrecycle	2301	7
TACO	1501	59
Waste Classification	850	1

ceptionResNetV2, Xception architectures. The accuracy achieved by DenseNet121 was 95%, followed closely by InceptionResNetV2 that achieved an accuracy of 94% after fine-tuning.

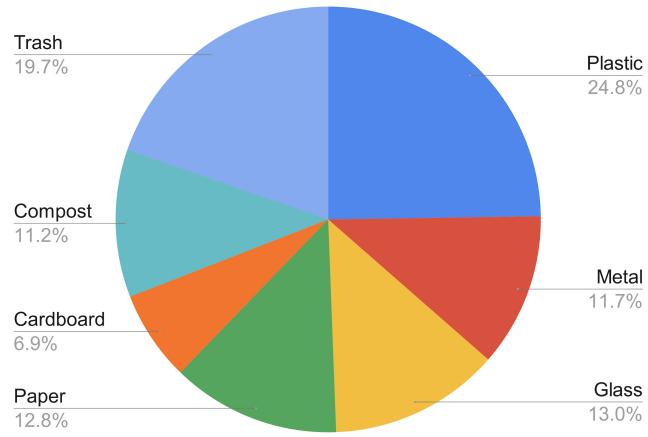
The related works have been summarized in Table 1. The drawbacks of the works along with how the paper aims to solve them has been highlighted as well.

### III. DATA

Most research in this domain is done either on datasets that are very particular, like the Labelled waste in the wild [17] dataset which had images of left-over waste on Food Tray, or they were simulated and captured in a controlled environment with limited variety, like the dataset used in [6] and lastly, many datasets focus on limited classification categories like the VN-trash dataset [20]. As a result, Trash classification models are rarely generalized and only distinguish a specific kind of waste. As a result, these models do not represent the actual waste that required sorting. To overcome this problem, we created our dataset that would cover a broad range of categories and simulate real-life waste in a better way. We unified and standardized multiple datasets into one dataset through filtering and extraction of images in part to create the ScrapNet dataset. This dataset is used in all our experiments along with TrashNet, to provide a comparison. We hope to make this dataset available for the scientific community to use for further research.

**TABLE 3.** Distribution of data in ScrapNet.

Category	Number of images
Plastic	2014
Metal	950
Glass	1055
Paper	1043
Compost	913
Trash	1602
Total	8135



**FIGURE 3.** Distribution of images over 7 categories(Plastic, Trash, Glass, Paper, Metal, Compost, Cardboard) in our new dataset

All the datasets used are listed in Table 2 along with the number of images classification categories.

All images from TrashNet [24] and Openrecycle [7] were taken as-is and filtered manually at the annotator's discretion to remove erroneous and unrecognizable images. The images from TACO [14] contained several objects in an image, and these objects were separated through bounding boxes and then extracted to create several images. During data creation, it was noticed that images were scarce in the compost category, and to deal with this 850 representative compost images were extracted from the Waste Classification data [16] available on Kaggle.

To account for the discrepancy in the classification categories, the images were classified by 2 annotators with domain knowledge with an overlap rate of 99% to ensure proper classification. The segregated images accounted for the material of the objects, and the possibility of recycling them as classification of trash does not have clear boundaries. For example, one might assume that paper cups belong in the paper category, but for recycling purposes, it does not belong there as they have a thin lining of plastic that prevents water from leaking also prevents them from being recycled.

Images in the following Six categories – Plastic, Metal, Glass, Paper, Cardboard, Compost, are classified as belonging to the particular material class and being recognized as recyclable. On the other hand, objects in the Trash category are objects that are either in poor condition or made of a combination of materials, which renders them unrecyclable through most simple processes. The distribution of data is in Figure 3.

Lastly, while these Four datasets combined had only 7179 images distributed over a varying set of categories, ScrapNet has a total of 8135 images distributed over Seven categories. This increase of 956 images was achieved through multiple object extraction from the TACO dataset. The data distribution is in Table 3.

## IV. METHODS

In this paper, we experimented with, compared, and analyzed the implementation of different CNN architectures on our dataset. Experimentation with multiple models allowed us to analyze the results, and determine the level of augmentation needed. Also, it led to the crystallization of complex relations within the data that were not visible before. The overall objective of analyzing different models was to understand the performance-accuracy trade-offs between the models.

### A. COMPARISON OF DIFFERENT NETWORKS

We explored multiple neural network architectures – ResNet [10], ResNext [23], and EfficientNet [18], along with their different sub-architectures. The deep learning models currently used for Trash Classification implement ResNext101 and ResNext152, which are relatively huge compared to EfficientNet. EfficientNet consistently reduces parameters by order of a magnitude of 8.4x and increases speed by 16x Floating Point Operations(FLOPS) when compared to the current state-of-the-art models. This increased speed is necessary to apply deep learning models in real-time applications where efficiency is crucial for success. Moreover, bigger CNN models are harder to train and struggle to operate on edge-computing applications.

As a baseline experiment, we used the ResNet50 model. Although not a very large model, it is still considered one of the leading models to achieve speed and accuracy in image classification tasks.

### B. ARCHITECTURES

#### 1) EfficientNet and EfficientDet Architecture

EfficientNet [18] Architecture was introduced in 2019 at the International Conference on Machine Learning (ICML) by Authors Mingxing Tan and Quoc V. Le. Constructed on top of the idea that balancing the network depth, image resolution, and width can increase performance. To bring this idea to life, they introduced the concept of compound scaling through EfficientNet. Compound scaling uses a coefficient to scale up the width, image resolution, and depth uniformly.

The authors propose the following compound scaling parameters and constraints:

$$\begin{aligned}
 \text{depth} : d &= \alpha^\phi \\
 \text{width} : w &= \beta^\phi \\
 \text{resolution} : r &= \gamma^\phi \\
 \text{s.t. } \alpha \cdot \beta^2 \cdot \gamma^2 &\geq 2 \\
 \alpha \geq 1, \beta \geq 1, \gamma \geq 1
 \end{aligned} \tag{1}$$

$\phi$  is the user specified coefficient used to control the Floating Point Operations (FLOPs) for constructing the model.

$\alpha, \beta, \gamma$  are the parameters used for distributing the depth, width and image resolution.

The FLOPs required for a convolutional operation are directly proportional to  $d$ ,  $w^2$  and  $r^2$  as represented in Equation 1. The authors restricted the value of the parameters of

$\alpha, \beta^2, \gamma^2$  to 2 so that with every new  $\phi$ , the FLOPs for the new model goes up by  $2^\phi$ .

The architecture of the base EfficientNet model: EfficientNet B0, was found by manipulating the depth, width, and resolution parameters to maximize accuracy while minimizing the FLOPs. The search space is the same as Tan et al., 2019 and as a result, the EfficientNet B0 is very similar to the MnasNet [3]. This network was used as a baseline to construct the large networks in the EfficientNet series.

The same concept was further propagated into the EfficientDet [19] Architecture. Compound Scaling was applied to the BiFPN Network to scale alongside the EfficientNet backbone and provide a faster convergence time.

The experiments conducted by the authors revealed that for the same FLOPs, the accuracy of EfficientNet was higher than any existing architecture. Additionally, it provided a faster convergence along with better performance in computer vision tasks [18].

### C. TRANSFER LEARNING

When all the parameters of a neural network are learned from scratch, the weights of the neural network get initialized randomly, and as a result, the model learns very slowly and often has to be trained for a long time to reach convergence. Instead, a concept known as Transfer learning [13] is used quite often. Generally, transfer learning is used for very large networks that have been pre-trained on a vast dataset. As a result, minimal training is required for achieving state-of-the-art accuracy, even with a small amount of data. Transfer learning can either be done by retraining the pre-existing layers in the model or adding new head layers to the end of the model and training either the newly added layers or all of them.

We applied transfer learning by using the architectures above as the base model and building upon them by adding new head layers. Experiments on all the architectures have been done using the widely popular weights trained on ImageNet, but EfficientNet models have also been trained using the Noisy Student Weights [22] which achieves 88.4% at the top-1 accuracy on ImageNet, which outperforms the previous state-of-the-art model. Also, we have experimented with ResNet and ResNext from scratch but observed that applying Transfer learning using pre-trained weights provided better results.

### D. EXPERIMENTAL SETUP

The experiments were performed on a computer system having an eight-core CPU with 32 GB RAM, GTX 1080 GPU running Ubuntu 20.04 operating system. The deep neural network model was coded in Python using Pytorch 1.8.0 and CUDA 10.2. The assumption that pre-processing tasks like normalization and reshaping of data would not modify the informational integrity of the data has been taken.

Experiments were performed to investigate the effect of different network architectures, optimizers, and transfer of weights. The first phase was to find the best architecture

**TABLE 4.** Comparison of RMSProp, Adagrad, SGD, Adam optimizers across different networks

Network	RMSProp	Adagrad	SGD	Adam
EfficientNet B0	0.92	0.61	0.63	0.92
EfficientNet B1	0.91	0.81	0.82	0.90
EfficientNet B2	0.91	0.87	0.84	0.89
EfficientNet B3	0.92	0.86	0.84	0.89
EfficientNet B4	0.88	0.81	0.82	0.88
EfficientNet B5	0.92	0.78	0.82	0.88
EfficientNet B6	0.86	0.83	0.84	0.86
EfficientNet B7	0.89	0.81	0.83	0.88
Resnet50	0.83	0.83	0.81	0.79
Resnet101	0.81	0.78	0.78	0.75
Resnext101	0.81	0.75	0.81	0.78
Resnet152	0.79	0.75	0.78	0.77

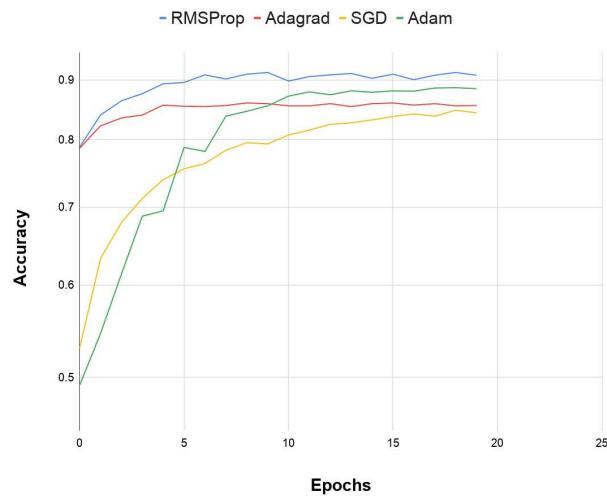
for the dataset. From there on, the next problem we tackled was deducing the best sub-architecture. For example, in the EfficientNet architecture, the sub-architecture would include the B0, B1 ... B7. In this phase, we also compared the performance of pre-trained weights while deducing the sub-architectures. Finally, we compared the performance of different optimizers on the dataset. Each optimizer has a different set of parameters attached to it, the optimizer parameters were set to their default values or as suggested by the authors of the architectures. A total of 22 different experiments were conducted on the dataset. A combination of metrics, Precision, Recall, F1-Score, and Accuracy, of the validation set, was used for evaluating the different architectures. Each network was trained with a batch size of 4 and gradient accumulation of batch size 16.

### E. TRAINING AND EXPERIMENTS

The primary objective was to make sure we had objects separated in images with multiple objects. Without having separated objects, any model would not be able to adequately determine the class. We looked into the state-of-the-art object detection models and came across EfficientDet, a recently released family of object detectors that outperforms its competitors by a significant margin. Afterward, we performed tests by constructing different models using ResNet, ResNext, and EfficientNet architectures as the base, followed by additional layers. We also conducted tests using two different ways, either by initializing the base model from scratch or using pre-trained weights.

When trained from scratch, the weights for all the layers in the architecture are randomly initialized using a gaussian distribution. Meanwhile, when training using pre-trained weights, the three architectures are initialized using pre-trained weights, and the new fully connected layers are randomly initialized and finally trained on the new categories of our dataset.

Initially, all the EfficientNet models, along with Resnet and Resnext models were trained using all 4 optimizers to understand the performance differences as seen through Table 4. The optimizer comparison for EfficientNet B3 has been visualized in Figure 4 to see the change in accuracy with increasing Epochs. During our testing, the RMSProp



**FIGURE 4.** Comparison of different Optimizer accuracy vs Epochs on EfficientNet B3(best-performing) network

optimizer achieved the highest accuracy, followed by Adam, Adagrad, and SGD. While Adagrad converged the fastest at four epochs, RMSProp, SGD, and Adam converged much later at 7-15 epochs. Even though the Adagrad optimizer converged earlier than RMSProp, it had a much lower accuracy, and as a result, we decided to go forward with the RMSProp optimizer.

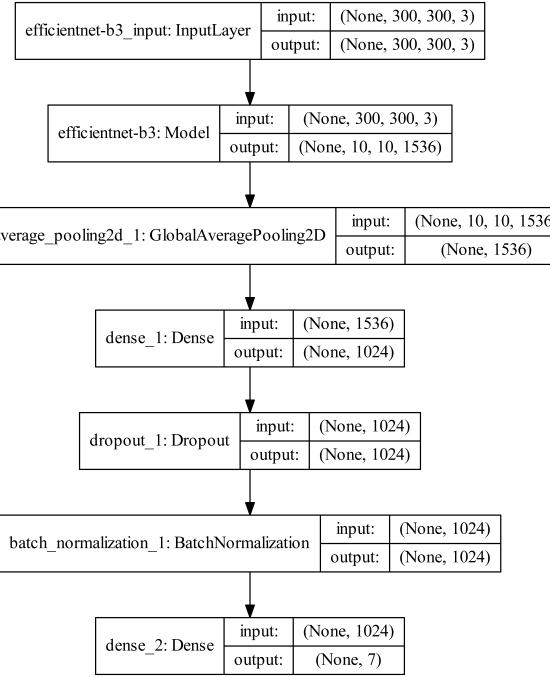
Weight Standardization(WS) [21] was implemented as part of the Convolution Layer to improve results. WS standardizes the weights in the Convolution Layer so that they have zero mean and unit variance. This reduces the Lipschitz constants of loss and smoothes out the landscape. WS has been tested out on computer vision tasks and has shown significant improvement in image classification and object detection tasks among many others.

$$W' = \frac{W - \bar{W}}{\sqrt{\frac{\|W - \bar{W}\|^2}{n}}}, \quad (2)$$

where  $W'$  denotes the modified weight,  $W$  denotes the original weights,  $n$  is the number of weights.

The models combined with the EfficientNet architecture were tested using both methods. When the models were trained from scratch, the weights of the model were initialized from random Gaussian distribution. The model was trained for up to 50 epochs using RMSProp Optimizer at a learning rate of 0.0001. For fine-tuning the model, early stopping and learning rate Scheduler have also been applied by reducing the learning rate by a factor of 2 every time, along with reducing the patience for Early Stopping [5] by a factor of 2.

In the last fully-connected layer, the softmax function is used in order to calculate the confidence of each prediction as follows:



**FIGURE 5.** Network Architecture of the modified EfficientNet B3 implemented

$$y_i = \frac{e_i^x}{\sum_{j=1}^n e_j^x}, \quad (3)$$

where  $y_i$  denotes the probability of each class,  $n$  is the number of classes,  $x_i$  and  $x_j$  are the inputs.

The architecture of our modified EfficientNet B3 is in Figure 5. This architecture is the general representation of all models used in our experiments. Only the model layer varied depending on which model was being implemented.

All the three architectures use the image sizes they were primarily trained on for the results, i.e., 224x224 was used for ResNet50 and 380x380 for the EfficientNet B4. According to our experiments, the pre-trained models perform better when retrained using original image sizes. The EfficientNet models performed about 2% lower with the image size of 224x224 than the experiments performed with their original image size.

The model's standard naming is used to refer to the models initialized with random weights, like – “EfficientNet B0” while the models trained with pre-trained weights are referred to with a suffix “PW” which stands for “Pre-trained weights”, like – “EfficientNet B0 – PW”.

The ResNet models on average took a longer time to converge than the EfficientNet models with about 7-10 epochs more on average as compared to the much smaller EfficientNet architecture. EfficientNet models converged between 4 to 7 epochs, while the ResNet models took 14 epochs. The ResNet models had lower performance than all the EfficientNet models as seen in Figure 6.

As another validating parameter for our model, we tested it on the TrashNet dataset. For this, our data was split into

80/10/10 as training, validation, and testing datasets. Also, we understood the possibility of a fluke where the entire TrashNet data might end in the training set, and, as a safeguard, we split the TrashNet data within our larger dataset into 70/15/15 as well. Moreover, we implemented Transfer Learning to use the weights found during testing on our primary dataset. These weights were mapped onto all layers except the final layer due to the absence of the compost category in TrashNet.

We have also implemented the concept of Adaptive Gradient Clipping. Gradient clipping is used to clip the size of gradients to ensure optimization of deep neural networks in areas with a higher loss. Adaptive Gradient clipping removes the need for hand-tuning the clipping parameters and allowed us to train our model with larger batch sizes and better data augmentations.

The process flow used on the ScrapNet Dataset to detect if a plastic item is recyclable or not has been summarized in Figure 7. And the process flow for Trashnet is summarized in Figure 8. A subset of TrashNet is separated to use for testing, and the rest of TrashNet, along with OpenRecycle, TACO, and Waste Classification datasets were combined as explained previously and called ScrapNet. The dataset was split into Training, Test sets, and the ScrapNet Model is trained on the training set. The ScrapNet model uses Conv and WSConv layer, followed by a Global Average Pooling and then BatchNorm and Dropout to get a 7 class classifier. AGC, Gradient Accumulation, and RandAugment were also used to achieve state-of-the-art accuracy with ScrapNet. Finally, as a comparative measure, we replaced the 7-class classifier layer with a 6-class classifier layer. Then the TrashNet Test set was fed to the model and successfully achieve the highest accuracy on TrashNet.

The Plastic category in particular was chosen to be sub-classified. The main reason being that Plastic objects are recyclable only if they belong to a certain sub-group of plastic, and even then, specialized equipment is often needed to recycle them. The driving factor behind this decision was that plastic is still one of the most major waste components and one of the toughest to sort and recycle due to the complexity of the material. Traditionally, plastic is divided into seven categories, and some of these are recyclable while others are non-recyclable. This leads to a lot of confusion in manual sorting, eventually leading to a lot of recyclable plastic being thrown out instead [2].

The UK Sustainability guide [8] was used as a baseline for the division of the TACO Dataset into Recyclable and Non-recyclable objects. The EfficientNet B3 network, which performed the best for 7 class Trash classification, was trained on the TACO Dataset's plastic category, and then further used to classify the entire plastic category of ScrapNet. The division for Recyclable and Non-recyclable Objects in the Taco Dataset is explained in Table 5.

**TABLE 5.** Division of Plastic Category on the basis of Recyclability.

Plastic Subcategory	Recyclability
Six Pack rings	Recyclable
Garbage bag	Recyclable
Single-use carrier bag	Recyclable
Polypropylene bag	Recyclable
Spread Tub	Recyclable
Tupperware	Recyclable
Plastified Paper bag	Non-recyclable
Plastic film	Non-recyclable
Other Plastic wrapper	Non-recyclable
Crisp Packet	Non-recyclable
Disposable food container	Non-recyclable
Foam food container	Non-recyclable
Other plastic container	Non-recyclable
Plastic gloves	Non-recyclable
Plastic utensils	Non-recyclable

**TABLE 6.** Performance of various experiments performed on our data.

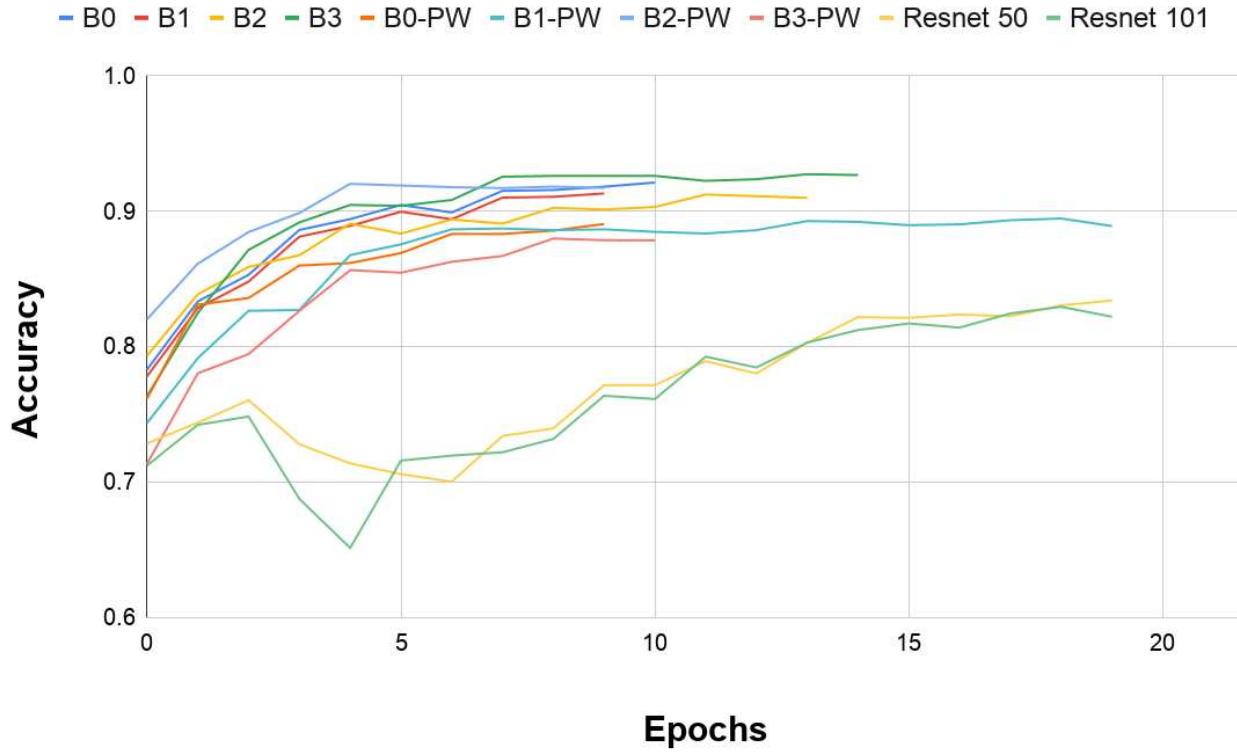
Model	Accuracy	Precision	Recall	F1-score
ResNet50 (Baseline)	0.8311	0.83	0.83	0.83
ResNet101	0.8121	0.81	0.81	0.81
ResNet152	0.7902	0.78	0.77	0.77
ResNext101	0.8109	0.81	0.81	0.81
EfficientNet B0 - PW	0.8901	0.89	0.89	0.89
EfficientNet B1 - PW	0.8907	0.82	0.82	0.82
EfficientNet B2 - PW	0.9171	0.92	0.92	0.92
EfficientNet B3 - PW	0.8784	0.88	0.88	0.88
EfficientNet B4 - PW	0.8717	0.87	0.87	0.87
EfficientNet B5 - PW	0.8784	0.88	0.88	0.88
EfficientNet B6 - PW	0.8483	0.84	0.84	0.84
EfficientNet B7 - PW	0.8228	0.82	0.82	0.82
EfficientNet B0	0.9208	0.92	0.92	0.92
EfficientNet B1	0.9122	0.91	0.91	0.91
EfficientNet B2	0.9103	0.91	0.91	0.91
EfficientNet B3	0.9287	0.93	0.93	0.93
EfficientNet B4	0.8821	0.88	0.88	0.88
EfficientNet B5	0.9214	0.92	0.92	0.92
EfficientNet B6	0.8600	0.86	0.86	0.86
EfficientNet B7	0.8925	0.89	0.89	0.89

## V. RESULTS AND DISCUSSION

The EfficientDet B3 achieved the highest mAP in our testing of 82.34% and was used for creating bounding boxes and separating multiple objects in a scene. The results of the trash classification experimentation are shown in Table 6. The Precision, Recall, and F1-Score have been computed using a weighted average. The experiments used augmented data to increase the heterogeneity of our dataset.

The experiments were performed with 5 runs for each model and the mean accuracy has been considered as the primary data point going forwards. The experiments are summarised in Table 7 and Figure 9 with the Standard deviation, Minimum and Maximum accuracy received in the 5 runs for each of the model.

ResNet50 was chosen as the baseline model because it is one of the most popular models for image classification. While being relatively easy to implement, it achieved a decent accuracy of 83.11% on our dataset. In the world of CNN, ResNet50 is still a relatively small model, and we decided to implement larger models with higher complexity. ResNet101 and ResNet152 were chosen. A fall of 1.9% was observed

**FIGURE 6.** Validation accuracy of models over time until convergence (based on validation loss)

on dropping to ResNet101, and the accuracy decreased even further by 2.1% on ResNet152. Then, ResNext101 was implemented, but the results were close to ResNet101, and the accuracy was lower than that of ResNet50 and ResNet101.

While researching different models and techniques to improve the Trash classification, a new deep neural network architecture released by Google called EfficientNet was discovered. The team at Google created Eight models based on this architecture, and released them in 2019, ranging from EfficientNet B0 to EfficientNet B7 with an increasing depth and parameters. The architecture claimed to outperform all existing networks on ImageNet. As the existing results weren't at par even with fine-tuning and optimization, it was decided to experiment with the EfficientNet architecture.

Firstly, EfficientNet B0 was implemented, as it is the base EfficientNet model, and it outperformed ResNet101 by an astounding 6% and 9% for the pre-trained and non-pre-trained versions, respectively. It became our top-performing model yet. With this positive result, we decided to implement and fine-tune all the other EfficientNet models as well.

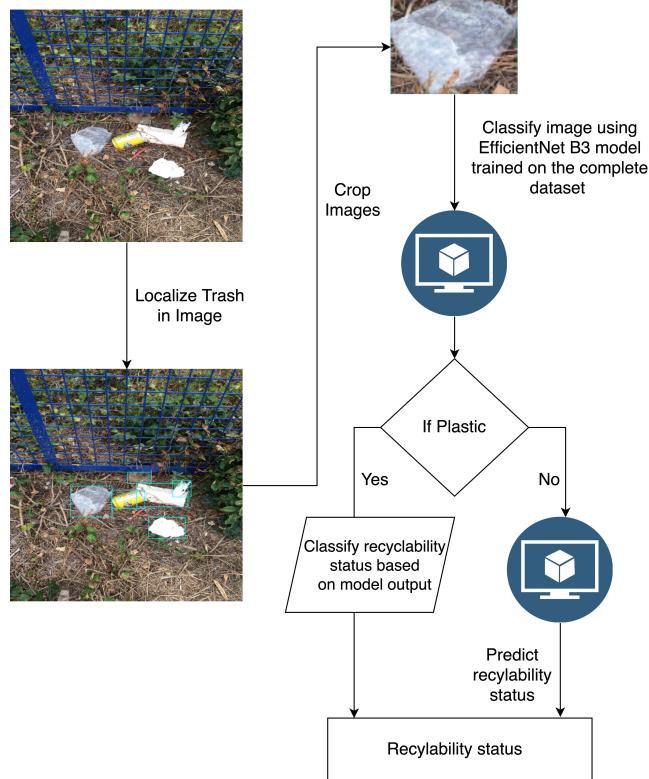
Through the experimentation process, we observed that, for the pre-trained versions of the models, the accuracy increased from EfficientNet B0 to B2 and peaked at B2 before declining substantially from B3 to B7. For the non-pre-trained versions, the accuracy showed erratic behavior and no consistent pattern, but the best accuracy was achieved

**TABLE 7.** Granular performance metric for the models across multiple runs visualized through Figure 9

Model	Average	Minimum	Maximum	STD
ResNet50 (Baseline)	0.8311	0.81	0.85	0.014
ResNet101	0.8121	0.80	0.83	0.010
ResNet152	0.7902	0.77	0.80	0.012
ResNext101	0.8109	0.80	0.82	0.010
EfficientNet B0 - PW	0.8901	0.87	0.92	0.017
EfficientNet B1 - PW	0.8907	0.88	0.90	0.009
EfficientNet B2 - PW	0.9171	0.90	0.92	0.007
EfficientNet B3 - PW	0.8784	0.86	0.91	0.017
EfficientNet B4 - PW	0.8717	0.86	0.89	0.011
EfficientNet B5 - PW	0.8784	0.86	0.90	0.019
EfficientNet B6 - PW	0.8483	0.82	0.86	0.017
EfficientNet B7 - PW	0.8228	0.81	0.84	0.015
EfficientNet B0	0.9208	0.89	0.94	0.015
EfficientNet B1	0.9122	0.88	0.93	0.017
EfficientNet B2	0.9103	0.88	0.93	0.020
EfficientNet B3	0.9287	0.91	0.93	0.006
EfficientNet B4	0.8821	0.86	0.89	0.008
EfficientNet B5	0.9214	0.90	0.94	0.014
EfficientNet B6	0.8600	0.83	0.88	0.019
EfficientNet B7	0.8925	0.87	0.91	0.013

on the non-pre-trained versions of B0, B3 and B5. B5 has not been chosen for comparison because of its much larger size, and no real benefit on top of the smaller networks. After further fine-tuning and training, the best accuracy achieved was 92.87% by EfficientNet B3.

Additionally, we achieved an accuracy of 89.21% with



**FIGURE 7.** ScrapNet paper process flow from localizing Trash in images to eventually predicting the recyclability status of the Trash

EfficientNet B3 for the final sub-classification of Plastic as Recyclable and non-recyclable.

The non-pre-trained weight version of the models had a higher accuracy compared to their pre-trained counterparts in all cases except one(B2). And the same was true for our top-performing model, where B3 beat B3-PW by almost 5%, as seen in Figure 10.

During the testing and comparison of the models, some interesting patterns were observed and the same have been discussed further. Referring to Table 6, we noticed that EfficientNet B0 & B3 had a similar F1 score and an accuracy close to our top model, and we decided to take a look at the confusion matrix to find where B0 might be lacking. Figure 11 and 12 show the confusion matrix for B0 and B3 respectively. We can observe the following things:

- 1) B0 performed better than B3 in the paper category by 3.3%, in metal by 0.5% and in compost, by 2.7% while performing similar to B3 or only marginally lower in the other four categories. This means that B0, which has less than half of the parameters of B3, can perform similarly. Additionally, it classifies better where Paper, metal, or compost might be the priority or the major category. Its small size also enables quick training when new data is added.
- 2) B3, on the other hand, performed better than B0 in the trash category by 1.5%, in plastic by 4.2% and glass by 1.4% category while performing similarly to

**TABLE 8.** Performance of related works on TrashNet.

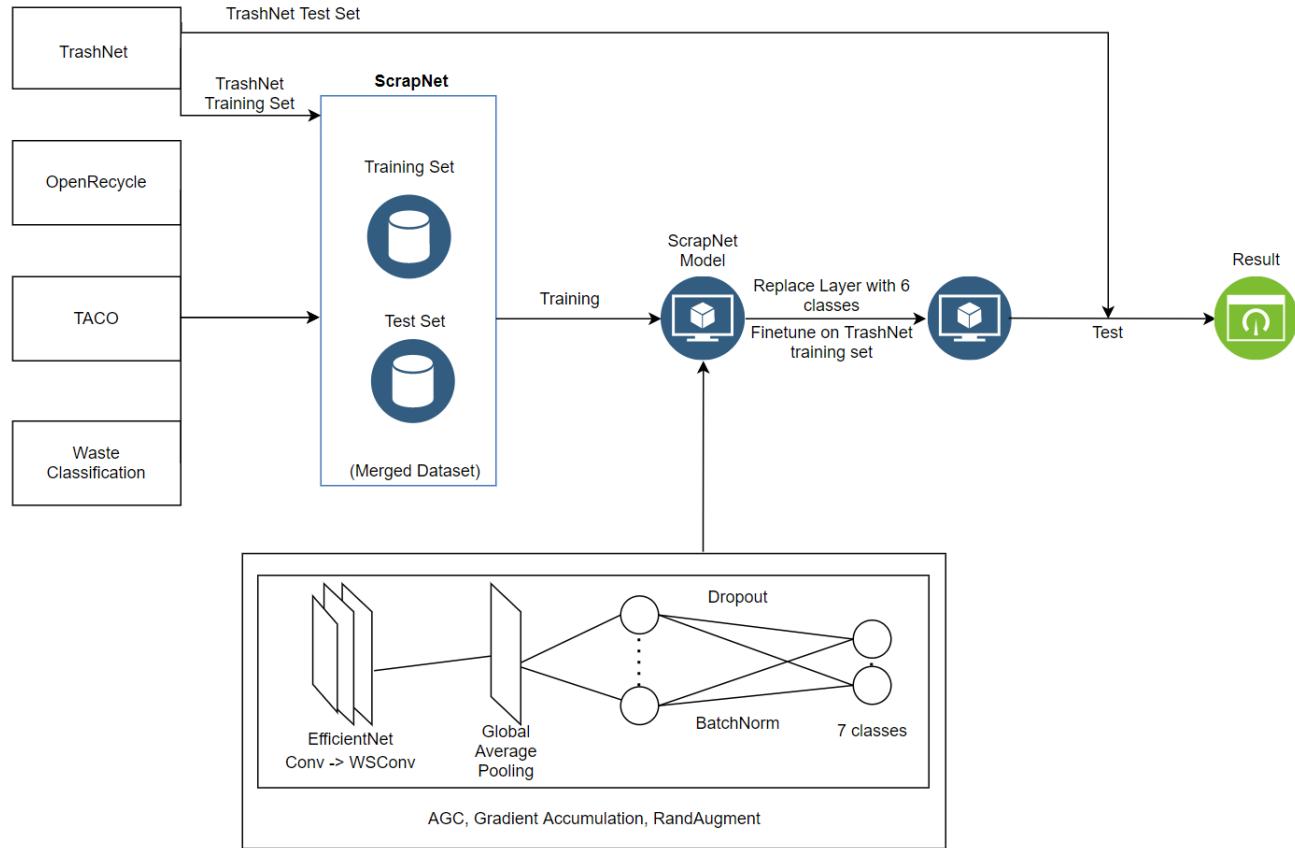
Paper Author	Model	Accuracy
M. Yang et al.	SVM	0.63
C. Bircanoglu et al.	RecycleNet	0.81
O. Adedeji et al.	ResNet50 + SVM	0.87
V. Ruiz et al.	ResNet	0.88
A. H Vo et al.	DNN-TC	0.94
R. Arda Aral et al.	DenseNet121	0.95
Ours	ScrapNet	0.98

B0 or marginally lower in the other four categories. Even though B3 has a higher number of parameters, it's useful due to its higher overall accuracy, especially in the above three categories. Moreover, the accuracy gap might even increase as the model trains on more data over time.

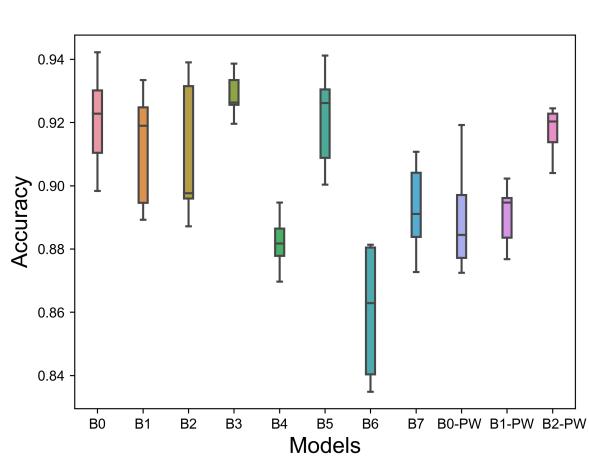
This observation led to another use case for EfficientNet B0, it could be used in smaller devices as it works on fewer parameters and would require lesser computing power. On the other hand, B3 can be used in places where accuracy is a priority without regard for computation power. The final EfficientNet B0 and B3 networks were trained using RMSProp optimizer with a learning rate of 0.0001 and a Batch size of 4 and gradient accumulation of batch size 16 on a Training set the size of 6840 images and tested on the Testing dataset of 1260 images. The callbacks used in the model were Early Stopping and reduce learning rate on the plateau of 5 epochs based on validation loss by a factor of 0.5 and a minimum learning rate of 0.41. Finally, stochastic weight-averaging was used with a starting period of 5 epochs and averaging period of 3 epochs.

The standard trash data currently used is the TrashNet dataset. To get comparative results for our models, we put our model against all other models we found in our research and literature review in Table 8 plotted in Figure 13. The model was trained on ScrapNet with seven categories, and transfer Learning through substitution of the final seven-element classifier with a six-element classifier for classifying TrashNet was used. The model achieved an astounding accurate of 98.4%, and in our research, this was the highest accuracy ever achieved on TrashNet. The model outperformed modified SVM's, ResNets, and DenseNets by a margin of 3.5%. TrashNet is quite limited in its categories, size and as the size and variety increased, these models would improve substantially, and the ScrapNet dataset is a step in that direction. As the research community starts using this dataset as the new standard to build new models surpass ours and achieve higher accuracy, we get closer to real-life implementation of these technologies in Industries.

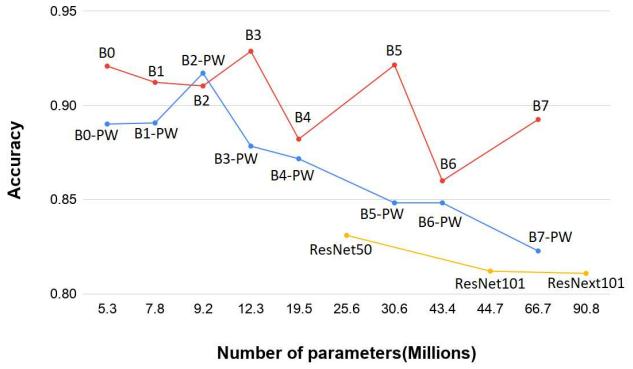
For future work, the aim is to improve the model by further enabling it to classify the objects into sub-categories. This sub-classification would be immensely useful in real-life applications as all objects are not recycled through the same procedure and often require different machinery. This sub-classification would allow recycling plant owners to have the option to accept only those items that the plant can recycle,



**FIGURE 8.** Full process flow of Experimentation and modifications done to test the model on the TrashNet Dataset



**FIGURE 9.** Boxplot of the top performing models to see the minimum, maximum, and deviation across multiple runs



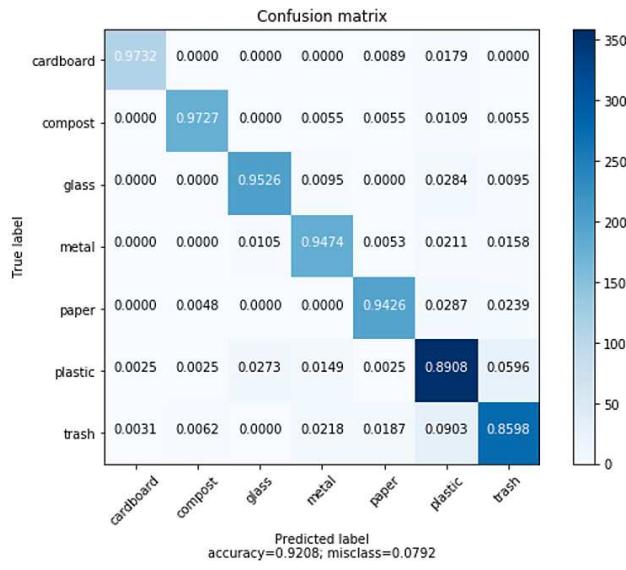
**FIGURE 10.** Comparison of top accuracy's achieved by various models during experimentation with respect to number of parameters

leading to a direct decrease in the amount of non-recyclable trash being collected at these plants. Additionally, we aim to work on improving the size and quality of the data so that human intervention and validation can be eliminated to improve safety measures in the recycling industry. At the

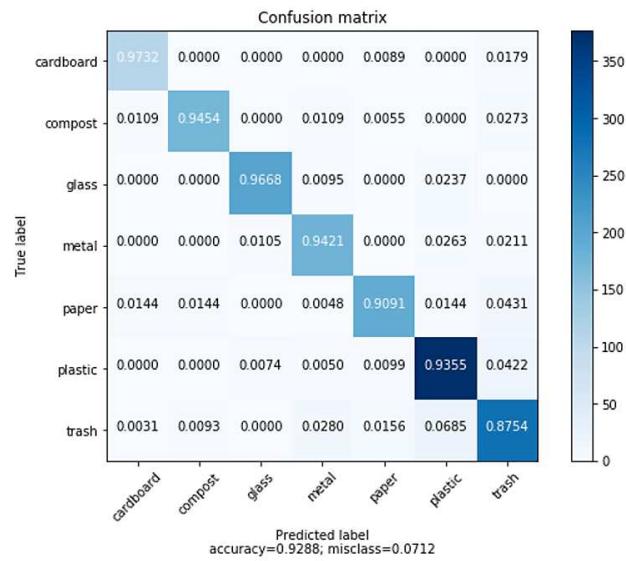
time of writing, we see a lot of untapped potential in this domain and the field is ripe for more research which we hope happens soon.

## VI. CONCLUSION

We set out to create a novel dataset, compare and analyze different Deep learning approaches to classify waste. We successfully constructed a new dataset for trash classification

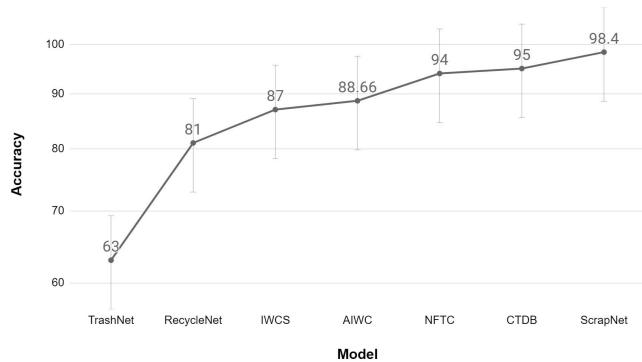


**FIGURE 11.** Confusion Matrix with Actual vs Predicted labels for EfficientNet B0



**FIGURE 12.** Confusion Matrix with Actual vs Predicted label for EfficientNet B3

that consists of over 8100 images of objects spanning seven categories. This dataset is larger and more diverse than any existing dataset. We achieved an mAP score of 82.34 for object detection through EfficientDet B3. After intensive comparison and analysis, we proposed a new deep neural network model for trash classification using the EfficientNet B3, which was modified and fine-tuned to be more efficient to achieve an accuracy of 92.87%. To provide a comparative study, showcase the robustness and effectiveness of our model, we performed several experiments ourselves using different architectures such as ResNet, ResNext, and our model outperformed all of these. In addition to this, we tested our model on TrashNet, compared it with several previous



**FIGURE 13.** Comparison of accuracy of several related works compiled in table 8 and ScrapNet on TrashNet dataset

related works on this dataset as well, and achieved one of the highest accuracies of 98.4% on TrashNet Figure 13. Lastly, we utilized EfficientNet B3 to achieve an accuracy of 89.21% for sub-classification of the plastic category.

## ACKNOWLEDGMENT

We want to express special gratitude to our esteemed university, Manipal University Jaipur, for the invaluable help. Our project could not have been accomplished without their support and helpful insights of all the people along the way.

## REFERENCES

- [1] Olugboja Adedeji and Zenghui Wang. Intelligent waste classification system using deep learning convolutional neural network. *Procedia Manufacturing*, 35:607–612, 2019.
- [2] Lippi Andersen, Kenneth Sandberg, Per Lundqvist, Jens Thulin, Janus Kirkeby, Trine Lund Neidel, and Asbjørn Weidling. Plastic sorting at recycling centres: Guideline. TemaNord. 2015.
- [3] Rahmi Arda Aral, Şeref Recep Keskin, Mahmut Kaya, and Murat Hacıömeroğlu. Classification of trashnet dataset based on deep learning models. In 2018 IEEE International Conference on Big Data (Big Data), pages 2058–2062. IEEE, 2018.
- [4] Cenk Bircanoglu, Meltem Atay, Fuat Beşer, Özgün Genç, and Merve Ayyüce Kızırk. Recyclenet: Intelligent waste sorting using deep neural networks. In 2018 Innovations in Intelligent Systems and Applications (INISTA), pages 1–7. IEEE, 2018.
- [5] Rich Caruana, Steve Lawrence, and C Lee Giles. Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping. In *Advances in neural information processing systems*, pages 402–408, 2001.
- [6] Yinghao Chu, Chen Huang, Xiaodan Xie, Bohai Tan, Shyam Kamal, and Xiaogang Xiong. Multilayer hybrid deep-learning method for waste classification and recycling. *Computational Intelligence and Neuroscience*, 2018, 2018.
- [7] Vadim Evard and Maxim Smirnov. Waste classification data. url=<https://github.com/openrecycle/dataset>, 2019.
- [8] Fiona. What plastic can I recycle? Feb 2018.
- [9] Center for Sustainable Systems. Municipal solid waste factsheet. Pub, (CSS04-15).
- [10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [11] Silpa Kaza, Lisa Yao, Perinaz Bhada-Tata, and Frank Van Woerden. What a waste 2.0: a global snapshot of solid waste management to 2050. The World Bank, 2018.
- [12] Leadpoint. Recycling and waste industry staffing and recruiter: Leadpoint, Apr 2020.
- [13] Lorien Y Pratt. Discriminability-based transfer between neural networks. In *Advances in neural information processing systems*, pages 204–211, 1993.

- [14] Pedro F Proença and Pedro Simões. Taco: Trash annotations in context for litter detection. arXiv preprint arXiv:2003.06975, 2020.
- [15] Victoria Ruiz, Ángel Sánchez, José F Vélez, and Bogdan Raducanu. Automatic image-based waste classification. In International Work-Conference on the Interplay Between Natural and Artificial Computation, pages 422–431. Springer, 2019.
- [16] Sashaank Sekar. Waste classification data. url=https://www.kaggle.com/techsash/waste-classification-data/version/1, 2019.
- [17] Joao Sousa, Ana Rebelo, and Jaime S Cardoso. Automation of waste sorting with deep learning. In 2019 XV Workshop de Visão Computacional (WVC), pages 43–48. IEEE, 2019.
- [18] Mingxing Tan and Quoc V Le. Efficientnet: Rethinking model scaling for convolutional neural networks. arXiv preprint arXiv:1905.11946, 2019.
- [19] Mingxing Tan, Ruoming Pang, and Quoc V. Le. Efficientdet: Scalable and efficient object detection, 2020.
- [20] Anh H Vo, Minh Thanh Vo, Tuong Le, et al. A novel framework for trash classification using deep transfer learning. IEEE Access, 7:178631–178639, 2019.
- [21] Ruosi Wan, Xiangyu Zhang, Jian Sun, and Zhanxing Zhu. Spherical motion dynamics: Learning dynamics of neural network with normalization, weight decay, and sgd. stat, 1050:27, 2020.
- [22] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 10687–10698, 2020.
- [23] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1492–1500, 2017.
- [24] Mindy Yang and Gary Thung. Classification of trash for recyclability status. CS229 Project Report, 2016, 2016.



**MAHESH JANGID** is currently working as an Associate Professor. He has his Ph.D. in Deep Learning. The objective of his research is to use Deep learning approaches in computer vision field and document analysis and recognition. His research interests include machine learning, soft computing, pattern recognition and image processing. He is an IEEE senior member and actively involved in IEEE activities. Apart from that he is also an active member of various globe societies like CSTA-ACM, IEEE-Computer Society, UACEE and IAENG. He has guided Bachelor and Master Degree students in the field of Deep Learning and Digital Image Processing. He has published and presented more 20 research papers in the peer reviewed journals and International Conferences



**RAJESH KUMAR** is with Department of Computer Science, College of Informatics, Bule Hora University, Ethiopia. He is pursuing Ph.D in Computer and Communication Engineering Department from Manipal University, Jaipur. He holds M. Tech Degree in Computer Science and Engineering Department from Guru Jambheshwar University of Science and Technology, Hisar (Haryana). He has more than 8 years of experience in teaching and 2 years of experience in Industry.

His area of Expertise and Interest is in Computer Networking, especially with wireless protocols, and Cyber Security. He has in depth knowledge in Cisco Device Configurations with Routing, Switching and Security. Currently he is working in 5G wireless Technology and has published several research papers in various Peer Review journals and conferences.



**ABHISHEK MASAND** is a final year Bachelors student in Computer science engineering, currently studying at Manipal University Jaipur. His research interests include Imitation Learning, Reinforcement Learning, NLP, and Computer Vision.



**SATYABRATA ROY** is an Assistant Professor (Senior Scale) in the Department of Computer Science and Engineering, School of Computing and Information Technology at Manipal University Jaipur, Rajasthan, India. He received his Ph.D and M.Tech (with honors) degrees in Computer Science and Engineering in 2020 and 2014 respectively; and B. Tech in Computer Science and Engineering in 2009. His research interests include Cryptography, Internet of Things, Cellular Automata, Computer Networks, Computational Intelligence, Machine Learning and Formal Languages. He is an enthusiastic and motivating technocrat with more than 10 years of research and academic experience at different reputed institutes. He has supervised many students for their M.Tech dissertation work and currently supervising Ph.D scholars in the domain of information security. He has published many research articles in top quality International Journals, National/ International conferences of repute. He is also working as reviewer for several reputed International Journals. He has served as member of technical program committee of many international conferences and symposiums. He has organized many international conferences, FDPs and Workshops. He has participated in many Short-Term Courses, Faculty Development Programmes, Workshops and MOOCs offered by prestigious universities of India and abroad. He has served as resource person of many FDPs and seminars. He is senior member of IEEE and professional member of ACM.



**SURYANSH CHAUHAN** is a final year Bachelors student in Computer science engineering, currently studying at Manipal University Jaipur. His research interests include Machine Learning, Data and Business Analytics along with Data exploration.

• • •