Garbage classification: a comparitive study for various deep learning cnn models

Mizani F., Modaberi S., Abbaspour A., Sharifisadr N., Zirahi A.

University of Calgary

# Abstract

Canada generates approximately 31 million tons of garbage a year, of which only 30% gets recycled. The development of a reliable model to identify garbage types can significantly help increase recycling. This report focuses on developing garbage classification models that aim to distinguish the four major types of waste: Black (trash), Blue (recycle), Green (Compostable), and Take-to-recycle (recycling drop-off). The chosen classification models were VGG19, DenseNet121, MobileNet, InceptionV3, InceptionResNetV2, and ResNet50 Convolutional Neural Networks. The models were trained on an evenly distributed training dataset that contained over 5000 images of the four classes of waste. The validation and test sets contained more than 1000 images evenly distributed. The models were pretrained on the ImageNet dataset and were run under two conditions: 1) freezing the weights at initial training and then fine-tuning for five epochs, and 2) letting the weights change on training. The obtained result indicated that the highest scores are 82% for the InceptionV3 model and 84% for the DenseNet121.

**Index Terms—** Garbage classification, classifications, CNN, Deep Learning, Machine Learning

# 1. Introduction

Garbage recycling and management plays an important role in improving the quality of living environments and resource conservation. Nowadays, the amount of daily waste that is being produced is increasing. It is predicted that waste production will increase by 70 percent in the year 2025 (Bhada-Tata, 2012). Increasing waste discharge results in environmental pollution and it is hazardous to human health. Therefore, waste management is becoming a great concern for different countries and governments across the world. One of the most important steps of waste management is to separate waste accurately at the source to different components and this process was previously done manually by human. Today, with the development of artificial intelligence and machine learning methods, the classification and separation of waste can be done simply without human intervention. In previous studies, many different algorithms that have been proposed for image classification, were used for garbage classification purposes. Most of them were based on convolutional neural network (CNN) architecture. This research is a comparative study that aims at classifying our garbage dataset to four different groups. The data set that is being used for this purpose is relatively small, therefore we relied on transfer learning and trained several pretrained models on our data to classify garbage.

The first model we used is VGG-19 (Simonyan, 2015). This model is a 19-layer (16 conv., 3 fully connected) architecture that strictly uses 3×3 filters with stride and pad of 1, along with 2×2 max-pooling layers with stride 2. VGG-19 architecture, due to their depth are slow to train and produce models of very large size. Rectified linear unit is used to make the model classify better. It is trained on more than a million images and can classify images into 1000 object categories. As a result, the model has learned rich feature representations for a wide range of images.

The next model we trained on data on, is Inception model. In this model,inception modules were invented for more efficient computation and deeper networks to solve the problem of computational expense, as well as overfitting, among other issues. They allow a dimensionality reduction with stacked 1×1 convolutions. In addition, they take multiple kernel filter sizes within the CNN, instead of stacking them sequentially, and order them to operate on the same level. The most simplified version of an inception module works by performing a convolution on an input with three different sizes of filters (1x1, 3x3, 5x5) in addition to a max pooling. Then, the outputs are concatenated and sent to the next layer by structuring the CNN to perform its convolutions on the same level, which makes the inception network gets progressively wider, not deeper (Szegedy, 2015).

The third model that is widely used in image classification problems is ResNet model (He, 2016). ResNet is a CNN architecture that consists of a series of residual blocks (ResBlocks) with skip connections. ResNet architecture was proposed aiming to help with training extremely deep neural networks, as it was difficult to train deep networks due to the problem of vanishing gradients. In contrast to the general thought that increasing the number of layers would increase the performance of neural network, studies show that increasing the number of layers at some point can even worsen the performance due to vanishing gradients. ResNet employed skip connections to address the mentioned problem of deep networks (Alzubaidi, 2021). Upon invention of ResNet, the architecture won ImageNet competition in 2015 by a significant margin due to the skip connections in its networks (Han, 2018). ResNet has been widely used for different proposes such as image classification and segmentation. utilized ResNet-152 for twelve skin lesion classification (Zhang, 2019). modified the base ResNet-50 by enhancing residual learning with attention modeling to be used for melanoma recognition. They reported that the modified model outperformed the base ResNet-50 (Li, 2019). applied attention-based ResNet to U-Net segmentation for chest X-ray segmentation.

Considering that mobile applications can be useful in the process of waste classification, we used MobileNet as the fourth model we trained on data with. Although several well-known models such as VGG19, ResNet and Incpection show promising accuracy on ImageNet, they are large, complex and need plenty of calculations. Thus, these models are not suitable for mobile and embedded vision applications. MobileNet is a lightweight efficient model with high accuracy which has been applied for image classification and specially for garbage classification. (Aral, 2018; Rabano, 2018; Feng, 2021; Liu, 2018)This model is recommended by the Google research team. MobileNet is a CNN containing 28 layers consists of depth-wise separable convolutions and it has a streamlined architecture (Howard., 2017).

Finally, we tested Densely Connected Convolutional Networks (DenseNet) (Gao Huang, 2016) which was specially developed to improve accuracy caused by the vanishing gradient in high-level neural networks when they go deeper. This is because the path for information from the input layer until the output layer (and for the gradient in the opposite direction*)*becomes so big, that they can get vanished before reaching its destination. The authors solve the problem ensuring **maximum information (and gradient) flow**. To do it, they simply connect every layer directly with each other. By connecting this way DenseNets require fewer parameters than an equivalent traditional CNN, as there is no need to learn redundant feature maps*.* Furthermore, some variations of ResNets have proven that many layers are barely contributing and can be dropped. In fact, the number of parameters of ResNets are big because every layer has its weights to learn. Instead*,*DenseNets layers are very narrow (e.g. 12 filters), and they just add a small set of new feature-maps.

# 2. Dataset

We studied a dataset that was collected and labeled by students at the University of Calgary. The dataset consists of 7069 garbage images of 256x256 pixels that have been distributed in 4 bins. The dataset has been divided into Train, Validation, and Test folders. The objects in the images are expected to be centralized in the photo and have homogeneous background. Every photo should contain only one object. However, all the collected photos in the dataset are not correctly following these conditions.

Black (non-recyclable and non-compostable), Blue (recyclable), Green (compostable), and Take-to-recycle (landfill). The following histogram and table show that the number of images is balanced in each bin.

Graphical user interface, application, Excel

Description automatically generated

**Fig. 1.** Histogram of garbage distribution

**Table 1.** Overview of garbage dataset

|  |  |
| --- | --- |
| Garbage | Num. of images |
| Black | 1765 |
| Blue | 1812 |
| Green | 1814 |
| Take-to-recycle | 1678 |

There is a considerable number of samples in the dataset that may have been incorrectly labelled. Our team kept the labels as the original dataset in all the experiments and did not add any images to the dataset either.

## Train, Test and Validation split

There are 5139 images in the train set, 1123 images in the validation set, and 807 images in the test set. Each dataset has 4 classes that represent 4 garbage bins. The following figure shows some samples from the dataset.

Graphical user interface, application

Description automatically generated

**Fig. 2**. Train garbage dataset samples

# 3. Experimental Setup

In this section we explain our experimental setup in four subsections including data augmentation, models, training process and evaluation metrics.

## 3.1. Data Augmentation

As we have a small data set, we apply different data augmentation techniques such as rotation (20°), flipping horizontally and vertically, zooming (range 0.1), shearing (range 0.2) and shifting width and height (range 0.1). For all models batch size 64 was used for data augmentation.

## 3.2. Models

Since the applied garbage classification in this study is small, we used transfer learning to leverage features learned on a large dataset to classify a smaller dataset. For this aim we used pre-trained models on ImageNet including VGG19, Inception V3, InceptionResnet V2, DenseNet12, ResNet and MobileNet as base models. We discarded the top layer and added a dense and a softmax layer to each model. We trained these models with our data set through two scenarios: In the first one, we froze a model and then fine-tuned it. In the second one, we made the base model trainable and trained the whole model. Also, in order to investigate to what extent applying dropout could be effective in overcoming the overfitting problem in the VGG19 model we used the third scenario. In the third scenario we froze the model, created a new top of the model, used dropout and fine-tuned respectively. The input size for all models was (256,256).

## 3.3. Training Process

For all models, we initialized each model with Adam Optimizer with 50 epochs for MobileNet, ResNet 50 and DenseNet121, also 20 epochs for Inception V3 and InceptionResnet V2. Additionally, VGG19 trained with 10 and 50 epochs, respectively. In the first scenario, we fine-tuned all models through 5 epochs. We applied an early stopping callback setting with a patient of 10. The initial learning rate was 1e-5 which halves every 10 epochs. As this problem is a multiple classification, we used categorical cross entropy as loss function and accuracy as a metric to be observed. All models are trained on Google Colab.

## 3.4. Evaluation Metrics

As our problem is classification, we reported the performance of our models applying standard evaluation metrics: Accuracy and Confusion Matrix.

# 4. Results and discussion

Different models were trained to find the best performing model in order to best classify garbage into appropriate categories. A summary of models’ performance and metrics can be in Table 1.

**Table 2.** Performance of models

|  |  |  |  |
| --- | --- | --- | --- |
| Model | Train Accuracy | Validation  Accuracy | Test  Accuracy |
| VGG19-  freeze | 92 | 79 | 76 |
| VGG19-  trainable | 95 | 81 | 76 |
| VGG19-  freeze-Dropout | 95 | 83 | 81 |
| MobileNet-  freeze | 92 | 81 | 77 |
| MobileNet-trainable | 98 | 84 | 79 |
| ResNet50-  freeze | 93 | 83 | 77 |
| ResNet50-  trainable | 98 | 86 | 76 |
| DenseNet121-  freeze | 93 | 82 | 79 |
| DenseNet121-  trainable | 99 | 86 | 84 |
| InceptionV3-  freeze | 90 | 80 | 75 |
| InceptionV3-  trainable | 99 | 84 | 82 |
| InceptionResNetV2-  freeze | 93 | 83 | 80 |
| InceptionResnetV2-  trainable | 98 | 87 | 79 |

Analyzing the accuracy of models for different scenarios shows that all models have close performance to top-1 accuracy reported for ImageNet dataset in Keras official webpage. Results indicates that for some models such as VGG19, ResNet50, and InceptionResNetV2, there is not much difference between when we fine tune the whole model and the case that we freeze the functional model and just train the last activation layer. However, for MobileNet and InceptionV3 the accuracy increases up to 7 percent when we train the whole model with our dataset.

Classification report and confusion matrix show that for all models when we just trained the last layer, the main misclassification happens for Black and Blue categories where the model cannot sufficiently differentiate them and label them wrongly. However, when we trained the whole models, they could better classify Black and Blue categories and the accuracy increased. Table 2 and 3 shows the classification report and figure 1 and 2 shows confusion matrix for Inception V3 for two mentioned scenarios.

**Table 3.** InceptionV3-freeze classification report



Chart, treemap chart

Description automatically generated

**Fig. 3.** InceptionV3- freeze confusion matrix

**Table 4**. InceptionV3-trainable classification report



Chart

Description automatically generated

**Fig. 4.** InceptionV3- trainable confusion matrix

Confusion matrices and classification report show that when we tune the pre-trained weights with our dataset, it increases the model ability to distinguish better Black and Blue classes.

In order to perform error analysis and further explore the results, we investigated the images that were misclassified to find out the reason behind the misclassification. We realized that there are some wrongly labeled images in the dataset and sometime images are very distorted due to augmentation that makes it difficult for the model to classify. Also, we observed in some cases that the model predicted correctly, however, it was considered as misclassification due to initial wrong labeling. Figure 3 shows some misclassified images predicted by ResNet50 and MobileNet.

A picture containing calendar

Description automatically generated

**Fig. 5.** Example of misclassified items from MobileNet and ResNet50 models.

Our results showed that Inception V3 and DenseNet121 outperformed other models with 82 and 84 percent accuracy for test dataset. In order further improve the models, it is recommended that image labels get revised before training and validating the model. Also, it is worth mentioning that increasing the size of dataset with high quality and correctly labeled data can help models to train better and have more capability to classify garbage.

# 5. CONCLUSION and recommendation

In this work six CNN models were run on the garbage dataset under two conditions. The result indicates that not necessarily freezing the weights of a pretrained model gives higher scores for all models. In fact, the two highest scores achieved are from InceptionV3 and DenseNet121 models, trained with trainable weights.

# 6. acknowledgement

The authors of this report are grateful for the help and support from Dr. Roberto Souza. In addition, we acknowledge the help from students at the University of Calgary who contributed to the garbage classification dataset. Finally, we acknowledge help from the Schulich School of Engineering and the University of Calgary.

# 7. References

Alzubaidi, L. Z.‑A. (2021). Review of deep learning: concepts, CNN architectures, challenges, applications, future directions.

Aral, R. A.-2. (2018). Classification of trashnet dataset based on deep learning models. *IEEE International Conference on Big Data (Big Data)* (pp. 2058-2062). IEEE.

Bhada-Tata, D. H. (2012). A Global Review of Solid Waste Management. 1-116.

Feng, J. T. (2021). Garbage disposal of complex background based on deep learning with limited hardware resources. *IEEE Sensors Journal*, 21050-21058.

Gao Huang, Z. L. (2016). Densely Connected Convolutional Networks}. *CoRR*.

Han, S. S. (2018). Classification of the clinical images for benign and malignant cutaneous tumors using a deep learning algorithm. *Journal of Investigative Dermatology*, 1529-1538.

He, K. Z. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770-778.

Howard., A. G. (2017). MobileNets: Efficient Convolutional Neural Networks for Mobile Vision. *CoRR*.

Li, B. K. (2019). Attention-guided convolutional neural network for detecting pneumonia on chest x-rays. *41st annual international conference of the IEEE engineering in medicine and biology society*, (pp. 4851-4854).

Liu, Y. G. (2018). Research on automatic garbage detection system based on deep learning and narrowband internet of things. *Journal of Physics: Conference Series* (p. 012032). IOP Publishing.

Rabano, S. L. (2018). Common garbage classification using mobilenet. *IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)* (pp. 1-4). IEEE.

Simonyan, A. Z. (2015). Very deep convolutional networks for image recognition. *ICLR*.

Szegedy, C. (2015). Going deeper with convolutions. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*.

Zhang, J. X. (2019). Attention residual learning for skin lesion classification. *IEEE transactions on medical imaging*, 2092-2103.