CNN Thunderdome Showdown: Benchmarking YOLOv7, VGG16 and GoogLeNet for Recyclables Image Classification Accuracy Final Project Report

EECS 4404: Introduction to Machine Learning and Pattern Recognition

York University

Tom Sun, Dexuan Ren, Deepta Adhikary (Group A)

April 8, 2023

1 Abstract

Image classification is an important task in computer vision that has many practical uses. However, it is challenging due to the wide range of visual content it has to deal with and the variations that can exist within a single category. Additionally, building models from scratch and computing power requirements make image classification inaccessible for general problem-solving purposes. Transfer learning can address this issue by using pre-trained models such as Convolutional Neural Network (CNN). For this study we have selected the domain of recyclable sorting and classification. In our study, we tested three models - YOLOv7, VGG16, and GoogLeNet - to determine the best overall performance. We found that GoogLeNet was the most effective for our dataset, with an accuracy rate of 94% and faster training speed. This result is comparable to legacy techniques used in recyclable sorting used today, which includes manual labour or expensive sensors. Recycling plants can use our findings and apply them to reduce costs on machinery to maintain their processes. This idea can also be used to create garbage collecting mini-robots that can skim through the streets to collect waste and recycle them in the appropriate bins.

2 Introduction

2.1 Topic

Image classification is a challenging task in computer vision due to the diversity of visual content and variations within a single category. The complexity of building models from scratch and the high computational requirements also make it difficult for general problem-solving purposes. However, transfer learning using pre-trained models such as Convolutional Neural Networks (CNN) can overcome these challenges.

2.2 Scope and Assumptions of the Project

In our study, we focused on image classification of recyclables and used a dataset from Kaggle. We assume that recycling plants have techniques to isolate each piece of recyclable and have the means of capturing images. Since we are working on multi-class classification, having more than one recyclable in one image would make it multi-label classification.

We tested three pre-trained models - YOLOv7, VGG16, and GoogLeNet - and evaluated their respective pros and cons to determine the best overall performance.

- **YOLOv7** is a highly advanced object detection model that is typically used for real-time object detection in videos but can also be used with static images[5]. We were particularly drawn to this model because of its fast inference speeds and versatility. However, YOLOv7 is a resource-intensive model, and there is often a trade-off between speed and accuracy. Therefore, we did not expect this model to perform the best in our study.

- **VGG16** is a simple, lightweight deep convolutional neural network architecture that has gained immense popularity for its transfer-learning ability. It has been designed to be lightweight, which makes it a suitable choice for various computer vision tasks[3]. This model's excellent performance on the ImageNet dataset, which is similar to the dataset we will be using, caught our attention. Therefore, there is a high probability that VGG16 will perform well in our study.

- **GoogLeNet** is a model that uses state-of-the-art inception modules to power its architecture. An inception module consists of convolution layers arranged in a specific layout, along with a max-pooling layer. This enables the network to capture multi-scale features by performing convolutions at multiple scales simultaneously2]. By doing so, the Inception module reduces the computational cost of feature extraction while improving the network's accuracy. As this model is relatively new, we are not certain what to expect from it, and we are primarily interested in exploring and analyzing its learning curve.

2.3 Justification

Our proposed approach provides a foundation for automated and centralized garbage sorting in Solid Waste Management Facilities, which can free ordinary people from the time-consuming and monotonous task of sorting waste. Furthermore, this approach can cut down on the overall financial costs of recycling and classification, thanks to its speed and accuracy. By avoiding human error, such an application can fundamentally prevent recycling contamination.

2.4 Literature

Our search for a meta-analysis comparing VGG16, GoogLeNet, and YOLOv7 was unsuccessful, and most studies have focused on simpler datasets. However, our study's use of a complex image dataset has practical applications in waste management and recycling, potentially reducing environmental impact. Therefore, our proposed model has significant potential to make a positive impact in waste management and recycling industries.

There are several waste management apps available that use AI and other technologies to assist users in sorting their garbage. Waste Wizard uses natural language processing, Recycle Coach uses AI and crowd-sourced data, and BinCam uses image recognition. Bin-e is a smart waste bin that sorts and compresses waste automatically, while HUAWEI-Trash-Detection-YOLOv5 is a non-commercial AI project that predicts garbage sorting results. However, these approaches are mostly focused on assisting users, while our approach aims to centralize the task and minimize human effort.

2.5 Adjustments

Due to time constraints and limited experience with fine-tuning convolutional neural networks (CNNs), our group opted to use three pre-existing image classification models in our project, which was a significant change from Part 1. We discovered that creating CNNs from scratch is a time and resource-intensive process. Therefore, we decided to apply transfer learning techniques to pre-existing models to tackle the problem at hand.

3 Methodology

3.1 Design & Training Pipeline

We will use pre-trained VGG16 and GoogLeNet models, which are implemented in **PyTorch**, for our image classification task. To preprocess the images, we will utilize the **RoboFlow** platform's built-in YOLOv7 model and tools.

Our first step is to convert our image folder into a PyTorch ImageDataset, which is then transformed into a list of tensors with shape $3 \times 256 \times 256$ (representing images with red, green, and blue color-channels). We avoid flattening the tensor at this point because we want to perform convolution operations before flattening them in the models themselves.

Next, we convert the ImageDataset into a DataLoader with a batch size of 32. During each epoch, we feed the model a batch of 32 images until the entire dataset has been consumed. We then split the DataLoader into training, testing, and validation sets. Our dataset contains 2,529 images, which we split as follows: approximately $\sim 75\%$ `taining`, $\sim 15\%$ `testing`, $\sim 10\%$ `validation`.

To optimize our models, we experimented with both Adam and Stochastic Gradient Descent (SGD) optimizers. After initial testing with the validation dataset, we found that SGD works best for VGG16, while Adam's optimizer performs best for GoogLeNet and YOLOv7. We will begin with 100 epochs for each model and increase the number as needed. We set the initial learning rate, α, to 0.05.

3.2 Model Training Changes for YOLOv7

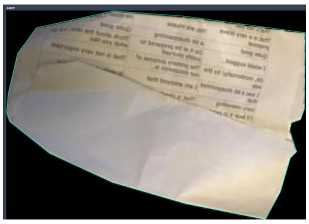


Figure 1: Removing background for YOLOv7 object detection model.

Since YOLOv7 is an object detection model, the backgrounds of our image dataset interfered with its accuracy. Therefore, we had to take out the background of each image and relabel then manually. This was a huge overhead for YOLOv7 model and the marginal improvement will be stated in the discussion section. The GoogLeNet and VGG16 models received the images unaltered.

3.3 Predictions

All three models used in this study are widely used in object detection applications. Therefore, we expect all three models to perform above our threshold value for accuracy of at least above 70%. We got this threshold by surveying few of the models on Kaggle build to tackle this problem and averaging their accuracies. However, out of the three we expect VGG16 to have the fastest training speed as it is a lightweight model with few parameters and have been show to perform particularly well for small objects and fine-grained classification. This would be useful to discriminate between objects such as plastic, glass and metal all of which are shiny and have similar appearance.

4 Results

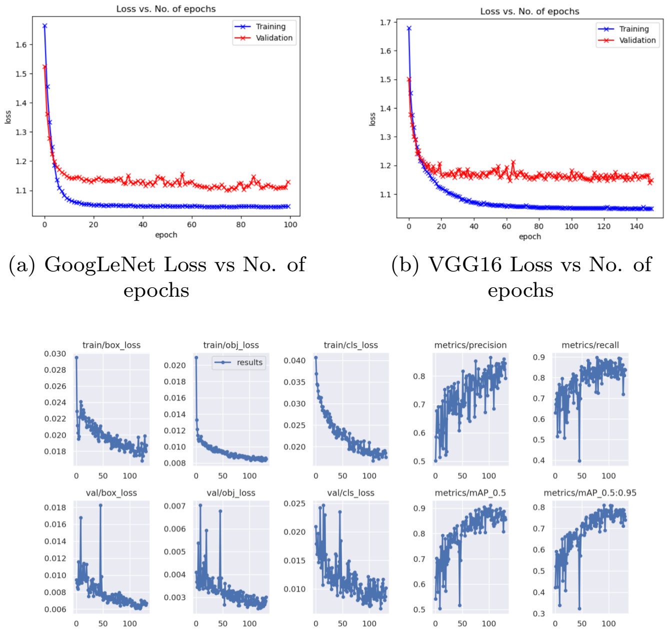
4.1 Baseline

The metric for evaluation in this classification problem will be accuracy. As mentioned before we have surveyed other models found on Kaggle to tackle this problem to get an average accuracy of 70%. It is difficult to predict how a human would perform on such a dataset, since we can ”cheat” by looking at the products and packing, gleaning extra context thus recognizing the material used easily. However, for this study we decided to put human accuracy at 99% which will serve as the accuracy ceiling.

4.2 Train, Validation and Testing Scores

Models	Training Loss	Val Score	Test Score
YOLOv7	0.020	90.0%	81.0%
VGG16	1.050	89.4%	88.0%
GoogLeNet	1.045	88.9%	94.0%

Above are the validation and testing scores of each model in this study. Below we show the losses validation and training losses of each model. Note, since this is a multi-class classification problem we are using cross entropy loss function.

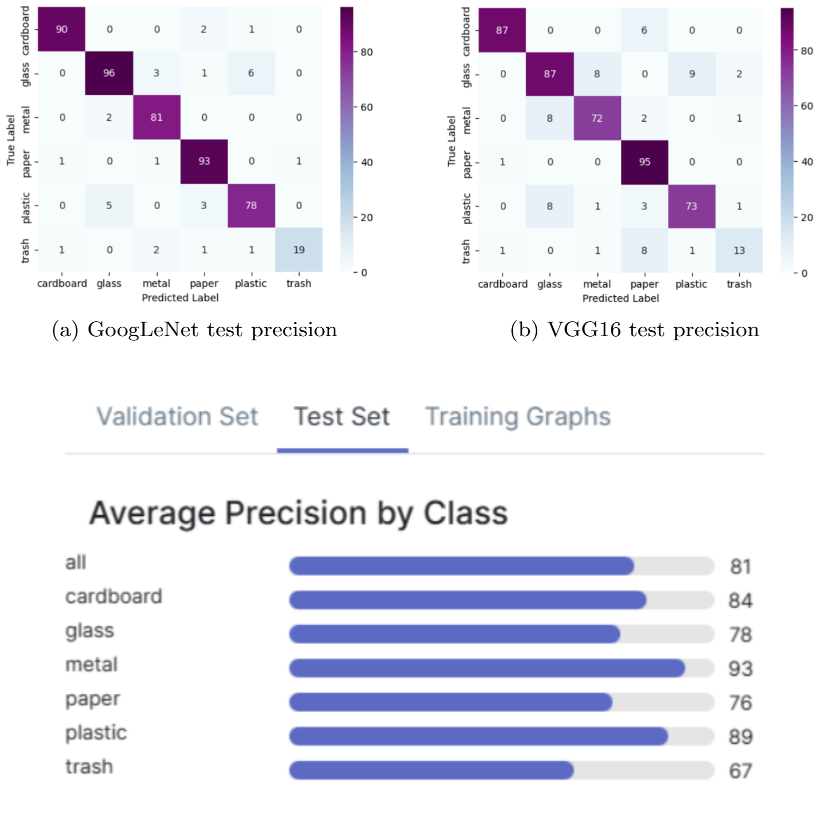


(a) GoogLeNet Loss vs No. of epochs

(b) VGG16 Loss vs No. of epochs

(c) YOLOv7 Loss vs No. of epochs

Figure 2: GoogLeNet has close train/val scores, VGG16 shows overfitting with far train/val scores, and YOLOv7 has fluctuating losses indicating the learning rate might have been too high.



(a) GoogLeNet test precision

(b) VGG16 test precision

Average Precision by Class

all	81
cardboard	84
glass	78
metal	93
paper	76
plastic	89
trash	67

(c) YOLOv7 test precision

Figure 3: There appears to be a wide variance in classifying images as trash, with all models struggling in this area. Specifically, all models seem to have difficulty distinguishing between cardboard and paper, as well as between glass and metal, which is to be expected. However, YOLOv7 performs better in classifying metallic objects, while GoogLeNet and VGG16 excel at classifying cardboard, glass, and paper. These observations suggest that different models may have strengths and weaknesses depending on the specific materials or objects being classified.

5 Discussion

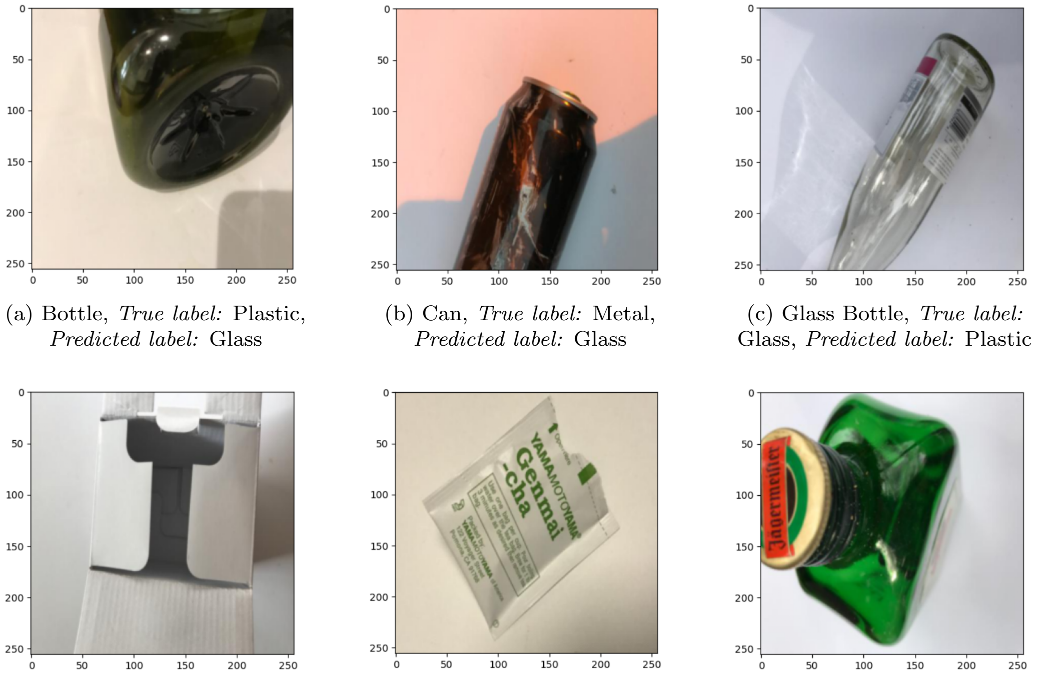
First, we define the criteria by which we will judge these models. Accuracy is our primary focus, as it is a key indicator of a model's usefulness. Additionally, we consider factors such as training speed, bias/variance trade-off, and model complexity as peripheral concerns. For general purpose classification, a model that is simple and faster to train would be ideal.

5.1 Strengths and Limitations

Based on these criteria, the significant overhead required for YOLOv7 makes it less suitable for image classification. While it is meant to be an object detection model, having to remove the background would impact its real-time performance. After removing the background, the accuracy did improve significantly, going from 62% to 90%. However, the YOLOv7 model took almost a day to train on our hardware, with training speed impacted by internet speed and the RoboFlow platform.

In contrast, VGG16 and GoogLeNet were much easier to train and had good validation scores even with pre-trained weights, particularly GoogLeNet. This was a huge advantage for GoogLeNet, as its accuracy was already high from the start, and 100 epochs were enough to achieve good accuracy. For VGG16, we had to increase the number of epochs to 150 and tune the learning rate multiple times. It was challenging to find the optimal learning rate for VGG16, as the initial value of 0.05 was too high, resulting in fluctuating losses. Ultimately, we found that a learning rate of 0.005 worked well for VGG16, while for GoogLeNet, smaller learning rates worked better, with a final value of $5x10^{-5}$.

Based on these findings, we conclude that GoogLeNet is the better model for this classification task. Despite its excellent performance, GoogLeNet still has some limitations.



(a) Bottle, *True label:* Plastic, *Predicted label:* Glass

(b) Can, *True label:* Metal, *Predicted label:* Glass

(c) Glass Bottle, *True label:* Glass, *Predicted label:* Plastic

(d) Box, *True label:* Cardboard, *Predicted label:* Paper

(e) Packaging, *True label:* Trash, *Predicted label:* Paper

(f) Jägermeister Bottle, *True label:* Glass, *Predicted label:* Plastic

Figure 4: Incorrectly labeled images by the GoogLeNet model.

The GoogLeNet model appears to struggle with correctly classifying Glass, Plastic, and Metal. Upon examination of the images, it is understandable why the model is making incorrect predictions. Even a human may make similar mistakes. For example, both Glass and Metal are shiny, and it can be challenging to distinguish between cardboard and paper without tactile feedback. Overall, these observations suggest that certain materials or textures may present challenges for the GoogLeNet model, as well as for human perception.

5.2 Future Directions

To further test the transfer-learning ability of these models, it would be necessary to evaluate their performance on a variety of datasets. Although our study is focused on recyclable image classification, some models may perform better with facial or shape recognition. Exploring these areas could be an interesting avenue for future research.

5.3 Peer Evaluation

We found the following feedback very useful:

- The most common feedback we received was to create subsection within each section to answer questions about scope/assumptions, justifications etc separately.
- Another valid feedback was that we failed to explain the application of our results and usefulness in the abstract and we didn't include training loss in our Results section.
- We also received a few feedback stating we should use more than one dataset to evaluate the models.

Changes we made based on the feedback:

- This feedback was very constructive and a quick fix. We feel that layout of the report is now much easier to read as well.
- We have further elaborated on the usefulness of our project with an example of a street roaming robot can help clean up recyclables left out in the open.

- This is very good point, we should have used more than one dataset for our evaluation. However due to time constrain this was infeasible to implement. However, to incorporate more datasets into our pipeline, not much has to change. In our research we found WasterNet which provides a plethora of labeled datasets of different materials. We can certainly incorporate this into our project given more time.

References

1. Garbage Classification Dataset from Kaggle
2. Richmond Alake; Deep Learning: GoogLeNet Explained
3. MathWorks VGG16 Documentation
4. Chien-Yao Wang, et al.; Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors
5. Manish Chablani; YOLO, You only look once, real time object detection explained