Applying Transfer Learning Techniques Using the ResNet Architecture for Garbage Classification

Gurjot Singh Email Id: gurjot08@yorku.ca Student Id: 220190104

Abstract

The aim of this project is to experiment the application of transfer learning techniques using the ResNet (Residual Networks) architecture for garbage classification. For better sustainability and a clean and green environment the garbage classification is of utmost importance. Which further implies a robust classification method or model. Here, Pre-trained models (of ResNet architecture) on large datasets like imagnet are used, further implying the application of transfer learning techniques is done. Thus implying the knowledge gained from pre-existing problems to new unseen problems. Further, the ResNet architecture or deep residual learning which is known for its skip/ shortcut connections provide a good and efficient result and is making the model converge properly. This project aims to build a robust garbage classification model, contributing to efficient waste management.

1. Introduction and Background

The increasing problem of waste management and the need for efficient and accurate garbage segregation, has triggered the field of machine learning as a potential solution.Further, the population is growing day by day, so the garbage is also increasing. Hence, the need of dealing with waste classification is also rising. For this the artificial intelligence techniques have been tried to ease this process.

The garbage classification is a solution to issue of waste management. It is necessary for the recycling process of the waste materials, which further contributes to reducing the environmental effect of the garbage as now the garbage has been handled effectively. Previously this task has been a labor-intensive task which was prone to error as it involved a lot of human input and manual work. But the invention of deep learning has helped to overcome the problem by making the task efficient and easier, thus waiving the human input that was earlier required.

Further, In the tasks dealing with image classification, transfer learning is mostly used in the development of these models. The idea behind the transfer learning is to use a pretrained model which was trained on a large dataset like imagenet and use the knowledge gained from there and apply it to a different issue. Here, in context to garbage classification, it helps the model to converge easily and faster, by using the weights from the training done on the imagenet dataset, further improving the classification performance.

In deep learning networks, the ResNet has performed very well in the image classification tasks. Due to it's skip/ shortcut connection architecture, it has addressed the problem of degardation of training accracy. Which makes it an ideal model for transfer learning, especially on small datasets.

This project aims to observe the performance of transfer learning using ResNet architecture of 50 layers, 101 layers, and 152 layers. This project also aims to observe the results of application of the ResNet models to the task of garbage classification, to see that whether we can achieve a robust model or not. Additionally, in this project a small comparison has been done of these models performance has been done with a proposed method for waste classification using a updated structure of ResNet, that was proposed in *An Improved ResNet-50 for Garbage Image Classification*(Ma et al., 2022).

2. Related Work

Residual Networks present a groundbreaking architecture in the context of deep learning, which is specifically designed to address the challenges associated with training extremely deep neural networks(He et al., 2015). It was developed by Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun at Microsoft Research, ResNet was introduced in their seminal paper *"Deep Residual Learning for Image Recognition"*(He et al., 2015).

First of all, the question arises, Why go Deep? in context to training of neural networks. This is so beacuse going deep in neural networks, specifically using deep models with multiple layers, has proven effective in achieving state-of-the-art accuracy on datasets like ImageNet, like stated in VGG 16/19 architectures in *Very Deep Convolutional Networks For Large-Scale Image Recognition*(Simonyan et al., 2015) and Inception Architecture in *Going deeper with convolutions*(Szegedy et al., 2014). The increased depth allows these models to learn a more complex non linear function. Which leads to improved performance of the model. As a result, deep neural networks play an important role in the field of computer vision.

These Deeper models have some demerits also, as discussed in *"Deep Residual Learning for Image Recognition"*(He et al., 2015). First of all the Vanishing and Exploding Gradient problems, the vanishing gradient problem occurs in deep neural networks where the gradients that are used to update the network become extremely small or "vanish" as they are backpropogated from the output layers to the earlier layers. On the other hand, the exploding gradient problem arises when the gradient continues to get larger which causes a large weight update and results in the Gradient Descent to diverge. Both issues can delay or affect the training of deep networks. These issues were handled by the Normalized Initialization of Weights(LeCun et al.,1998 ; Glorot et al., 2010 ; He et al., 2105 ; Saxe et al., 2013) and the intermediate Normalization layers - Batch Normalization(loffe , 2015).

The next main issue in deep networks is the Degradation of training accuracy that was discussed in (He et al., 2015 ; Srivastava et al., 2015). It indicates that as the network depth increases, training accuracy saturates and then degrades instead of improving. This degradation is not because of overfitting or the addition of more layers also do not lead to a better training performance. The happens because the deeper models with increased complexity find it difficult to converge during the training. Which further suggests that current optimization solvers face difficulties in effectively learning these kind of deep models(He et al., 2015), thus preventing them from performing better than their shallower counterparts, as we may always think that a deep model should always be better than a model with less layers, but it is not true as now the deeper model is giving low accuracy as it is not able to converge and is facing degradation accuracy problem. This issue highlights the need for improved optimization techniques in training deep neural networks.

The above issue was solved by introducing a residual block with a skip connection(He et al., 2015). Here, first let the desired underlying mapping be H(x), where x is the input to the layers. Thus, H(x) is the mapping that the layers need to learn.In a residual connection, we let the layers fit another mapping F(x) such that:-

$$F(x) = H(x) – x$$

. The identity skip connection provides us the original x, so that the output after addition (before ReLU) is:-

$$F(x) + x = H(x)$$

, which is the actual desired mapping. Now, if H(x) were to be an identity mapping, which becomes quite plausible as we increase the depth of the model, then:-

$$H(x) = x$$

.If there is no identity skip connection, the layers would need to learn

$$H(x) = x$$

which would be difficult. If there is an identity skip connection, the layers would need to learn

$$F(x) = 0$$

which is Easy (as we just push the weights of this layer to be zero).

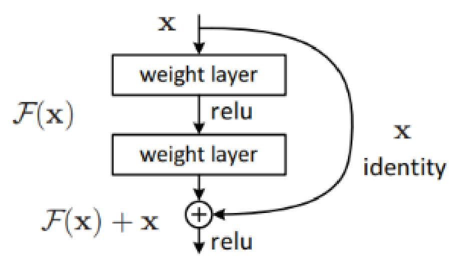

Figure 1. Residual Learning: A Building Block

Further the Identity connection and shortcut connections were introduced. In a neural network, when the input x and the output y of a layer have the same shape, an identity shortcut is used, which allows the direct addition of the input to the learned residual mapping $F(x)$. This is represented as

$$y = F(x, W_i) + x$$

, which enables the network to learn residual information directly without additional transformations. And in cases where $F(x)$ and x have different shapes, a projection shortcut is used. Here,

$$y = F(x, W_i) + W_s x$$

, involving a linear projection W_s, making the match of dimensions between the input and the output for effective residual learning.

In the Network Architecture of ResNet in the image, the solid line represents the identity shortcuts, that are used when the dimensions are same , and the dotted line represents the projection shortcuts, that are used when the dimensions are not same.

Here, the Image is of ResNet 50(an 50 Layer Residual Network).

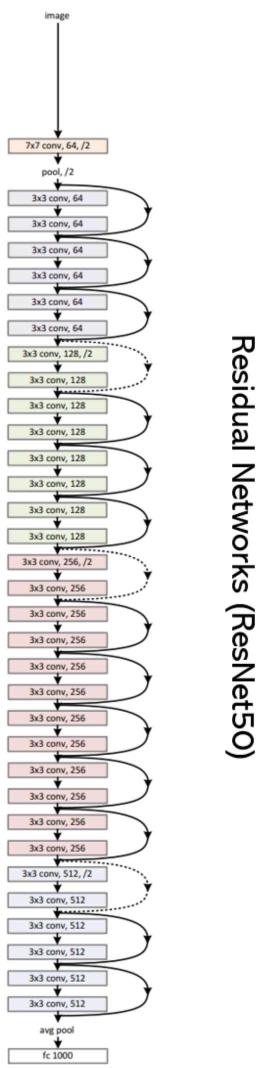

Figure 2. ResNet 50

In this study, we have taken the ResNet50, ResNet 101, ResNet152. There is one structural differnece in all the three models, that is the number of layers and the number of FLOPs.Typically, ResNet101 is a ResNet50 with added layers, same for ResNet152, as shown in the below image. FLOP stands for Floating Point Operations, and it represent the number of floating-point operations needed to process a single input data point through the network.

ResNet50	3.8×10^9
ResNet101	7.6×10^9
ResNet152	11.3×10^9
VGG19	19.6×10^9

Table 1. FLOP Comparison

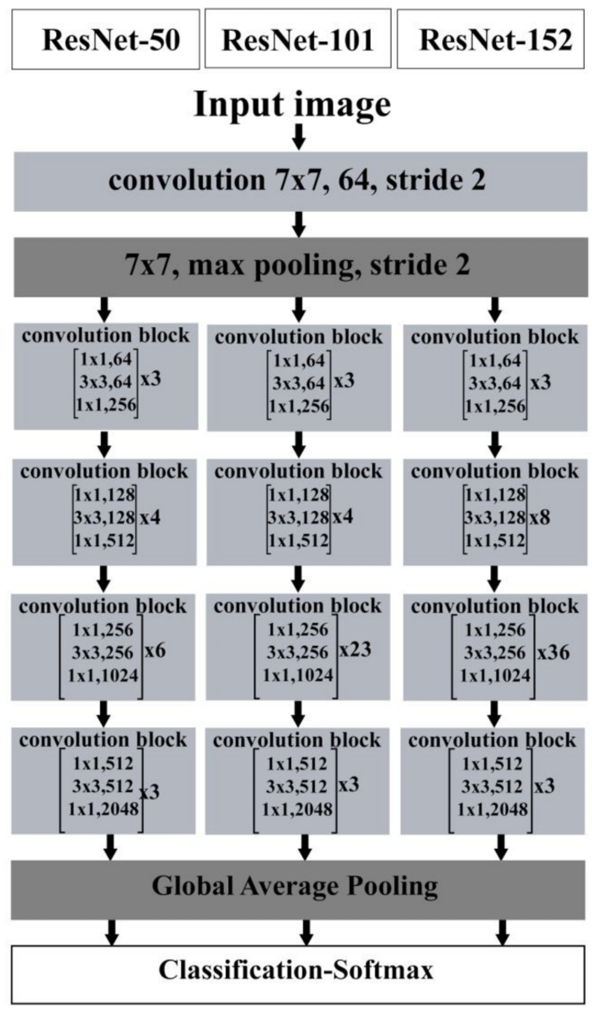

Figure 3. ResNet50-101-152 Architecture

From the above it is evident that ResNets are better and are preferred to be used in the deep networks, as they have less number of FLOPs as in a 152 layer network the FLOPs are less than 19 layer deep VGG Network.

Now as a result, extremely deep residual nets are easy to optimize, but the counterpart "plain" nets (that simply stack layers) exhibit higher training error when the depth increases, and the deep residual nets can easily enjoy accuracy gains from greatly increased depth, producing results substantially better than previous networks(He et al., 2015). We can say that in shallower models the performance of both the residual and simple networks would be comparable, but the ResNets in that case allows faster convergence than shallower models

Talking about transfer learning. Transfer Learning usually takes place in two steps.
Using a pre-trained model for feature extraction: When working with a small dataset, it is a common practice to take advantage of features learned by a model trained on a larger dataset in the same domain. This is done by instantiating the pre-trained model and adding a fully-connected classifier on top. The pre-trained model is "frozen" and only the weights of the classifier get updated during training. In this case, the convolutional base extracted all the features associated with each image and you just trained a classifier that determines the image class given that set of extracted features.(Transfer Learning and Fine-tuning, n.d.)

Fine-tuning a pre-trained model: To further improve performance, one might want to repurpose the top-level layers of the pre-trained models to the new dataset via fine-tuning. In this case, you tuned your weights such that your model learned high-level features specific to the dataset. This technique is usually recommended when the training dataset is large and very similar to the original dataset that the pre-trained model was trained on.(Transfer Learning and Fine-tuning, n.d.)

3. Methodology

The methodology of this project is the application of transfer learning(as discussed above) using three differnet ResNet Architectures ResNet-50, ResNet-101, and ResNet-152. Further checking whether the notion about the ResNets hold true or not, that is the notion that increasing the layers will lead to increased accuracy, which was originally not possible in deep models due to the problem of degradation accuracy, which was solved in Residual Networks. Further, we also observe the benefit of using transfer learning, by comparing the results between using transfer learning (using imagenet weights) and by not using the transfer learning. Further drawing a comparison between my approach and a proposed approach for using an updated ResNets for Garbage Classification in *"An Improved ResNet-50 for Garbage Image Classification* (Ma et al., 2022)".

3.1. Data Selection and Preprocessing

For the purpose of this project the data of TrashNet available on kaggle as " Garbage Classification" was used. This data contains a total of 2527 images belonging to 6 classes plastic, glass, metal, paper, cardboard, and trash. :-

cardboard	glass	metal	trash	paper	plastic
403	501	410	137	594	482

Table 2. Images per class

To ensure uniformity among the images all images undergo a preprocessing step in which all the images are resized from 512 X 384 into 180 X 180 pixels. Further the Data Augmentation, flipping, rotating, changing the brightness and contrast, is done to enhance the model's ability to generalise across diverse conditions.

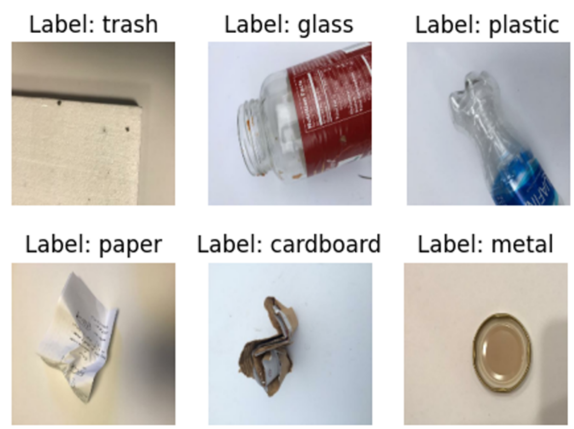

Figure 4. Sample Images from each class

3.2. Model Selection and Transfer Learning

For this task 3 variants of ResNets were selected namely ResNet-50, ResNet-101, and ResNet-152, based on their performance on larger datasets like CiFar10 dataset where these 3 models performed the best. Further to implement transfer learning, these three selected models are loaded with pre-trained weights from imagenet, and the last classification layer is replaces by three added layers and one output layer which tailored to the specific Garbage Classification dataset. Moreover, all three models are trained without transfer learning that is the model learn it's own weights.

3.3. Trainig and Validation

For this dataset, the code has been compiled using the T4 GPU of Google Colab, and the data was split into 80:20 train: validation split. 100 epochs were run for each ResNet variant and results were monitored. The model's performance were measured by the validation accuracy.

4. Experiment and Results

In this task, first of all the data was split into 80:20 train and validation datasets and the data was read from the drive using the flowfromdirectory function of Keras. After that, to ensure the ability of model to perform well, data augmetation was performed using the below parameters.

Function	**Parameter**
RandomFlip	horizontal
RandomRotation	0.45
RandomContrast	0.2
RandomBrightness	0.2

Figure 5. Data Augmentation Parameters

The Augmentation was applied on the training dataset. After that the ResNet50 Model was intialized using the imagenet weights. For the purpose of transfer learning the intial model layers were freezed. And 3 layers and one output layer was added to the model. The added layers were

Layers added
Flatten()
Dense(units=1024, activation = 'relu')
Dense(units=128, activation = 'relu')
Dense(6, activation = 'softmax')

Figure 6. Added Layers

The parameters of these layers were decided after experimenting different set of parameters. The parameters and the number of layers that gave maximum accuracy on both training and the validation set were chosen. The last layer or the output layer was chosen to be softmax as it outputs the probability of an image belonging to a certain class, from which we assign the label of the image to be the class with highest predicted probability.

Next, these added layers were trained on 10 epochs after freezing the top layers. Here the learning rate was 0.001. After that fine tuning was done, implying the model's freezed layers were unfrozen and the full model was trained. The training was done on complete model in 100 epochs, and the model with best val-accuracy was saved, for further predicting the external images. Here the learning rate used was 0.0000055. The learning rate value was decided on the basis of experiments as various learning rates were tried and tested and the one with best accuracy was finally used.

This whole same process was done 3 times one each for the ResNet50, ResNet101, and ResNet152. And the results were noted and graphs were plotted, between all these 3 models.

Models	val_accuracy	val_loss
ResNet50	92.475	0.4407
ResNet101	93.861	0.3342
ResNet152	94.059	0.3067

Figure 7. Model Performance

The evaluation of the ResNet models in our project shows a notable improvement in validation accuracy and reduction in validation loss with increasing model depth - making the basic notion about the deep models of achieving same or better accuracy with increaing model depth, instead of facing the degradation problem. Where the accuracy increased from 92.4 to 93.8 in resNet50 to ResNet101, further to 94.05 in ResNet152. Now, ResNet152 which is the deepest among the tmodels used, achieved the highest validation accuracy of 94.059 further showing its superior ability to capture and learn more complex features of the data. The strong correlation of increasing depth and accuracy is visible in the graph.

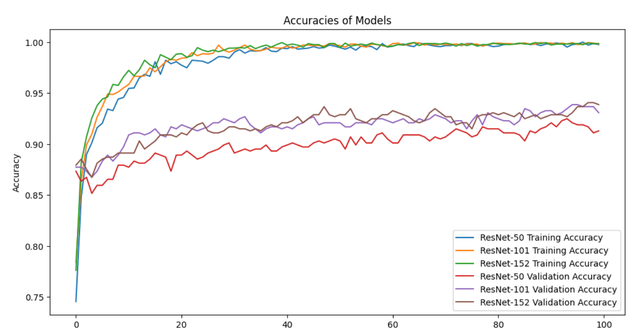

Figure 8. Training Accuracies vs Test Accuracies for 3 Models

Furhter a noticable trend of increasing depth and decreasing loss can be seen table as well as from the graphs. The loss decreased from 0.44 to 0.33 in ResNet50 to ResNet101 and further to 0.3 in ResNet152. This result aligns with the inherent design of residual networks, allowing for the effective training of deep networks by mitigating the degradation accuracy problem.

Moreover, the loss was minimum for the ResNet152, thus proving its robustness in this garbage classification task. A better model convergence can also be infered from this, in the case of ResNet152.

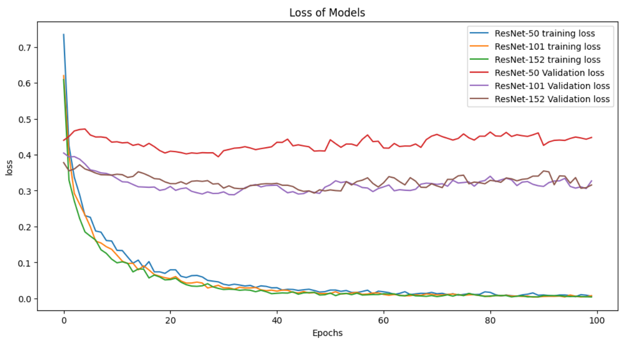

Figure 9. Training vs Validation Loss of 3 Models

Below is the prediction of the three models when the models were given the images of glass and plastic.

```
1/1 [==============================] - 2s 2s/step
1/1 [==============================] - 2s 2s/step
1/1 [==============================] - 3s 3s/step
```

```
Image: /content/drive/MyDrive/glass477.jpg
Resnet50 Prediction: glass
Resnet101 Prediction: glass
Resnet152 3 Prediction: glass
------------------------------
1/1 [==============================] - 0s 31ms/step
1/1 [==============================] - 0s 34ms/step
1/1 [==============================] - 0s 53ms/step
```

```
Image: /content/drive/MyDrive/plastic472.jpg
Resnet50 Prediction: plastic
Resnet101 Prediction: plastic
Resnet152 3 Prediction: plastic
------------------------------
```

Figure 10. Model Predictions - All Models are predicting correctly

4.1. Benefit of Using Transfer Learning

In the previous section we saw that all 3 models performed pretty well, which is evident from their achieved accuracies for the task. Now, further the experiments were also conducted to check the model learning wothout using transfer learning. The experiments were conducted exactly the same as with transfer learning, the only difference was that in this case the weights were not intialised as the imagenet weights. The model learnt it's own weights. On conducting the experiment on ResNet50, I saw that the achieved accuracies were nearly equal as compared to the model with tranfer learning. But the loss seemed to be diverging.(Like in the image below). From here, I observed that the loss seemed

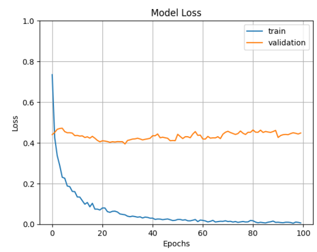

Figure 11. Resnet 50 Loss

to be diverging. So I run 130 epochs on all models to test whether it is really diverging or not. After running all 130 epochs, I came to a conclusion that the model without transfer learning (without the image net weights) was overfitting and it was evident from the val loss generated graphs.

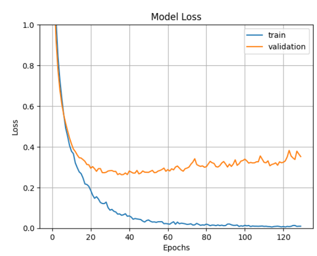

Figure 12. Resnet 50 loss - 130 epochs

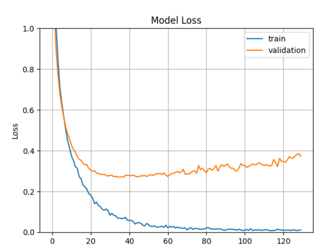

Figure 13. Resnet 101 loss - 130 epochs

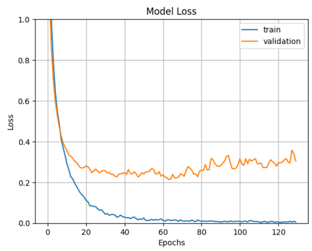

Figure 14. Resnet 152 loss - 130 epochs

From here we can conclude that Using transfer learning gave us better results, however the accuracy is nearly same as in this we got ResNet50 - 92.04, and was the same in ResNet101 and ResNet152 that was 93.693. Which is nearly equal to the model with tranfer learning and only a minute differnece of 0.2-0.4 percent can be observed.

5. Discussion

In this section we will have a discussion over the comparison between the proposed appoach in *"An Improved ResNet-50 for Garbage Image Classification* (Ma et al., 2022)" and my approach of transfer learning using ResNet. But first we will discuss about the reason of difference in the models with and without learning. Now, as in this project SoftMax is used int he last layer, which outputs the probabilities of each class and the class with maximum probalitiy is selected. Let's take an example, if the softmax outputs [0.32,0.2,0.1,0.33,0.075,0.075] , from here class 4 with probability of 0.33 would be selected. In an another case the softmax outputs [0.1,0.1,0.1,0.65,0.0250,0.0250] from here also class 4 would be selected. But we can clearly see that the loss in example 2 would be less than that of example 1. From here we find out the reason why the models with and without transfer learing are having similar accuracies as the class with the highest probability is predicted even if there is very minute differnce between the 6 probabilties like in example 1, which implies that the loss would be more. As a result of which we can see that the models without transfer learning are overfitting the dataset. And teh model with Transfer Learning is performing better. Now, we can draw a samll comparison between the proposed appoach in *"An Improved ResNet-50 for Garbage Image Classification* (Ma et al., 2022)" and my approach of transfer learning using ResNet. The paper proposes an improved ResNet-50 model for garbage image classification to address issues with existing models like lack of data, few samples, and inter-class similarity. Modifications are made in two areas:

(a) Residual block is altered by adding an attention module (CBAM)(Woo et al., 2018) to filter features and changing the downsampling method to reduce information loss.

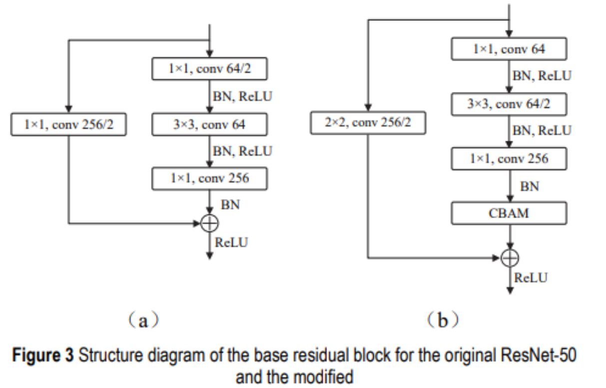

Figure 15. Structure diagram of the base residual block for the original ResNet-50 and the modifies approach as proposed in (Ma et al., 2022)

(b) Multi-scale feature fusion is added through horizontal and vertical fusion to optimize feature usage.

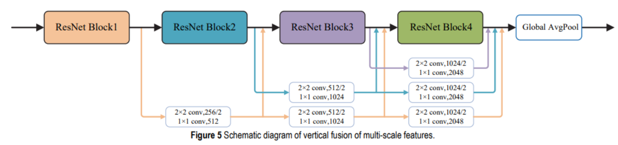

Figure 16. Proposed vertical fusion

In this paper the author used the same TrashNet dataset, that was used in the current study. The author reports the accuracy from the proposed approach Experiments on the

Model	Valloss	Valaccuracy
ResNet-50	0.8039	84.46%
ResNet-50-A	0.4472	88.40%
ResNet-50-B	0.3371	92.08%

Figure 17. Performance achieved(Ma et al., 2022)

TrashNet dataset show the improved model (ResNet-50-B) achieves 92.08 accuracy, outperforming the original ResNet-50 (84.46 accuracy) by 7.62. In summary, the paper presents an enhanced ResNet-50 model for garbage classification that achieves higher accuracy and robustness compared to prior art. Key innovations include adding attention modules and multi-scale feature fusion. Now Further drawing the comparison:- Here all the training has been done on 250 epochs, and the below parameters,
Further, it is evident that with less and a different data augmentation and different parameters my proposed model is performing better in terms of both validation accuracy and validation loss. As I achieved 92.475 validation accuracy in

Project	Settings
shear_range	0.1
zoom_range	0.1
width_shift_range	0.1
height_shift_range	0.1
horizontal_flip	True
vertical_flip	True
rotation_range	20
fill_mode	nearest

Figure 18. Data Augmentation(Ma et al., 2022)

a simple ResNet50 model which is still larger than the proposed ResNet50-B(Ma et al., 2022), where it is achieving 92.08 percent validation accuracy. On increasing the layers the validation accuracy in my proposed model went upto 94 percent and the loss went down to 0.30 in ResNet152. Importantly, these results are achieved in a shorter training duration of 100 epochs, demonstrating the efficiency and efficacy of the proposed model in achieving robust performance with optimized parameters and reduced training time.

6. Conclusion

In context to Garbage Classsification through transfer Learining using the ResNet architecture, this project has yielded successful and promising results as evident from the achieved accuracies. Further, the evaluation of ResNet50, ResNet101, and ResNet152 on the kaggle dataset, provided valuable details on the models' effectiveness and adaptability in the context of waste management.

The accuracy achieved for ResNet-50 (92.475) , ResNet-101(93.861), and ResNet-152(94.059) imply that the transfer learning has been successfully integrated with the ResNet architecture. Moreover the performance is highlighting the strength of ResNet architecture, in capturing important features when pre-trained models are used.

The observed trend of the increasing model performance when adding more layers is observed as seen in the increased accuracy achieved in ResNet152, Furhter, the accuracy of the ResNet-152, suggests that the model has effectively learned to classify the various waste materials, thus contributing to the optimization of waste management processes.

Moreover we can also say that the observed differences in model performance with and without learning is due the use of argmax. As a result, it doesn't care much how high the output probability, rather it focuses on which node has the highest probability, from which we further select the class with highest probability, which leads similar accuracies for bith models. However, a difference between the loss was also seen, implying the overfitting of models without transfer learning. Thus, the use of transfer learning techniques with ResNet architecture in this project, proves benficial in mitigating overfitting and enhancing the overall performance of the models.

Further, we may also say that, in future the proposed methods in *"An Improved ResNet-50 for Garbage Image Classification* (Ma et al., 2022)" can also be applied onto this project, and test the performance of the model, as there may be chances to achieve a better model than this.

Concludingly, we can say that, the results of this project not only proves the efficacy of the ResNets with transfer Learning, but also open gates for future work. Future work may delve into the optimization of hyperparameters used, applying diffrenent architecture like we can test the approach in *"An Improved ResNet-50 for Garbage Image Classification* (Ma et al., 2022)" , and further hope to see an improved performance of this model.

References

Ma, X., Li, Z., Zhang, L. (2022). An improved ReSNet-50 for garbage image classification. Tehnicki Vjesnik-technical Gazette, 29(5). https://doi.org/10.17559/tv-20220420124810

He, K., Zhang, X., Ren, S., Sun, J. (2015). Deep Residual Learning for Image Recognition. ArXiv (Cornell University). https://doi.org/10.48550/arxiv.1512.03385

Simonyan, K., Zisserman, A. (2015, April 10). Very Deep Convolutional Networks for Large-Scale Image Recognition. ArXiv.org. https://arxiv.org/abs/1409.1556

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A. (2014). Going Deeper with Convolutions. ArXiv.org. https://arxiv.org/abs/1409.4842

Ioffe, S., Szegedy, C. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. ArXiv.org. https://arxiv.org/abs/1502.03167

Glorot, X. and Bengio, Y. (2010) Understanding the

Difficulty of Training Deep Feedforward Neural Networks. Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics. Proceedings of Machine Learning Research, 9, 249-256. http://proceedings.mlr.press/v9/glorot10a.html

He, K., Zhang, X., Ren, S., Sun, J. (2015). Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. IEEE International Conference on Computer Vision (ICCV 2015), 1502.

LeCun, Y., Bottou, L., Orr, G. B., Müller, K.-R. (1998). Efficient BackProp. Lecture Notes in Computer Science, 9–50. https://doi.org/10.1007/3-540-49430-82

Saxe, A. M., McClelland, J. L., Ganguli, S. (2014, February 19). Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. ArXiv.org. https://doi.org/10.48550/arXiv.1312.6120

He, K., Sun, J. (2014, December 4). Convolutional Neural Networks at Constrained Time Cost. ArXiv.org. https://doi.org/10.48550/arXiv.1412.1710

Srivastava, Rupesh Kumar, Greff, K., Schmidhuber, J. (2015). Highway Networks. ArXiv.org. https://arxiv.org/abs/1505.00387

Transfer learning and fine-tuning. (n.d.). TensorFlow. https://www.tensorflow.org/tutorials/images/transferlearning

Woo, S., Park, J., Lee, J. Y., Kweon, I. S. (2018). Cbam: Convolutional block attention module. Proceedings of the European conference on computer vision (ECCV), 3-19. https://doi.org/10.1007/978-3-030-01234-21