An Efficient Approach for Recyclable Waste Detection and Classification using Image Processing Techniques

by

Prabal Kumar Chowdhury
22241150
Md. Aminul Islam
19101398
Md Aminul Haque
19101580

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
School of Data and Sciences
Brac University
January 2023

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.

2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.

3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.

4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Prabal

Prabal Kumar Chowdhury
22241150

Aminul

Md. Aminul Islam
19101398

Aminul

Md Aminul Haque
19101580

Approval

The thesis titled "An efficient approach for Recyclable Waste Detection and Classification using Image Processing Techniques" submitted by

1. Prabal Kumar Chowdhury(22241150)
2. Md. Aminul Islam(19101398)
3. Md Aminul Haque(19101580)

Of Fall, 2022 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on January 19, 2023.

Examining Committee:

Supervisor:
(Member)

Md. Ashraful Alam, PhD
Assistant Professor
Department of Computer Science and Engineering
Brac University

Program Co-ordinator:
(Member)

Md. Golam Rabiul Alam, PhD
Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi, PhD
Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Ethics Statement

In conducting this undergraduate thesis, we are committed to upholding ethical standards by obtaining informed consent, maintaining confidentiality, following laws and regulations, conducting research with integrity and responsibility, and addressing any potential biases or conflicts of interest. This research paper has never been presented in full or in parts to some other university or other institution for the purpose of conferring a degree.

Abstract

One of the world's most pressing issues right now is the lack of a competent waste management system, particularly in emerging and underdeveloped countries. Recycling solid waste, which comprises numerous dangerous non-biodegradable substances like glass, metals, plastics, etc., is the most essential step in reducing waste-related issues in the environment. Typically, collected waste includes all types of waste that must be thoroughly sorted to recycle efficiently. Most countries use manual waste sorting techniques, which are efficient. Nevertheless, the waste sorting process by human being is not safe as there is always a risk of exposing themselves to toxic wastes, which could be serious for their health. Our thesis presents a Deep Learning technique based on computer vision for automatically identifying waste. To construct the model, we used Convolutional Neural Networks, real-time object detection systems, such as YOLOv5 and YOLOv7, as well as several transfer learning-based architectures, including VGG16, MobileNet, Inception-Resnet-v2. The model is trained on numerous images for each type of waste to ensure no overfitting and greater accuracy. The highest accuracy we achieved for our waste detection model YOLOv5x is 93.7%.

Keywords: TrashNet; Deep Learning; Object Detection; Image Classification; CNN; VGG16; Inception-Resnet-v2; MobileNet; YOLOv5; YOLOv7; Neural Network; Image Processing

Dedication

We dedicate this thesis to our parents, for their unwavering support, encouragement, and love throughout our life. Their sacrifices and constant belief in us have been the foundation of our success.

Acknowledgement

First and foremost, praise to the Almighty Allah for enabling the uninterrupted completion of our thesis. We would like to express our deepest gratitude to all those who have supported us throughout the completion of this undergraduate thesis.

Secondly, we would like to extend our sincere thanks to our thesis supervisor Dr. Md. Ashraful Alam sir, Assistant Professor, Dept. of Computer Science and Engineering, for his invaluable guidance, support, and encouragement. His expertise and knowledge in the field of image processing have been an inspiration to us, and We couldn't have completed this project without his constant support and motivation. We would also like to thank Shakib Mahmud Dipto, Research Assistant at the Department of Computer Science and Engineering at Brac University, for providing us with incredible suggestions and guidance throughout the research journey.

We would like to express our sincere gratitude to Late Hossain Arif sir who was our previously assigned supervisor, for his invaluable guidance and support during the starting phase of our thesis. His passing is a great loss to us and to the academic community. We will always be grateful for his contributions and his memory will be honored in our work.

Finally, we would like to thank our family and friends, for their unwavering support, encouragement, and understanding throughout this journey.

This research would not have been possible without the support and help of all these people, and we are truly grateful for their contributions.

Table of Contents

List of Figures

List of Tables

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

CNN Convolutional Neutral Network

ReLU Rectified Linear Activation Function

VGG Visual Geometry Group

YOLO You Only Look Once

Chapter 1

Introduction

The volume of trash generated, grows in parallel with the world's population and industrialization. Without proper disposal and trash management, these wastes wind up in landfills and our oceans. The lack of an effective waste management system is becoming a major concern around the world, particularly in developing countries [1]. To demonstrate the issue of waste piling in terms of numbers, the global population creates between 7 to 9 billion tons of garbage every year, among which 70% is mismanaged and disposed of in landfills, potentially damaging natural ecosystems [2]. As a consequence, this circumstance is causing numerous health and environmental issues, as well as a significant depletion of natural resources. Natural resources are limited all across the world. Therefore, to properly manage wastes we must recycle.

Plastic, glass, metal, paper, and other waste items have high recycling values. Even if their intended purpose has been served, these materials can be recycled and reused several times. 75 percent of waste is believed to be recyclable, yet we barely recycle only 30 percent of it [3]. Moreover, recycling plastics can assist in reducing the demand for new plastic products, recycling paper products can help save our forests, and recycling glass can help to limit the use of new natural materials like sand. Metals can also be recycled numerous times while preserving the majority of their properties [4].

However, classifying and sorting the collected waste into multiple waste categories is one of the most difficult aspects of recycling. The majority of the waste collected is cluttered with a variety of materials. To recycle, different types of materials must need to be separated first. It helps to recover useful materials and reduces the amount of waste transported to landfill. Recycling industries need to classify and recycle waste materials to regain value from them [5]. Therefore, the need for intelligence-based technology is increasing constantly in order to classify and sort waste materials.

The use of intelligent and automated classification and sorting systems offer advantages over traditional approaches since they can distinguish a huge number of objects in different waste classes in a short amount of time. Machine learning is a subfield of AI that is capable of solving this type of classification, prediction, and decision-making problems. Combining huge volumes of data with advanced learning algorithms allows machines to learn specialized and complicated tasks [6]. Machine learning can be highly effective for recycling plants because of its performance and scalability and can be used to classify various sorts of waste materials to improve

productivity.
In this research, we intend to develop deep-learning models based on computer vision that can classify and detect various sorts of waste items such as cardboard, glass, metal, paper, and plastic.

1.1 Waste Management

Waste management [7] refers to the various strategies for controlling and dealing with trash and garbage. The primary purpose of waste treatment is to manage the number of worthless products while also avoiding potential environmental and health concerns. Garbage can be cleaned, decreased, handled, regenerated, transformed, or monitored. Complimentary rubbish collection is often supported by different administrations. Waste is dealt with by a variety of techniques, including landfill compression and burning. Heavy waste is burned to prevent further spread by 85 to 90 percent [7], transforming it into gases, vapor, charcoal, and warmth. It is encouraged to adopt other methods, such as refining, recycling, and reusing the material. Biodegradable wastes are permitted to decompose so that they can be used as fertilizer or manure in farming, with methane into the atmosphere produced by microbial decomposition gathered and then used to generate energy to heat. For example, sewage is purified to produce sewage sludge, which may be dealt with via burning, recycling, or open dumping. Furthermore, waste management seeks to decrease the negative effects of garbage on public health and the environment. Municipal solid trash, which is produced by industry, commercial, and domestic activities, represents a large portion of garbage management.
Classification of waste is very important for recycling. The classification of garbage refers to how it is divided among various MSW categories. Even though most garbage generated in metropolitan areas falls into the MSW group, it is determined primarily by the type of the substance and its management methods. Waste can be classified into various groups, such as industrial waste, agricultural waste, municipal solid waste, hazardous waste, and so on.

1.2 Problem Statement

With the assistance of image processing and machine learning models, we would like to determine the types of recyclable waste that are currently present in the environment as part of our research. There have been researches conducted on how waste should be categorized, but in our opinion, the researches have a few weaknesses. Real-time systems typically only have a limited amount of data to work with, which can result in a low success rate. The current model for the categorization of garbage is insufficient because, the majority of the time, it will only classify a single waste item from an image. We intend to construct our model in such a way that it will be able to deliver accurate waste detections through the implementation of real-time object detection algorithms utilizing webcams or cameras that may be found on mobile phones. For some limitations on videos in a public place, we will also use the image classification models to predict waste to make a comparison with the real-time object detection algorithms. This can be achieved by increasing the number of training images and improving the algorithm's accuracy. We planned to have more than 8000 images included in our dataset so that we could improve our accuracy and more accurately detect recyclable wastes. As previous research has shown that it is possible to identify a single image at a time most of the time, Our primary objective is to train the models in such a way so that they are able to differentiate between different kinds of trash in a single image.

1.3 Research Objectives

The incorrect disposal of trash has become an international issue in recent years. Improper management of our daily life wastages contributes to global warming and poor air quality, along with harming our environments and wildlife. Landfills, which are the waste hierarchy's ultimate resort, release methane, a very powerful greenhouse gas that has a direct effect on our climate [8]. In addition, the garbage is treated only once it is collected. The transportation process generates co2, the most prevalent greenhouse gas, and also harmful emissions such as particulate matter, in the weather. Then there's the theory of recycling, which not only allows us to lessen our carbon footprints but also reduces the pressure on raw - materials collecting, helps to conserve, reduces and limits greenhouse gas emissions, and much more [9]. Recycling minimizes our mining operations, therefore conserving natural resources including gasoline, coals, petroleum, water, wood, and minerals [10]. The objectives of this research are:
1. To separate the trash into separate categories, including metal, plastic, cardboard, paper, and glass.
2. To detect multiple types of waste from a single image or real-time video.
3. By identifying the recyclable garbage helps people to reuse it after minor processing, therefore keeping the environment clean and reducing the quantity of waste created.
4. Increase our recycling activities to help keep the environment clean and our natural resources safe.
5. To create a more efficient and accurate model than the present models
After detecting and classifying the recyclable waste, we can process the wastage for reuse again in our environment. By recycling wood, glassware, polymers, and other materials, we may save money and energy while simultaneously decreasing the environmental impact of their extraction and processing.

Chapter 2

Related Work

A. Vogiatzis et al. proposed a unique picture classification approach that may be used to differentiate recyclable materials with ease [11]. The usage of a convolutional neural network has been proposed, which comprises two branches: recycling classification and plastic-type classification. Both the TrashNet dataset and the WaDaBa dataset were utilized throughout the training process of the architecture.

B. W. House et al. proposed a method for the purpose of automatically classifying polycoat containers along with PET (Polyethylene Terephthalate) bottles [12]. This approach does not get influenced by Near-Infrared spectrometry and instead uses a visible light camera. This research presents a high-speed approach for automatically locating and removing portions of the picture that are likely to include these items. These sections are united into whole containers, which are then classed as either polycoat containers or PET bottles. This is accomplished using a linear support vector machine (SVM) that has been developed on the histogram of pixel intensities. The proposed method, which makes use of a field-programmable gate array, was successful in achieving a recognition rate of 93 percent and is capable of running at high frame rates in real-time.

S. Thokrairak et al. proposed a method of automatic trash classification optimization using SSD-Mobile Net, a Convolutional Neural Network, in their research article ”Valuable Waste Classification Modeling based on SSD-MobileNet” [13]. (CNN). They worked on plastic, glass, and metal bottles and cans. Their dataset included 952 photos that were trained using a 24,000-step model. They achieved the best accuracy with a plastic bottle (95 %). Furthermore, this model may be suitable for usage in an embedded system.

C. Bircanoğlu et al. [14] presented a model based on transfer learning that is more flexible, faster, and produces valid results with reduced accuracy loss. The model is known as RecycleNet. They started from the beginning with a modest dataset of 1768 pictures of various forms of garbage. Moreover, they also worked on the dataset from TrashNet which comprises mostly 6 forms of waste. In their study, they analyzed a variety of Deep CNN-based architectures, including Xception, DenseNet, Inception-ResNetV2, ResNet, and others. They further explored various techniques for better optimization. During testing, the majority of models which are based on CNN were producing results with an accuracy of around 75 percent. Whereas models such as Inception-ResNetV2 produced results that were nearly 90 percent accurate. DenseNet architecture, which was a layered 121 architecture, was able to generate up to 95 percent accurate results after plenty of tweaking and calibration.

Shuteng Niu et al. addressed a solution to the enormous dataset required in trash classification in their study ”Transfer Learning-based Data-Efficient Machine Learning Enabled Classification ” [15]. Using information from pre-existing deep networks like AlexNet, DensNet, and ResNet, a robust model was constructed with a small amount of training data using transfer learning (TL) techniques. To generate a more precise domain distance measurement, they propose a novel domain loss function, Dual Dynamic Domain Distance. They were the first to employ TL to sort garbage. The DeepCoral-ResNet50 has a test accuracy of 96%.

The convolutional neural network VGG16 model is utilized in this study paper [16] as a way to investigate how deep learning may be put to use for environmental protection, with the specific goal of solving the problem of domestic garbage recognition and classification. Before pre-processing the images into 224 × 224 pixels RGB images that the VGG16 network accepted, the OpenCV computer vision library has been used to recognize and select the indicated entities. For this study, the training dataset set has 8300 images, whereas the test set includes 2300. Following data improvement, a VGG16 convolutional neural system is created using the TensorFlow framework, with the RELU activation function and a BN layer added to speed up convergence while preserving the recognition rate. Household garbage can be categorized as reusable, toxic, culinary waste, or other trash in this study. The trash classification system that relies on the VGG16 network proposed in this paper has a 75.6% accuracy rate after real-world assessment. This project's accuracy has to be increased when compared to other deep project-based learning that uses image recognition and classification. By classifying more types of trash in our daily lives, we may improve accuracy.

Recycling solid waste is an important step toward reducing negative consequences such as sanitary and health hazards caused by landfill misuse. Recycling, on the other hand, needs the separation of solid waste, which is both difficult and costly. In this paper [17] they provide a Deep Learning system that utilizes computer vision to automatically distinguish the kind of waste and classify it into five primary categories: cardboard, paper, plastics, metals, and glasses. Their suggested solution contains an intelligent recycling bin that opens the cover immediately based on the type of garbage identified. The study's main focus is on Machine Learning methods for efficient identification that can be taught. Pre-existing data were used to train at least 12 Convolutional Neural Network (CNN) versions spanning three classifiers for this study: SoftMax, SVM, and Sigmoid. Their data shows that VGG19 with the SoftMax classifier achieves a greater rate of accuracy than the other classifiers, at roughly 88%. Due to dataset restrictions, they were unable to categorize food waste.

The world economy and climatic equilibrium rely on waste recycling. As a result, recognizing recyclable garbage is a vital goal for humanity, which Deep Learning models can help with. They investigated well-known Deep Learning algorithms to determine the most effective method in this study [18]. The TrashNet dataset was processed with the Densenet121, DenseNet169, InceptionResnetV2, MobileNet, and Xception architectures, with Adam and Adadelta serving as neural network optimizers. Adam beat Adadelta in terms of the expected accuracy, according to the outcomes of this research. In addition, the data augmentation approach was implemented to improve the accuracy of categorization. This was done because the TrashNet dataset only had a small sample size. The better outcomes were identified in the DenseNet121 applying fine-tuning with a test accuracy rate of 95% as a result of the studies. The InceptionResNetV2 model was fine-tuned to produce a similar rate of success of 94%.

Due to the fast rise of municipal solid trash in recent years, a large part of municipal solid garbage has been sent outside of the city for incineration and dumping. This paper [19] presents a deep learning model that is based on the municipal solid waste categorization and recycling program. This model makes use of a convolution neural network to develop garbage intelligence at the same time. This was necessary because the traditional method of treating MSW is very slow, and the level of garbage classification is very low. The method is compared to the traditional BP neural network methodology. According to the simulation results, the new method is 30% faster than the previous way. This study also presents a migration learning strategy for enhancing the efficiency and accuracy of municipal solid waste identification in complex situations using the AlexNet convolutional neural network. This work also has a positive impact on trash picture identification and classification in a complicated context, demonstrating the algorithm's viability.

In this research paper [20], they provide a unique way of identifying typical waste products while they are processed on a moving belt in garbage collecting facilities in this research as recycling is not an automated operation; enormous volumes of waste materials must be handled manually and to manage the rising volume of waste materials at recycling plants, new and unique procedures must be employed. A method for detecting and classifying waste materials is suggested, which may be employed in real-world integrated solid waste management systems. This approach is based on a convolutional neural network trained on a unique dataset of pictures taken on-site from actual garbage collection belts. The findings of the studies demonstrate that the suggested system achieves a higher level of accuracy in real-world scenarios than any of the previously developed algorithms, reaching 92.43%. They have also suggested that we can accomplish reduced on-site installation cost and complexity by using pre-trained tiny embedded devices.

J. Kim et al. [21] implemented deep learning with the combination of a redesigned LeNet model for categorizing various recyclable waste items in a robot. With the help of an RGB camera, this robot gathers things and applies a Convolutional Neural Networks (CNN) based model to classify them. They focused their research on only two forms of waste: cartoon and plastic. During the testing phase, the waste classification model's total accuracy was 96 percent.[28] This model, on the other hand, was not designed to categorize objects in a congested environment. Furthermore, because the model was only trained to classify cartoons and plastics, it struggles with a wide range of materials and unfamiliar items, such as organic or glass.

G. Sakr et al. [22] presented a system derived from machine-learning techniques with the aim to separate and classify several types of trash that can identify waste types from colored 256 x 256 PNG images of waste. CNN and SVM had been utilized in order to classify waste and it allows the model to distinguish between three major waste groups: plastic, metal, and paper. They [22] got superior accuracy with SVM, which was 94.8 percent compared to 83 percent for CNN. Furthermore, SVM has demonstrated impressive flexibility for a variety of waste materials. Although, a significant downside of the model is that the training dataset is small.

To enhance the performance and minimize the shortcomings of CNN, a new architecture based on CNN was developed by K. Sreelakshmi et al. [23], which is called Capsule neural network also known as CapsuleNet. The CapsuleNet is mainly composed with capsules that imitate how the human brain accumulates pertinent data. They implemented CapsuleNet for managing solid waste materials, that involves segregating plastics from non-plastic garbage. They [23] worked on two datasets collected from the public (Dataset 1) and private (Dataset 2) domains. Overall accuracy after Capsule-Net deployment was 96.3 percent and 95.7 percent, respectively, whereas accuracy after CNN implementation was 95.8 percent and 93.6 percent, respectively.

Chapter 3

Research Methodology

3.1 Working Process

First of all, in order to keep track of the working procedures and manage them more effectively, we have built a workflow for the approach we provided. With the help of our proposed method, we hope to detect various types of trash materials in real-time. To do this, we have implemented the YOLO architecture, widely recognised as one of the most effective architectures for object recognition, into our system. Cardboard, Plastic, Glass, Paper, and Metal are the five types of recyclable garbage that we are detecting in this research. We will focus on YOLOv5 which is developed by Ultralytics because it gives the most stable performance. There are several versions of YOLOv5 like YOLOv5s, YOLOv5m, YOLOv5l, YOLOv5x etc. We will use these models to train our dataset and evaluate and compare their performances. We will also train on recently released YOLO models like YOLOv7 and YOLOv8 to compare the performance with all the models. We are going to train our custom model with the help of transfer-learning techniques, analyze how well it performs, and then utilize it to make inferences.

We prepared and annotated our dataset with the help of roboflow platform [1], which is free for public use. The labelling format for YOLO provides a single text file for each image's annotations. Every text document contains bounding boxes for all the waste in each image. Every annotation information is normalized and ranges from 0 - 1. The training setups are separated into 3 YAML files that are included with the repository of YOLOv5 from Ultralytics. We will modify these documents based on the tasks to meet our requirements.

Then, we will train our custom models with the required parameters. The trained model must be exported in a format that the object detection program can load and use. After training the model, the performance of the models on test images will be evaluated. Then, we will be able to draw inferences from images or videos and detect waste. This model is also capable of making inferences on live images. For this, we will implement OpenCV in our system. OpenCV helps us take live video footage from webcams or cameras and feed the footage for inference in order to detect waste.

To detect objects from a video, the video first needs to be divided into individual frames, which are individually analyzed. Then, each frame is pre-processed to verify

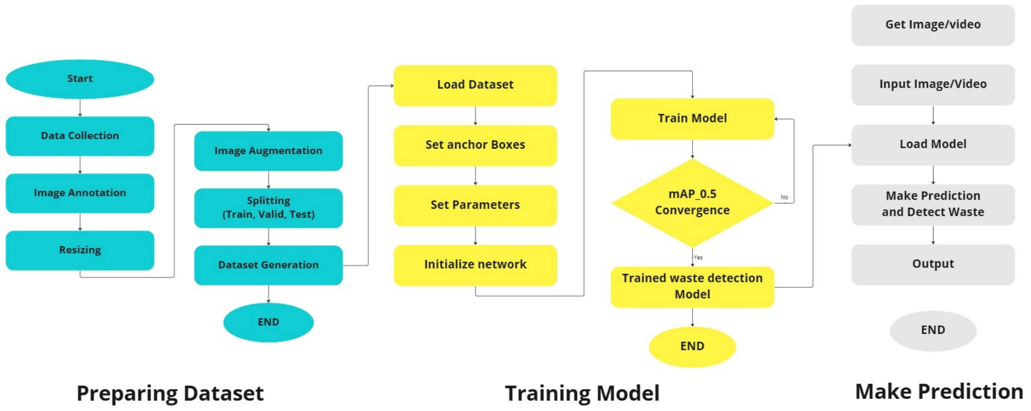

Figure 3.1: Working Process

that it has the necessary format and dimensions for the YOLO model. This may involve shrinking the image, turning it to grayscale, or adjusting the pixel values. The preprocessed frame is then fed into the YOLO model, which generates an output tensor comprising the bounding boxes and classification probabilities for each item in the frame. Post-processing is performed on the output tensor to interpret the anticipated bounding boxes and class probabilities. The post-processing step employs the concept of anchor boxes to enhance the precision of the bounding box predictions. Non-Maxima Suppression (NMS) is utilized by the algorithm to remove duplicate detections and accomplish real-time object detection. The final result is displayed with the detected items surrounded by bounding boxes and the corresponding class labels with confidence score displayed above each bounding box. The method is performed for each video frame, resulting in a series of frames in which the objects are recognized and labelled in real-time.

3.2 Computer Vision

In the domain of artificial intelligence (AI), computer vision is the study and construction of algorithms and models which make it possible for computers to comprehend and read images and videos from the real world. It aims to replicate the abilities of human vision in machines and enable them to perceive, understand, and interpret visual information. There is a vast array of usage for computer vision, including-

- Detecting and recognizing objects: detection and localization in still or motion images.
- Face recognition: identifying individuals in images or videos.
- Image processing: enhancing, restoring, and analyzing images.
- Robotics: enabling robots to navigate and interact with their environment.

- Medical imaging: diagnosing and treating patients with the help of image analysis.
- Augmented and virtual reality: creating immersive experiences.
- Deep learning methods, such as CNN, are frequently used in computer vision systems and models because of their ability to simulate the way the human visual system functions.

3.2.1 Image Classification

Assigning a class or category to a source image according to its visual characteristics is the goal of image classification. This is also a part of computer vision. Feature extraction, object classification, and image interpretation are just a few of the many areas where it is brought to use, making it one of the most essential objectives in computer vision. The traditional approach to image classification is based on handcrafted features and a linear classifier, such as SVM (Support Vector Machines), Random Forest, and other similar algorithms. However, image categorization has achieved a major leap in precision and speed with the introduction of deep learning.

There are several popular architectures for image classification, such as VGG, ResNet, and Inception, that have been developed over the years and have been used for different image classification tasks. Pre-trained models of these architectures are available and can be fine-tuned on new datasets to improve performance.
Overall, image classification is a well-studied task in computer vision, and deep learning has greatly improved the performance of image classification models. With the increasing availability of large annotated datasets and powerful computational resources, image classification models are becoming more accurate and widely used in various applications.

3.2.2 Object Detection

In the domain of computer vision, object detection refers to a technique used to identify and localize particular types of content inside an image, whether that content is stationary or dynamic. It's a core of computer vision and finds widespread use in things like autonomous vehicles, security devices, and visual analytics. Although there are many varieties in object detection methodologies, they can be roughly divided into two classes:

- Two-stage algorithms: These algorithms first generate a set of region proposals and then classify the objects within these regions. Examples of two-stage algorithms include R-CNN and Faster R-CNN.
- Single-stage algorithms: In a single step, these algorithms anticipate object positions and corresponding labels. You Only Look Once (YOLO) and Single Shot MultiBox Detector (SSD) are two instances of single-step algorithms.

One of the most popular and widely used approaches for object detection is the implementation of deep learning algorithms, in particular CNNs and R-CNNs (Region-based CNNs). These models are able to learn rich feature representations from large amounts of annotated data, and they have been shown to be highly effective in a wide range of object detection tasks. Improved performance and faster speeds in object recognition have recently received much attention for the introduction of new architectures including RetinaNet, FPN, and EfficientDet. Overall, object detection is a challenging task, but there has been remarkable development in the last few years. thanks to the development of powerful deep-learning techniques. New advancements in the field will continue to improve the accuracy, speed, and robustness of object detection models, making them useful for a wide range of practical applications.

3.3 The Architectures

In this research, we will utilize several deep neural network-based architectures. For example, for detecting waste from videos in real time, we utilized YOLO architecture. We have trained our dataset on four versions of YOLOv5 and YOLOv7. There may be situations where video features might not be available. In that case, YOLO models would also be able to identify waste from images. And to compare how well YOLO models perform to identify waste from images, we have used several state-of-the-art image classification algorithms like VGG16, InceptionResNetV2, and MobileNet.

3.3.1 Convolutional Neural Network (CNN)

When it comes to neural networks, CNN is by far the most well-known and widely-applied architecture. CNN's main benefit over its predecessors is that it can recognize these crucial features automatically, with no human involvement [24]. CNNs architecture is similar to classic neural networks and was inspired by the neuron synapses present in human brains. So CNN is a network that contains multiple convolutional layers and few other important layers. Multiple filters are applied to the input in the convolutional layer, where the convolutional operation takes place. Fully connected layers and CNN differ in a variety of ways that make this algorithm more effective. A fully connected neural network's complexity can be reduced via CNN in two different ways: by decreasing the amount of interconnections, using shared weights for the edges and the max pooling layers between the convolutional layers further decrease the complexity. In a CNN model, each input layer x is structured with height, weight and depth, these three dimensions. We can represent these as $w \times h \times d$. Grayscale images have a depth of 1, whereas RGB images have a depth of 3. As a result, this depth also represents the quantity of image channels. Each convolutional layer includes a set of filters, k, that are present. Much like input image, all filters also consist of three dimensions. The depth can be lower or equal to the input image, while the height and weight must be less than the input image. In the convolution layer, we produce feature maps by doing dot multiplication between the input layer and the filters. The quantity of filters used equals the number of generated feature maps. The output dimension of conv layer can be calculated using the given formula: For input dimension $W1 \times H1 \times D1$, the output dimension will

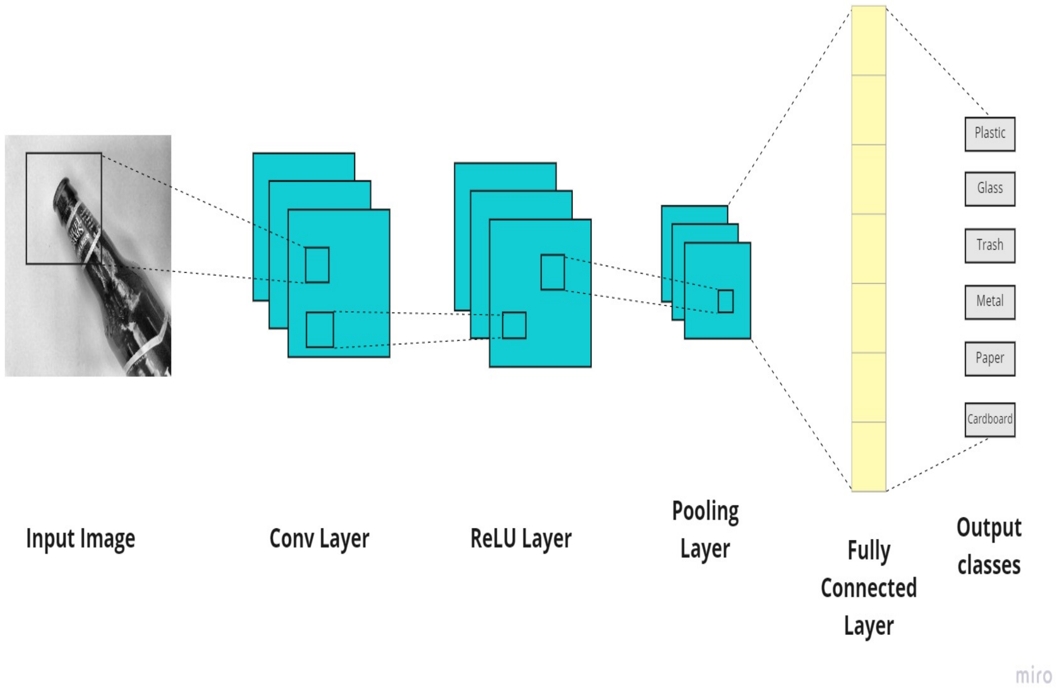

Figure 3.2: Image classification Using CNN

be $(W2 \times H2 \times D2)$:

$$W_2 = (\frac{(W_1 - F + 2P)}{S}) + 1 \tag{3.1}$$

$$H_2 = (\frac{(H_1 - F + 2P)}{S}) + 1 \tag{3.2}$$

$$D_2 = k \tag{3.3}$$

Now, in Eq. (3.1), F represents spatial extents, P denotes the zero-padding, and S indicates the stride. In Eq. (3.3), k represents the count of filters. After getting output from a conv layer, it goes as input in a max pooling layer where the dimensions get further reduced to half. So the output dimension of max pooling layer is $(W3 \times H3 \times D3)$:

$$W_3 = \frac{W_2}{2} \tag{3.4}$$

$$H_3 = \frac{H_2}{2} \tag{3.5}$$

$$D_3 = D_2 \tag{3.6}$$

After repeating these convolutional layers and max pool layers a few times, we flatten the inputs and finally feed these to a fully connected network for the final output.

3.3.2 YOLO Architecture

YOLO or You Only Look Once is an architecture that can detect objects in real-time, was developed in 2015 by Joseph Redmon and Ali Farhadi [25]. The defining aspect for YOLO is its ability to process an entire image in one forward pass within the network, making it much quicker than other object detection architectures such as Faster RCNN and RetinaNet. The key factor behind its popularity is that it utilizes one of the finest neural network architectures to provide high precision, accuracy and high information processing. The YOLO algorithm aims to identify the type of object present in an input image and determine its bounding box [26]. It uses four parameters to identify each box's boundaries: Box dimensions (height and width) and center (X, Y) coordinates. Now, how does the YOLO works? Let's see an example. The image contains two bounding boxes, one for each of the plastic and metal in the picture. A grid is created to organize the image as the initial step in YOLO's process. We have shown an example in Figure 3.3 using a 3x3 grid:

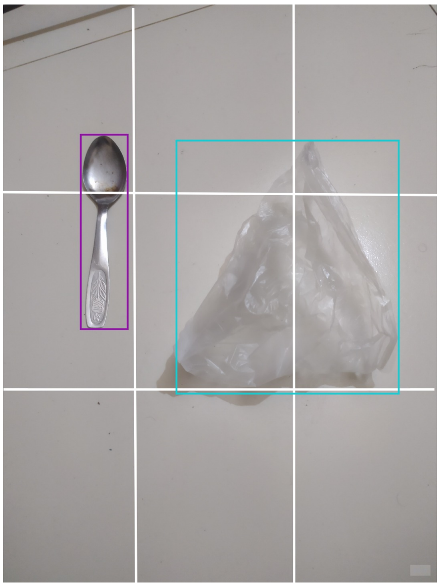

Figure 3.3: Grided Input Image

With a grid in place, we can now identify multiple objects in a single grid cell, rather than just one in each image. As an alternative to using labels, we may encode a vector which acts as a description for each grid cell. For example, we say that the bottom-left cell (which contains nothing) is:

$$\begin{aligned} Cell_{3,1} &= (P_c, B_x, B_y, B_h, B_w, C_1, C_2, C_3, C_4, C_5) \\ &= (0, ?, ?, ?, ?, ?, ?, ?, ?, ?) \end{aligned} \tag{3.7}$$

Here, in Eq. (3.7), the object class probability is denoted by P_c, the x and y coordinates of the bounding box's center are represented by (B_x, B_y), the bounding box's width and height are represented by (B_w, B_h), and the values of $(C_1, C_2, .)$ are either 0 or 1 depending on whether the bounding box represents a plastic or a metal or other waste classes. Here, in $Cell_{3,1}$, $P_c = 0$ because there is no waste in that cell. As $P_c = 0$, other values are not taken into consideration. Now if we check the top-left cell, which contains metal, the vector we will get is:

$$Cell_{2,1} = (1, 0.6, 0.3, 0.5, 0.2, 0, 1, 0, 0, 0)$$

Using this method, the image can be described by nine vectors of size 10, or a 3x3x10 tensor, if one vector is defined for each grid cell. Therefore, within the dataset, every image is tagged with a single 3x3x10 tensor. We will be able to train the CNN

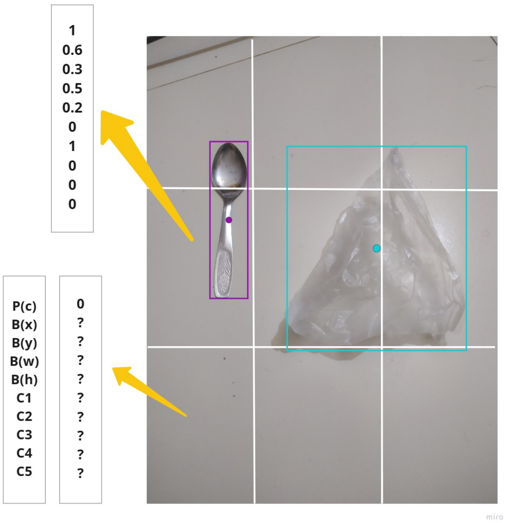

Figure 3.4: Vectors in different cells

utilizing this set of data and construct a training, valid, and test set.

The YOLO neural network architecture is divided into three parts- Backbone, Neck, and Head. The backbone consists of several convolutional layers which extract the features from the input image. The extracted feature map is then passed on to the Neck part which also consists of some conv layers. This part creates feature pyramid from the feature maps. Finally, it is passed on to the Head or Dense prediction part where the main prediction occurs. It renders the finished output, which includes bounding boxes, confidence scores, and classes, by applying anchor boxes to feature maps. This whole process of prediction is done in a single pass through the network.

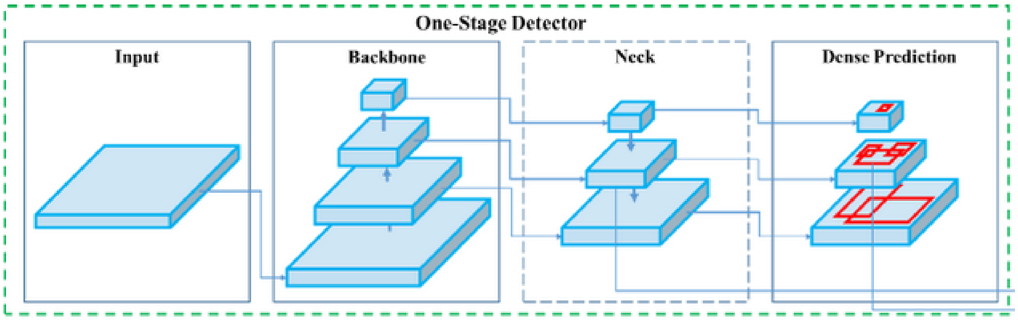

Figure 3.5: YOLO Architecture [27]

YOLOv5

YOLOv5 is the 5th version of YOLO, which is also a real-time object detection model developed by the team at Ultralytics. It is a single convolutional neural network that can perform object detection on images and videos. YOLOv5 is an improvement over previous versions of YOLO, with a focus on speed and efficiency while maintaining high accuracy.

For the purpose of determining bounding boxes as well as class labels for objects

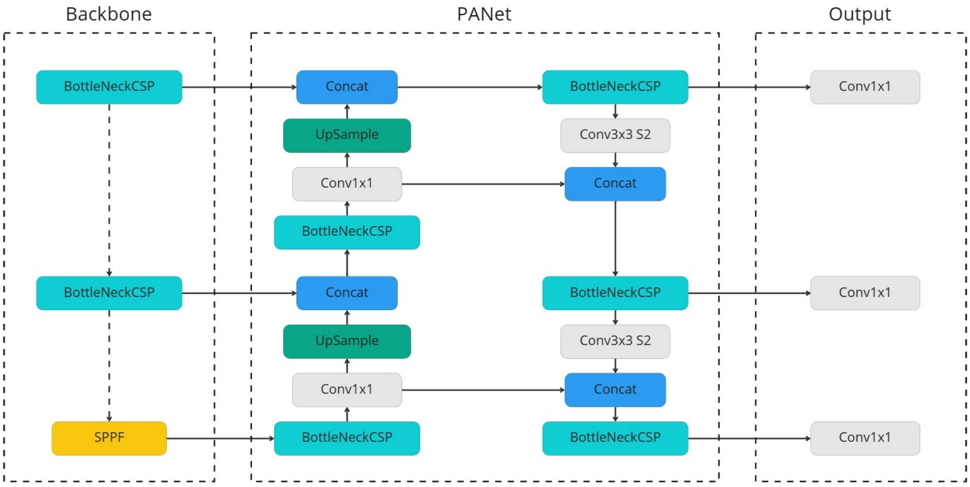

Figure 3.6: Overview of YOLOv5

present in an image, YOLOv5 utilizes one convolutional neural network. The model utilizes anchor boxes to increase object detection performance and was trained on MS COCO, a big dataset of annotated images with 80 classes and over 328,000 images of humans and daily objects.

The key difference between this version and the previous one is the usage of CSP-DarkNet53 as the backbone. Darknet-53 served as the foundation for YOLOv3. CSPNet (Cross-Stage Partial Network) has been added on top of that. By shortening the information flow, CSPNet assists in resolving the ”Redundant Gradient” issue brought on by Residual and Dense blocks. The neck of YOLOv5 saw two significant modifications. First of all, a version of SPP (Spatial Pyramid Pooling) more accurately SPPF has been employed to increase the inference speed, and by adding the BottleNeckCSP, PaNet (Path Aggregation Network) was transformed. In order to provide a consistent output length, the SPP block aggregates the data it gets from the inputs. To aid in the mask prediction problem, PANet is a feature pyramid structure that enables efficient data flow and localization of pixels. Then, the head consists of three conv layers which finally detect the object, class, and bounding box. The activation function SiLU was used in the hidden layers to help the convolution tasks and in the output layer, the Sigmoid function was utilized.

It can also detect multiple objects in a single image and can run on a variety of platforms, including smartphones and edge devices. In terms of performance, YOLOv5 has a high frame rate and can process up to 60 frames per second on a single GPU. It also has a relatively small model size, making it suitable for devices with limited memory and computational resources.

YOLOv7

YOLOv7 is a real-time object detection algorithm developed by Alexey Bochkovskiy in 2021. It is an improved version of the YOLO (You Only Look Once) algorithm, which is known for its speed and accuracy. There four major changes introduced in YOLOv7. The first architectural reformed that has been introduced is E-ELAN (Extended Efficient Aggregation Network) as the backbone of YOLOv7. Feature map learned from this type of network architecture is more efficient. Then the next

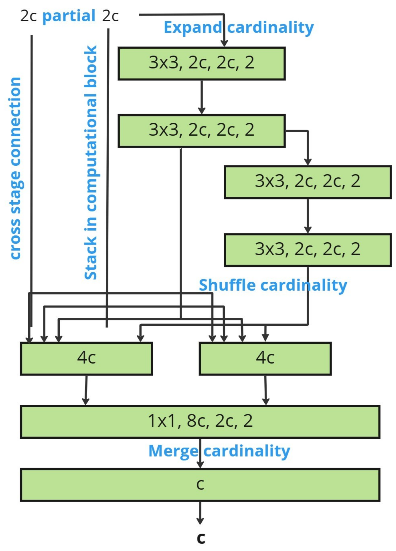

Figure 3.7: E-ELAN (Extended Efficient Aggregation Network)

architecture reformed is compound model scaling. The goal of model scaling is to adjust the attribute in order to generate models of various scales. For the training set

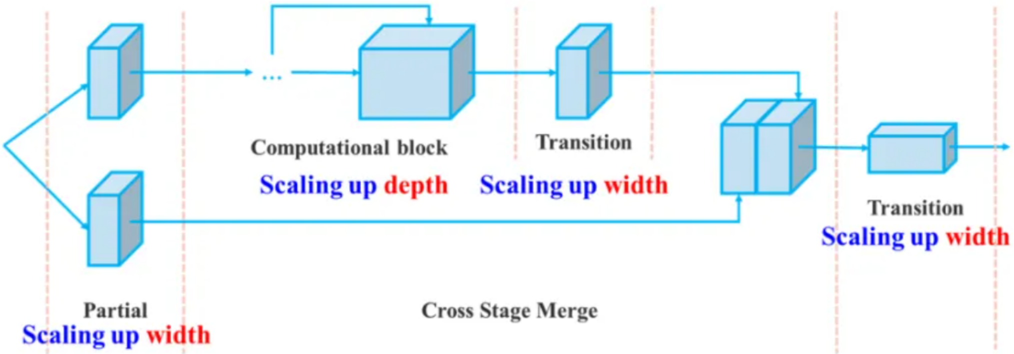

Figure 3.8: Compound model scaling in YOLOv7 [28]

bag of freebies has been introduced. Re-parameterization, a method for enhancing the model post-training, has been introduced. It extends the overall training period but accelerates inference. We know that YOLO architecture has a head part that predicts the object and gives the final output. However, YOLOv7 has two heads instead of one. The head which generates the final output is the Lead head and The auxiliary head helps the model while training in the hidden layers. YOLOv7 also uses a new loss function called CIoU (complete intersection over union) loss, which improves the accuracy of bounding box prediction. Additionally, YOLOv7 uses a technique called ”cross mini-batch normalization”, which helps to reduce overfitting.

Predicting bounding boxes and class probabilities for objects found within an image is accomplished by YOLOv7 through the use of a single convolutional neural network. The algorithm’s ability to handle small objects and its overall accuracy are both improved as a result of these changes. In addition, YOLOv7 makes use of something called anchor boxes. These are pre-defined bounding boxes that are employed in order to improve the prediction of objects of varying sizes and shapes.

3.3.3 VGG16

VGG16 is a neural network made up of 16 layers that is based on CNN and widely used because of its uniformity. The Imagenet dataset, that comprises 14 million photos, was used to train this model. Each of the configurations takes as input a 224px x 224px photo with three color channels called RGB [29]. The only thing that is performed before the image is utilized is to normalize the RGB values assigned to each pixel. The image first goes through the initial set of two conv layers. After this the ReLU activation is done. This first bundle of conv layers consists of 64 filters each. For the convolution operation, the stride and padding are both set to 1 pixel. With this setup, the spatial resolution is maintained, as well as the shape of such activation maps from output is identical to the shape of the input data. The map then moves to the layer with the most pooling, with the stride amount of 2 pixels. So the map size becomes half of the previous size. After that, the information passes through another set of 2 conv layers and a max pool layer, which is similar to the first set. However this time, there are 128 filters, which is twice as many as before. The data then goes through 3 bundles of conv layers. There are 3 conv layers and one max pool layer in each of the bundles. For each bundle, the filter was doubled then the one before it. Finally, the final output from these bundles is 7 × 7 × 512. After all of the convolutional and max pooling layers, the output is flattened and sent through three fully connected layers. Each of the initial two fully linked layers contains 4069 neurons. The final fully connected layer, with 1000 neurons, is the output layer. Following the output layer is the SoftMax activation layer, that is used for categorical classification.

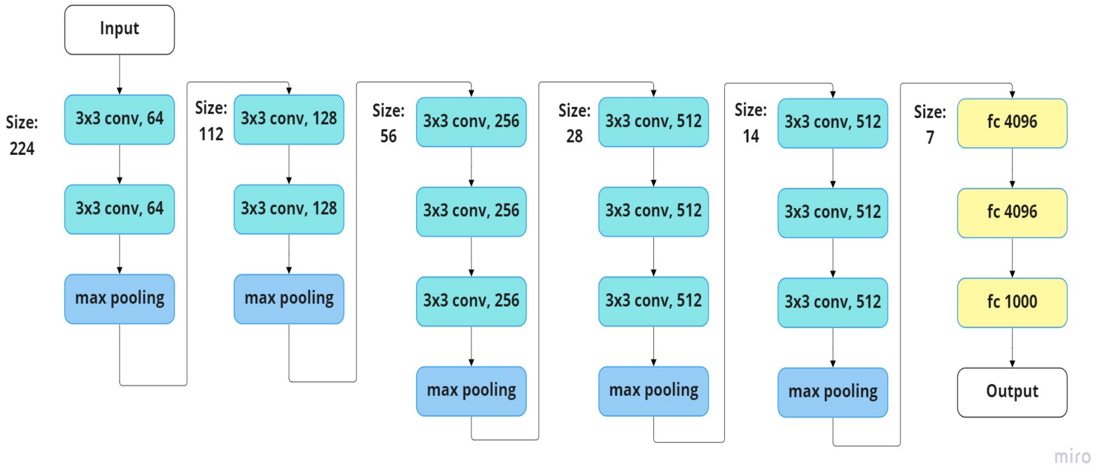

Figure 3.9: Information flow through VGG16

3.3.4 InceptionResNetv2

Inception-ResNet-v2 is a CNN architecture which was introduced by Google researchers in 2016. This was learned using the 14 million images and 1000 classes available in Imagenet. The network architecture consists of 164 layers. Inception-ResNet-v2 combines the Inception architecture with the residual connections introduced in the ResNet architecture. Using the residual units allows for more Inception blocks and hence more depth inside the network. Most of the difficulties with really deep networks occur during training, and this is where residual connections come in [30]. Whenever a significant amount of filters are applied in the network, the residual is downscaled as a convenient method of addressing the training issue. Whenever the amount of filters is greater than 1,000, disturbances occur in the residual variations, and it becomes impossible to train the network. As a result, network training is more stable when the residual is scaled. Figure 3.10 shows a compressed view of the schematic diagram of InceptionResNetv2

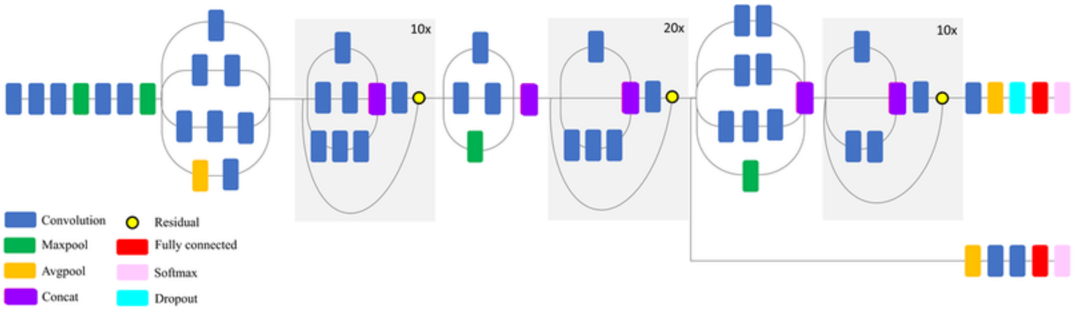

Figure 3.10: Schematic diagram of InceptionResNetV2 model [31]

The Inception architecture is known for its use of ”Inception modules,” which are sub-networks that are used to extract features from an input image at different scales. These modules use a combination of convolutional, pooling and other layers to extract features that are then concatenated and passed to the next layer. The Inception-ResNet-v2 architecture uses these modules for feature extraction from a source image and uses residual connections to combine these features with the output from previous layers. Inception-ResNet-v2 also uses batch normalization, which is a technique that helps to reduce the internal covariate shift and accelerate the training of deep neural networks. Additionally, factorized convolutions are used, which minimize the model’s computation and parameter count.

3.3.5 MobileNet

MobileNet is a deep CNN-based architecture developed for efficient on-device image classification. It is specifically built to run well on mobile and embedded devices, such as smartphones and small embedded computers. MobileNet consists of 27 convolutional layers. Among them 13 layers are 3 × 3 Depthwise convolution layers, 13 layers are 1 × 1 convolution, and one 3 × 3 convolution layer. There are also one average pool layer, one fully connected layer, and one Softmax layer for classification.

It takes a 224 × 224px image as input. The input layer takes in the image and scales it down to reduce computational costs. The first layer is a depth-wise separable convolution layer that performs a spatial convolution on all the channels of the source image distinctively.

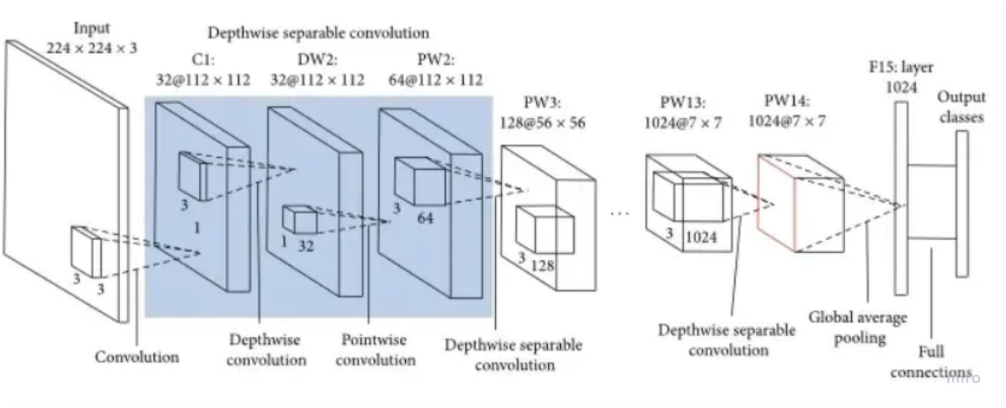

Figure 3.11: MobileNet Architecture [32]

The following layers are a series of depth-wise separable convolution layers, further reducing the number of parameters and computation. The convolution layers are used to increase the number of filters. The final layers of the architecture consist of fully connected layers, which are used for image classification.
MobileNet is a lightweight model, and it can be used for object detection and semantic segmentation as well as image classification tasks. The architecture is designed to be computationally efficient, making it well-suited for deployment on mobile and embedded devices.

Chapter 4

The Dataset

4.1 Dataset collection

After conducting some research, we came to the realization that will collect our waste data from different sources. At first, we used the waste pictures from the trashnet dataset. There were a total of 2527 images, which included pieces of paper, cardboard, glass, and metal, along with garbage. We did not make use of each and every image that was included in this dataset. We decided to take those pictures only that we felt would be useful to us. We added images from the Kaggle website that were related to garbage. We used certain images from the TACO dataset in our attempt to identify multiple types of garbage from a single image. Lastly, we took almost 300 waste images from our home and made a dataset with all these images.

Finally, our dataset contains 8373 images of waste. Plastic, cardboard, glass, paper, and metal are the five different types of waste.

4.2 Source

For our research, we primarily utilized the TrashNet dataset [33]. We have also used the Kaggle Waste Classification [34] dataset. For multiple types of waste from a single picture we used some pictures from the TACO [35] dataset. Additional real-time data from other sources including our household garbages were also incorporated.

4.3 Data Classification

For real-time object detection, we annotated 8373 images on the roboflow website. Then we generated the train, test, and validation set. The train set contains 5860 images, the valid set contains 1673 images, and the test set contains 840 images. 70% of the images are in the train set, 20% of the images are in the validation set and 10% of the images are in the test set. There are many images where there are multiple types of waste. From table 4.1, we can observe the train, test, and validation set according to all the classes. For the image classifier algorithms we used 8250 images where each class has 1650 images. We didn't use those image here where there are multiple types of waste in a single picture. For training, each class has 1320 images and for testing, each class has 330 images.

Class	Train	Validation	Test
Cardboard	1117	317	157
Glass	1184	290	113
Metal	1135	371	220
Paper	1215	344	172
Plastic	1274	365	191
Total	5860	1673	840

Table 4.1: Classification of our Dataset

4.4 Data Sample

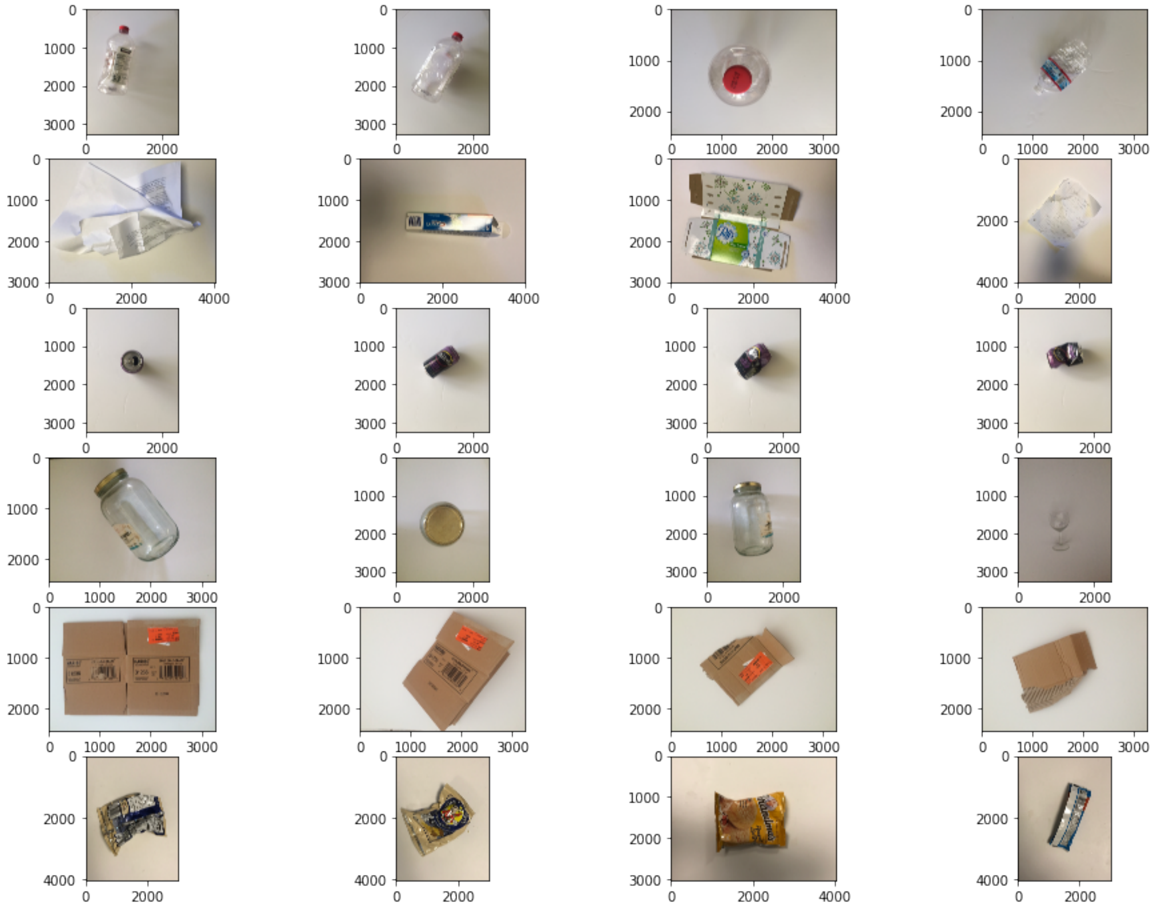

Figure 4.1: Sample Data

In the figure 4.1, the first three row images represent images of plastic, paper and metal. The bottom three row images represent the images of glass, cardboard, and trash.

4.5 Data Preprocessing

In order to make it easier for the computer to analyze and train the model effectively, dataset preprocessing is an important stage in the data analysis process. It involves transforming the original data from the actual dataset into different forms and types in accordance with the needs of the model. Our dataset contains images of different categories of waste. The majority of cases, there is no consistent size and design for the image. We are aware that computers can only read 0s and 1s and prefer to read nicely structured information. Therefore, before analyzing the data, non- numeric data like images needs to be prepared and cleaned. Data preprocessing for YOLO object detection models typically involves several steps like annotating the images in the dataset with bounding boxes around the objects of interest, converting the annotations into a format that the YOLO model can use, such as a .txt file or a .xml file, resizing and augmenting the images and finally splitting the dataset into a training set, a validation set, and a test set.

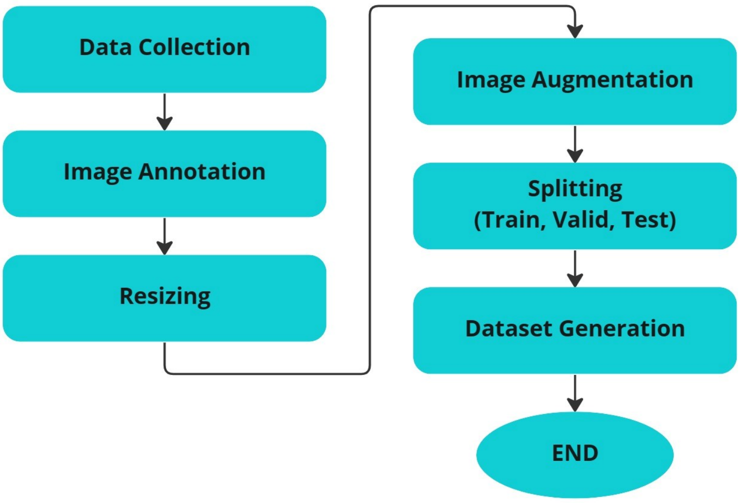

Figure 4.2: Data Preprocessing Flowchart

4.5.1 Annotation

Annotation is the technique of labelling the objects of interest inside an image or video In order to train an object detection model like YOLO. Bounding boxes are created around the objects of relevance as part of the annotation process, and each bounding box is given a label or class. For object detection models, annotation is an essential step since it gives the model the data it needs to figure out how to recognize and locate objects in a picture. The bounding boxes around the objects provide the model with the coordinates for where those objects are in the image. The model wouldn't be able to effectively detect and locate items inside an image without this information. Additionally, the model may learn to detect a wide range of objects in various contexts and scenarios by being given a vast and varied set of annotated images. One annotation text file per image is provided by the majority of annotation platforms when exporting in the YOLO labelling format. Every text document contains bounding boxes for all the waste in each image. Every annotation information is normalized and ranges from 0 - 1. We used the Roboflow platform to perform annotation on the images. All of the images were annotated with bound

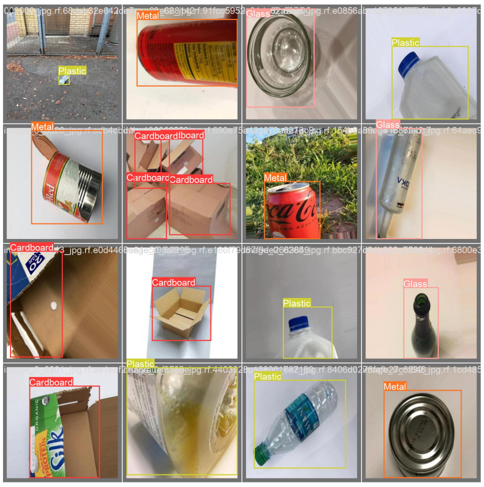

Figure 4.3: Annotated Images

boxes enclosing any waste that was visible. After annotating all the images, we can now move to resize and augment the images and finally create the dataset.

4.5.2 Resizing Images

The majority of the images in our raw dataset have irregular sizes. We need to resize the images in order to efficiently and effectively train the models. The sizes of each image should be uniform. In our circumstances, we fixed the image resizing to a 1:1 aspect ratio. When preparing an image for object detection using YOLO, it's essential to ensure that the image is resized to the correct size for the model. The size that the image needs to be resized to will depend on the specific architecture of the YOLO model being used. Generally, YOLO models expect the input image to be a square image with a specific width and height. The input image size should be a multiple of the number of grid cells used by the model. For our case, we used the image size of 416px × 416px. After annotating the images, we resized all the images to 416px x 416px size.

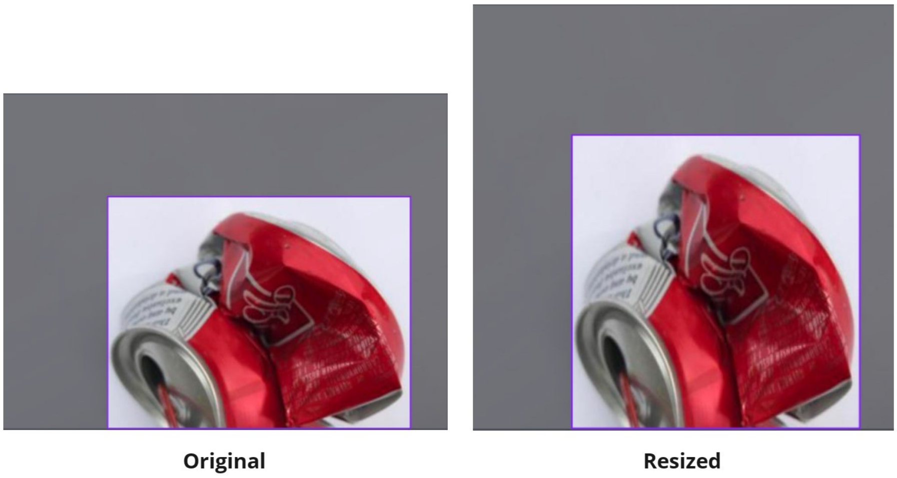

Figure 4.4: Resized Images

4.5.3 Augmentation

Image augmentation produces training images artificially by using multiple processing techniques or a mixture of several processing techniques, like random rotation, shear, shifts, flips, and so on [36]. This process is the transformed version of the previous images of the training set. Data overfitting can be eliminated with the aid of image augmentation. It improves the model prediction precision and data model generalization ability [37]. We set the rotation range 40, shear range 0.2, brightness 0 - 25 percent and exposure -25 - +25 percent. We set the value True for horizontal flip as it will help to flip both rows and columns horizontally.

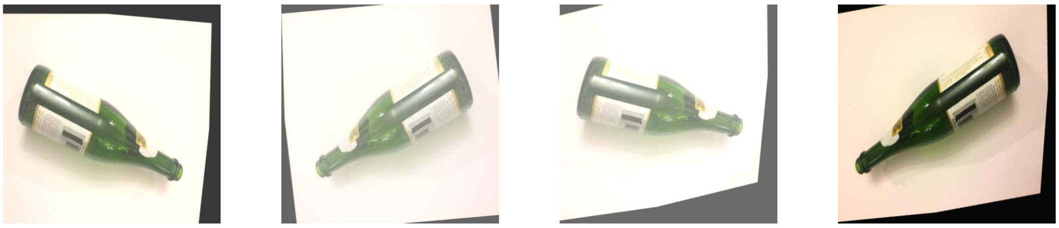

Figure 4.5: Augmented Images

4.5.4 Splitting

To evaluate how well our image dataset works for our machine-learning model, we divided the image dataset into the train, valid and test sets. The train set will be used to train our desired model valid set will be used to validate the model while training and the test set will be used for giving an accurate prediction. We split our images into 70:20:10 ratios for training, validation and testing. The image we used for our training set is not used again for the testing set. Test images are absolutely new images for the model and help to evaluate the model. In this way, we split our image dataset.

Chapter 5

Result & Analysis

5.1 Experimental setup

We have done our experiments with the help of google colab. Here are the hardware configurations.

CPU	Intel(R) Core(TM) i7-8700 CPU @ 3.20GHz
GPU	GeForce GTX 1650 Super
Storage	256 GB SSD
RAM	16.0 GB

Table 5.1: Hardware Configurations

5.2 Evaluation of Models

In this section, we are going to discuss how the performance of our trained model has developed over time. When testing our YOLO models for usage in real-time object recognition, we focused on confusion metrics, mAP, recall, and precision. For comparison, we also used other popular image classifiers such as VGG16, InceptionResNetV2, and MobileNet and evaluated their performance through accuracy, precision, and recall.

5.2.1 Confusion metrics

A confusion matrix is a chart that shows which classifications were accurate and which were wrong. There are four parameters in a confusion matrix.
TP(True Positives) - These accurately predicted positive values imply that both the actual class value as well as the expected class value are true.
TN(True Negatives) - These accurately predicted negative values imply that both the actual class value as well as the expected class value are false.
FP (False Positives) - These imply the value of the actual class is False but the value predicted class is True.
FN (False Negatives) - These imply the value of the actual class is True but the value predicted class is False.

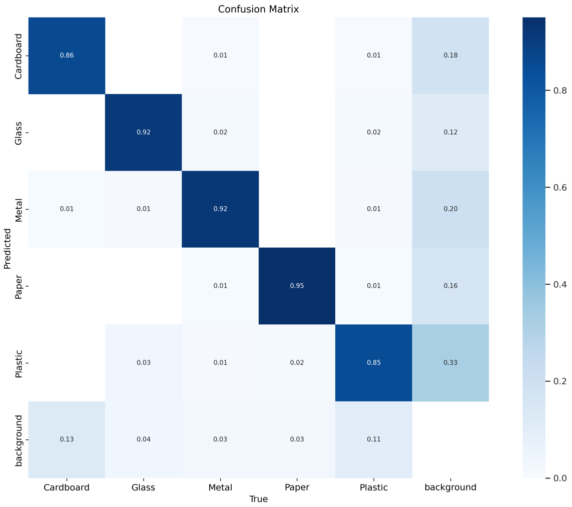

Figure 5.1: Confusion metrics of YOLOv5x Model

In Figure 5.1, we can observe the True Positive value of Cardboard, Glass, Metal, Paper, and Plastic is 0.86, 0.92, 0.92, 0.95, and 0.85 respectively.

5.2.2 mAP

The full form of mAP is mean average precision. It is used to figure out how well object detection models work. The mAP will then compare the ground-truth bounding box to the box that was found, and it will give a score based on that comparison. When the score is higher, it means that the model is able to find things more accurately.

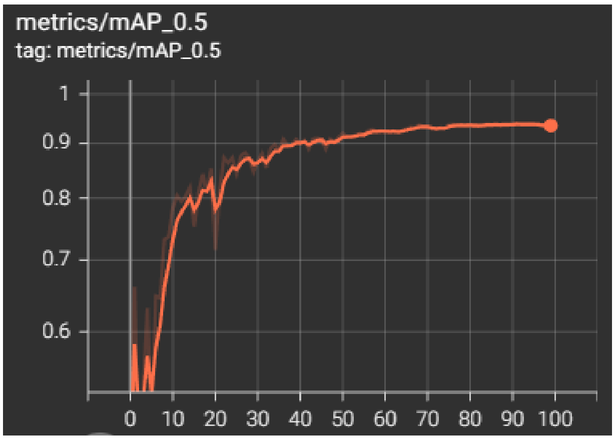

Figure 5.2: mAP_0.5 Accuracy of YOLOv5x Model

In Figure 5.2, we managed to get the accuracy up to nearly 94% in the YOLOv5x model. This model is successful 93.7% time in detecting the bounding box with the different types of waste in 100 epochs.

We have also reached the mAP_0.5:0.95 threshold with 70.5% accuracy. The accuracy graph of mAP_0.5:0.95 is illustrated in Figure 5.3.

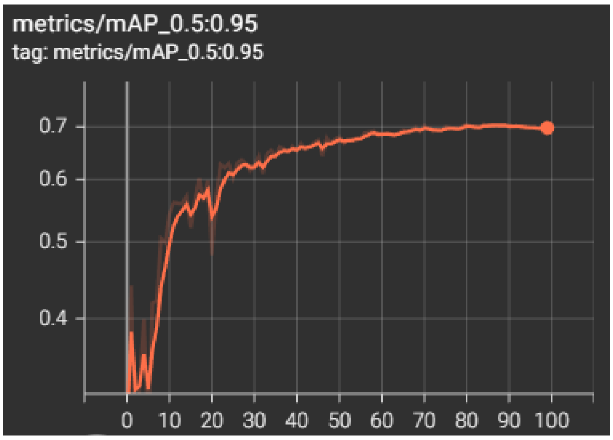

Figure 5.3: mAP_0.5:0.95 Accuracy of YOLOv5x Model

5.2.3 Precision

Precision is the ratio of accurately anticipated positive findings to the sum of all predicted positive findings. This demonstrates the capability of classification to identify positive values.

$$Precision = \frac{TP}{TP + FP} \tag{5.1}$$

In Figure 5.4, we observed that the YOLOv5x model has achieved 92.3% precision in terms of detecting waste.

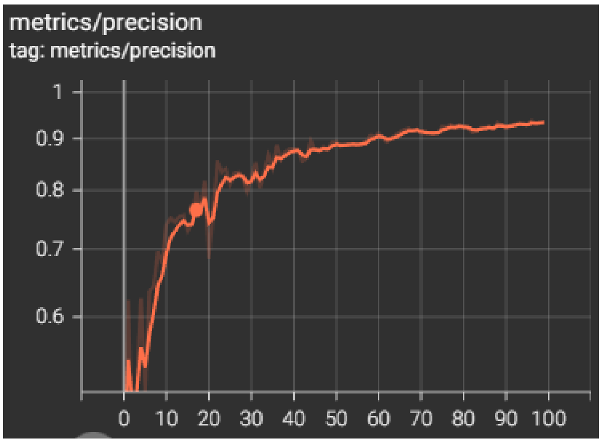

Figure 5.4: Precision of YOLOv5x model

5.2.4 Recall

The recall is computed by dividing the fraction of true positives by the total number of true positives. The recall is an evaluation of how effectively the model can identify positive samples. The greater the recall, the greater the number of positive samples discovered.

$$Recall = \frac{TP}{TP + FN} \tag{5.2}$$

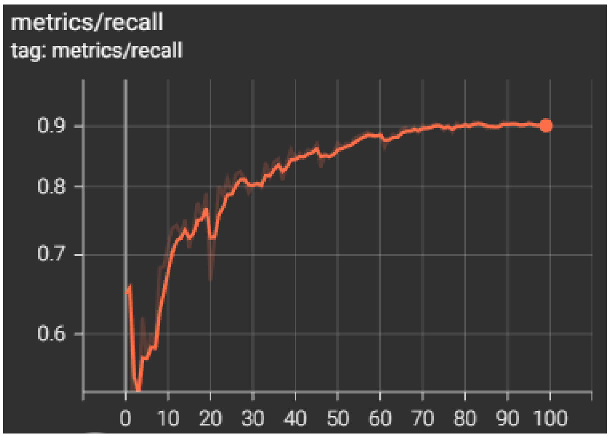

Figure 5.5: Recall of YOLOv5x model

In Figure 5.5, In our recyclable waste detection process YOLOv5x model has achieved 90.3% accuracy in finding the true positive samples.

5.2.5 Overall performance of the YOLO models

Our dataset was trained using the four different variations of the YOLOv5 model, which are most commonly referred to as YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x. We also used YOLOv7 to train our dataset.
We trained all the YOLOv5 variations with 100 epochs and the YOLOv7 model with 150 epochs. We also track how long it takes to train each variation of the YOLOv5 model and the YOLOv7 model. Our training YOLOv5s model took 1.341 hours. The other three variations - YOLOv5m, YOLOv5l, and YOLOv5x took 2.21 hours, 3.46 hours, and 6.23 hours respectively. The YOLOv7 model took 4.889 hours to complete 150 epochs. In contrast to the other versions of YOLOv5, the YOLOv7 model did not perform very well with our dataset.

Now we will observe the accuracy, precision, and recall of all the variants of the YOLOv5 model along with the YOLOv7 model.

Types of waste	YOLOv5s (mAP_0.5)	YOLOv5m (mAP_0.5)	YOLOv5l (mAP_0.5)	YOLOv5x (mAP_0.5)	YOLOv7 (mAP_0.5)
Cardboard	90.7%	91.4%	92%	92.6%	80.6%
Glass	95.4%	96%	95.5%	95.4%	85.6%
Paper	95.5%	96.5%	97.4%	96.9%	84%
Metal	95.6%	95.8%	95.6%	96.3%	89.9%
Plastic	87.1%	88.2%	87.6%	87.6%	74.9%
All	92.8%	93.6%	93.6%	93.7%	83%

Table 5.2: Comparison of the mAP_0.5 of the YOLO models

In the table 5.2, it is observed that the YOLOv5x variant of YOLOv5 model produces the highest mAP_0.5. All the other variants of YOLOv5 perform almost similarly

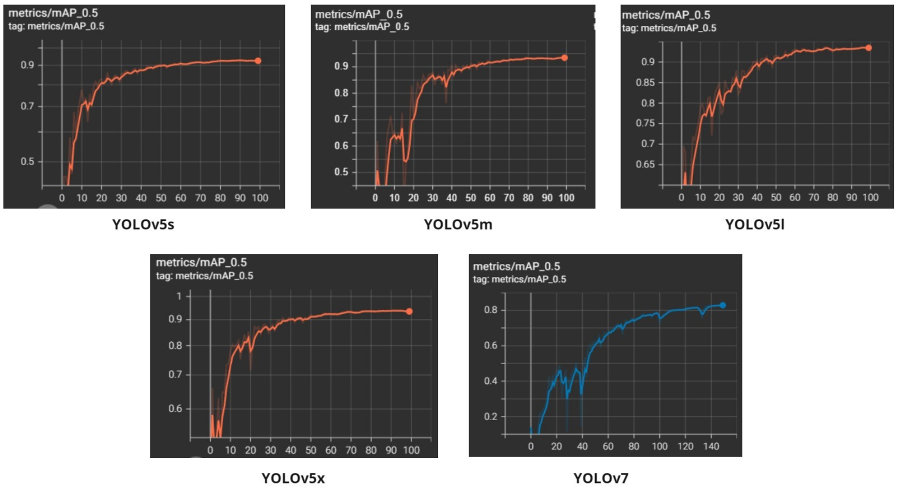

Figure 5.6: The mAP_0.5 graph of all the YOLO models.

but the YOLOv7 model obtained less mAP_0.5 among all the YOLO models here. In figure 5.6, we can see the mAP_0.5 graph of all the YOLO models.

Types of waste	YOLOv5s (precision)	YOLOv5m (precision)	YOLOv5l (precision)	YOLOv5x (precision)	YOLOv7 (precision)
Cardboard	92.4%	90.5%	94%	92.1%	81.2%
Glass	96.7%	95.6%	96.8%	94.6%	70.7%
Paper	89.3%	89.9%	92.4%	93.9%	72.3%
Metal	93.9%	93.3%	94.8%	94.5%	86.7%
Plastic	84.1%	86%	88.2%	86.5%	79.1%
All	91.3%	91.1%	93.2%	92.3%	78%

Table 5.3: Comparison of the precision of the YOLO models

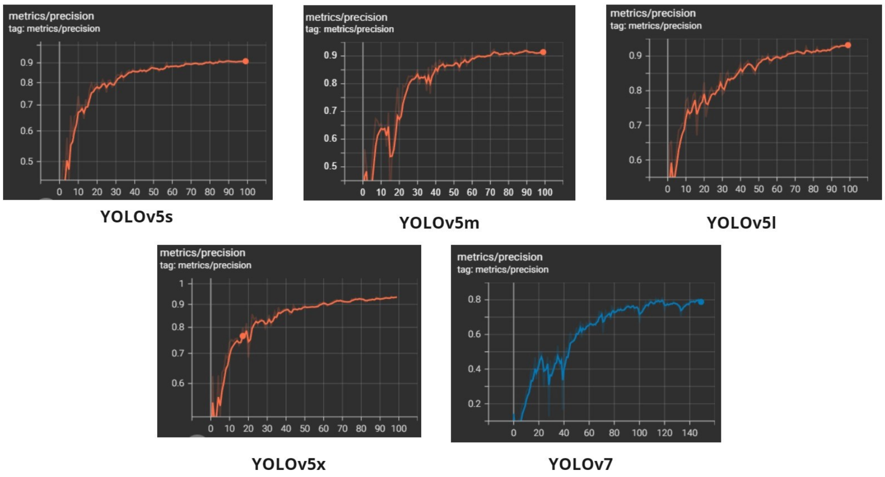

Figure 5.7: The precision graphs of all the YOLO models

In the table 5.3 it is observed that although the YOLOv5x model obtained the highest mAP_0.5 in terms of precision YOLOv5l obtained the highest precision.

The YOLOv5l model's 93.2% predictions were correct. All the other variants of the YOLOv5 model perform almost similarly and again YOLOv7 model performs less than all the other models. in figure 5.7, we can observe all the graphs of precision for all the YOLO models.

Types of waste	YOLOv5s (recall)	YOLOv5m (recall)	YOLOv5l (recall)	YOLOv5x (recall)	YOLOv7 (recall)
Cardboard	81%	85.5%	85.7%	84.7%	70%
Glass	90.7%	91.4%	92.8%	92.6%	84.9%
Paper	92.1%	93.6%	93.9%	94.1%	84.2%
Metal	93.9%	93.3%	94.8%	94.5%	86.7%
Plastic	84.4%	84.7%	86.2%	84.7%	63.4%
All	89.6%	90.1%	90.5%	90.3%	77.5%

Table 5.4: Comparison of the recall of the YOLO models

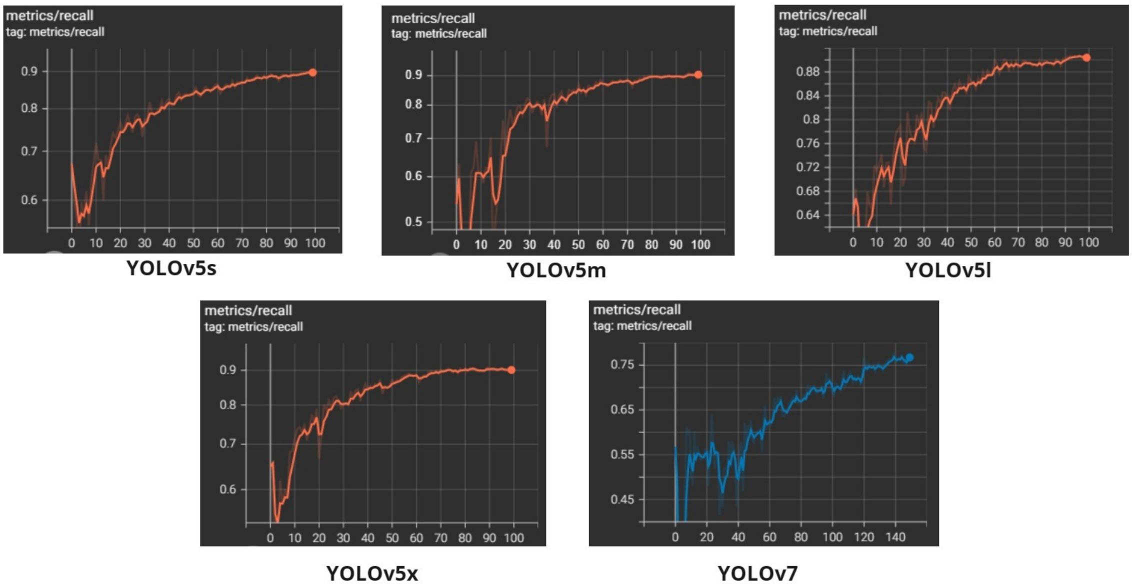

Figure 5.8: The recall graphs of all the YOLO models

In table 5.4, we can see that again the YOLOv5l obtained the highest recall which is 90.5%. The YOLOv5l model was successful 90.5% time in finding the true positive samples. In figure 5.8, we can observe all the graphs of recall for all the YOLO models.

5.2.6 Predicted wastes after training

After training here, we can observe our detected wastes from the images below. In figure 5.9, it can be observed that multiple types of waste are detected. In the first picture, one plastic and one metal are detected and in the second picture, two metals and one plastic are detected. These wastes are detected after the training of the YOLOv7 model. In figure 5.10, paper and plastic are detected from the video with the help of our webcam. In figure 5.11, we can see different types of wastes are detected after training the YOLOv5x model.

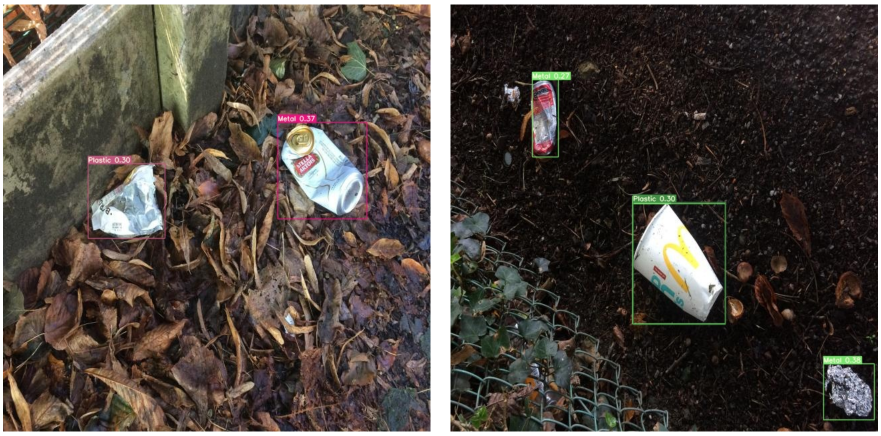

Figure 5.9: Predicted wastes after training the YOLOv7 model

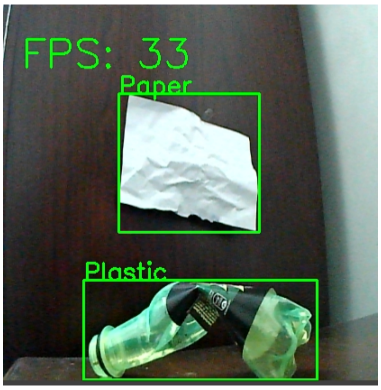

Figure 5.10: Real-time waste detection from video with webcam

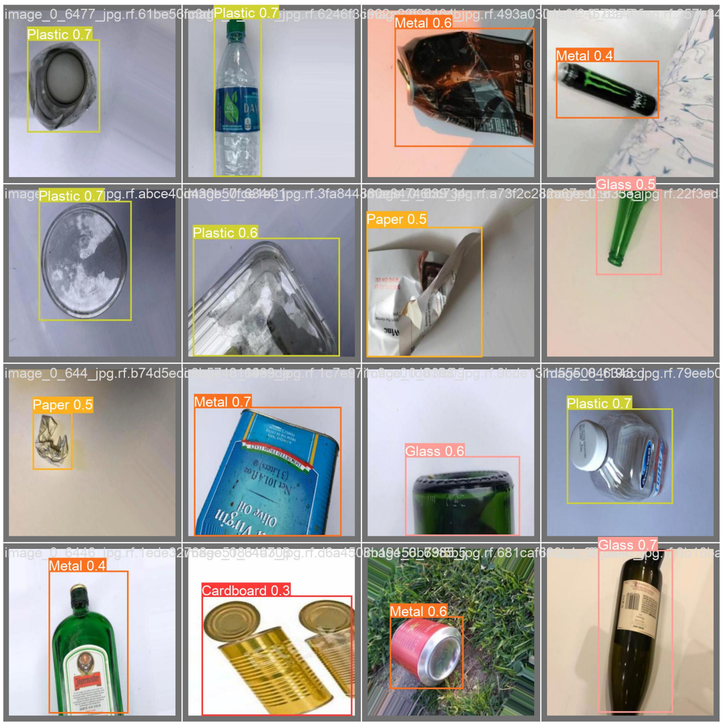

Figure 5.11: Predicted wastes after training YOLOv5x model

5.2.7 Image classification architecture Evaluations

In the YOLO models, we also used videos for prediction along with the images from the test set. There might be some places where the video is not allowed. There we can only detect wastes from images. This can also be done by the image classifier algorithms also. Object identification and image classification are two tasks in computer vision that are closely related to one another.

For comparison with the YOLO models, we have used popular image classifier algorithms such as VGG16, MobileNet, and Inception-ResNet-v2 on our dataset.

5.2.8 Image classifier algorithms Result analysis

We evaluated the models for performance measurement after completing the data pre-processing In the preliminary stage. For all these algorithms, we performed our training with 80% data and the rest of the data is used for the test set. When compared to the VGG16 and the Inception-ResNet-v2 architecture, we discovered that the MobileNet architecture attained the highest level of accuracy.

MobileNet architecture Performance

In order to train the MobileNet architecture, we operated for a total of fifty epochs with the dataset containing 8250 waste images.
After training the MobileNet architecture for 30 iterations, the highest accuracy we were able to achieve of 93%. The accuracy was 82% in the first epoch, and it ended at 91% in the 30th epoch. In figure 5.12, we can see the train and test accuracy of the MobileNet algorithm.

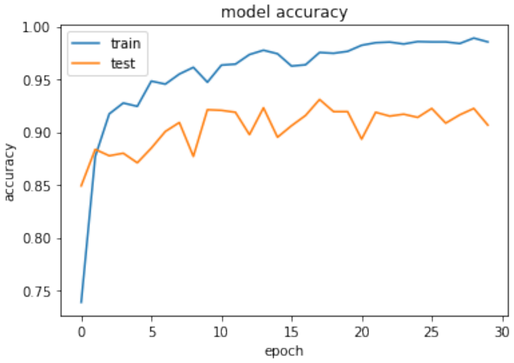

Figure 5.12: Train and Test the accuracy of the MobileNet algorithm

Performance analysis of the Inception-ResNet-v2 architecture

We also trained the model with the same dataset containing 8250 images. The accuracy of the test was 82% during the first epoch. The model's highest level of accuracy was determined to be 92.25%. Although the accuracy is not so bad in figure 5.13 we can observe that the accuracy is not stable like the YOLO algorithms. The train and test accuracy can be observed from the figure 5.13.

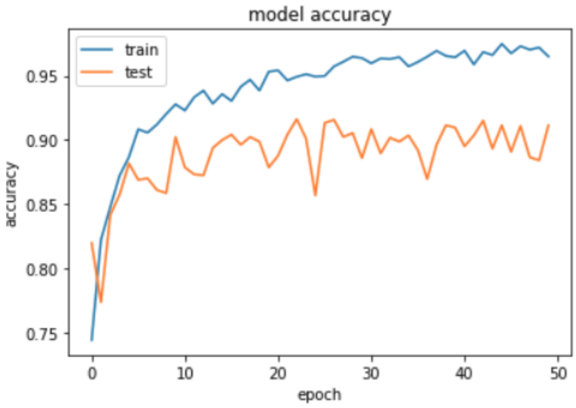

Figure 5.13: Train and test the accuracy of the Inception-ResNet-v2 algorithm.

Performance analysis of the VGG16 architecture

At first, we ran 10 epochs on VGG16 architecture but the accuracy result was not very good that's why decided to run 50 epochs on VGG16 The accuracy result was less than from the other two models. We achieved the highest accuracy of 88.12% in 50 epochs. The train and test accuracy curve is illustrated in figure 5.14.

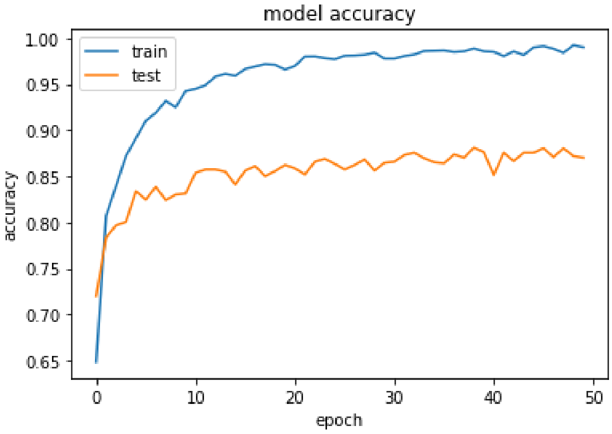

Figure 5.14: Train and Test the accuracy of the VGG16 algorithm.

5.2.9 Overall Performance analysis of all the models

We discussed the real-time object detection algorithms and the popular image classification algorithms so far. Now we will evaluate the overall performance based on accuracy, precision, and recall.

If we look at table 5.5, we can see that all the versions of the YOLOv5 model are performing quite well in terms of accuracy, precision, and recall. The YOLOv5x model came out on top as having the most accuracy out of all the models. So far, this is the most successful model that we've created. The YOLOv7 model couldn't perform well like the other YOLO models. The YOLOv7 is very advanced architecture. We think our dataset doesn't suit this model. If we could have provided more high-quality waste images we believe the YOLOv7 model also could have performed well.

Models	Accuracy	Precision	Recall
YOLOv5s	92.8%	91.3%	88.4%
YOLOv5m	93.6%	91.1%	90.1%
YOLOv5l	93.6%	93.2%	90.5%
YOLOv5x	93.7%	92.3%	90.3%
YOLOv7	83%	78%	77.5%
Inception-ResNet-v2	92.25%	91.4%	91%
MobileNet	93%	91%	90.6%
VGG16	88.12%	88.4%	87%

Table 5.5: Comparison of the performance of all models

The image classifier algorithms also perform pretty well, with Inception-ResNet-v2 and MobileNet doing somewhat better than the VGG16 model which is another image classifier algorithm we have used.

Chapter 6

Conclusion and Future Work

6.1 Future Scope

In the future, we will make an effort to include a greater number of images in our dataset. We will try to add high-quality images for better performance. In this research, we observed that plastics and cardboard performed less than the other types of waste. If we add more varieties of cardboard and plastic we hope to get e better accuracy in overall results. We have noticed that the color of the background can cause it to be misinterpreted as being cardboard sometimes. We will work on it for the perfect separation of the waste from the background. We can increase the dataset's diversity by including photographs from various environments, lighting situations, and camera angles. This would help to increase the model's stability and make it more applicable to real-world settings. In the future, this model can also be integrated into the robotic industry related to recycling for plants.

6.2 Conclusion

To summarize, recycling is an essential component of waste management since it contributes to the preservation of natural resources, the reduction of pollution, and the preservation of energy. It is necessary for individuals as well as communities to recycle and sort their waste in an appropriate manner. This will ensure that the waste is processed appropriately and provides the greatest possible benefit. In this research, we tried our best to detect recyclable waste. This work can be further extended by fine-tuning the model with more data, testing it in more complex scenarios, and exploring different architectures.

Bibliography

[1] L. A. Guerrero, G. Maas, and W. Hogland, "Solid waste management challenges for cities in developing countries," *Waste Management*, vol. 33, no. 1, pp. 220–232, 2013, PMID: 23098815. DOI: 10.1016/j.wasman.2012.09.008.

[2] D. M. C. Chen, B. L. Bodirsky, T. Krueger, A. Mishra, and A. Popp, *The world's growing municipal solid waste: Trends and impacts*, 2020. DOI: 10.1088/1748-9326/ab8659.

[3] *11-facts-about-recycling*, 2015. [Online]. Available: https://www.dosomething.org/us/facts/11-facts-about-recycling.

[4] F. Koop, *Why is recycling so important? the dirty truth behind our trash*, Jan. 2021. [Online]. Available: https://www.zmescience.com/other/feature-post/why-is-recycling-so-important-the-dirty-truth-behind-our-trash/.

[5] C. Capel, *Waste sorting - a look at the separation and sorting techniques in today's european market*, Jul. 2008. [Online]. Available: https://waste-management-world.com/artikel/waste-sorting-a-look-at-the-separation-and-sorting-techniques-in-today-rsquo-s-european-market/.

[6] D. G. Solla, *Advanced waste classification with machine learning*, Jan. 2022. [Online]. Available: https://towardsdatascience.com/advanced-waste-classification-with-machine-learning-6445bff1304f.

[7] T. World and B. Washington, *Decision makers' guide to municipal solid waste incineration.* [Online]. Available: https://www.biologyonline.com/wp-content/uploads/attachments/DecisionMakers.pdf.

[8] *Waste: A problem or a resource?* 2014. [Online]. Available: https://www.eea.europa.eu/publications/signals-2014/articles/waste-a-problem-or-a-resource.

[9] *Why is recycling important? 10 benefits of recycling*, 2021. [Online]. Available: https://northhillbottledepot.ca/why-is-recycling-important-10-benefits-of-recycling/.

[10] S. Ann, *8 reasons why we should recycle*, Jul. 2019. [Online]. Available: https://www.paulsrubbish.com.au/8-reasons-why-we-should-recycle/.

[11] A. Vogiatzis, G. Chalkiadakis, K. Moirogiorgou, G. Livanos, and M. Papadogiorgaki, *Dual-branch cnn for the identification of recyclable materials*, 2021. DOI: 10.1109/IST50367.2021.9651347.

[12] B. W. House, D. W. Capson, and D. C. Schuurman, *Towards real-time sorting of recyclable goods using support vector machines*, 2011. DOI: 10.1109/ISSST.2011.5936845.

[13] S. Thokrairak, K. Thibuy, and P. Jitngernmadan, *Valuable waste classification modeling based on ssd-mobilenet*, 2020. DOI: 10.1109/InCIT50588.2020.9310928.

[14] C. Bircanoğlu, M. Atay, F. Beşer, Ö. Genç, and M. A. Kızrak, *Recyclenet: Intelligent waste sorting using deep neural networks*, 2018. DOI: 10.1109/INISTA.2018.8466276.

[15] S. Niu, J. Wang, Y. Liu, and H. Song, "Transfer learning based data-efficient machine learning enabled classification," in *2020 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*, IEEE, 2020, pp. 620–626.

[16] H. Wang, "Garbage recognition and classification system based on convolutional neural network vgg16," in *2020 3rd International Conference on Advanced Electronic Materials, Computers and Software Engineering (AEMCSE)*, IEEE, 2020, pp. 252–255.

[17] N. Ramsurrun, G. Suddul, S. Armoogum, and R. Foogooa, "Recyclable waste classification using computer vision and deep learning," in *2021 Zooming Innovation in Consumer Technologies Conference (ZINC)*, 2021, pp. 11–15. DOI: 10.1109/ZINC52049.2021.9499291.

[18] A. Vogiatzis, R. A. Aral, Ş. R. Keskin, M. Kaya, and M. Hacıömeroğlu, "Classification of trashnet dataset based on deep learning models," *2018 IEEE International Conference on Big Data (Big Data)*, pp. 2058–2062, 2018. DOI: 10.1109/BigData.2018.8622212.

[19] B. Gan and C. Zhang, "Research on the algorithm of urban waste classification and recycling based on deep learning technology," in *2020 International Conference on Computer Vision, Image and Deep Learning (CVIDL)*, IEEE, 2020, pp. 232–236.

[20] D. Ziouzios, N. Baras, V. Balafas, M. Dasygenis, and A. Stimoniaris, "Intelligent and real-time detection and classification algorithm for recycled materials using convolutional neural networks," *Recycling*, vol. 7, no. 1, p. 9, 2022.

[21] J. Kim, O. Nocentini, M. Scafuro, *et al.*, "An innovative automated robotic system based on deep learning approach for recycling objects.," in *ICINCO (2)*, 2019, pp. 613–622.

[22] G. E. Sakr, M. Mokbel, A. Darwich, M. N. Khneisser, and A. Hadi, "Comparing deep learning and support vector machines for autonomous waste sorting," in *2016 IEEE international multidisciplinary conference on engineering technology (IMCET)*, IEEE, 2016, pp. 207–212.

[23] K. Sreelakshmi, S. Akarsh, R. Vinayakumar, and K. Soman, "Capsule neural networks and visualization for segregation of plastic and non-plastic wastes," in *2019 5th international conference on advanced computing & communication systems (ICACCS)*, IEEE, 2019, pp. 631–636.

[24] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, *et al.*, *Review of deep learning: Concepts, cnn architectures, challenges, applications, future directions*, 2021.

[25] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, *Proceedings of the ieee conference on computer vision and pattern recognition*, 2016. [Online]. Available: https://arxiv.org/pdf/1506.02640v5.pdf.

[26] E. Zvornicanin, *What is yolo algorithm?* 2022. [Online]. Available: https://www.baeldung.com/cs/yolo-algorithm.

[27] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *Institute of Information Science Academia Sinica*, Apr. 2020.

[28] C.-Y. Wang, A. Bochkovskiy, and H.-y. Liao, "Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," Jul. 2022. DOI: 10.48550/arXiv.2207.02696.

[29] S. Tammina, "Transfer learning using vgg-16 with deep convolutional neural network for classifying images," *International Journal of Scientific and Research Publications (IJSRP)*, vol. 9, p9420, Oct. 2019. DOI: 10.29322/IJSRP.9.10.2019.p9420.

[30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016. DOI: 10.1109/CVPR.2016.90.

[31] M. Mahdianpari, B. Salehi, M. Rezaee, F. Mohammadimanesh, and Y. Zhang, "Very deep convolutional neural networks for complex land cover mapping using multispectral remote sensing imagery," *Remote Sensing*, vol. 10, p. 1119, Jul. 2018. DOI: 10.3390/rs10071119.

[32] A. Pujara, *Image classification with mobilenet*, Jul. 2020. [Online]. Available: https://medium.com/analytics-vidhya/image-classification-with-mobilenet-cc6fbb2cd470.

[33] M. Yang and G. Thung, *Classification of trash for recyclability status*. [Online]. Available: https://drive.google.com/drive/folders/0B3P9oO5A3RvSUW9qTG11Ul83TEE?resourcekey=0-F-D8v2tnSfByG6ll3t9JxA.

[34] S. SEKAR, *Waste classification data*. [Online]. Available: https://www.kaggle.com/datasets/techsash/waste-classification-data.

[35] P. Simões, *Taco-trash-datase*. [Online]. Available: https://github.com/pedropro/TACO.

[36] S. Lau, "Image augmentation for deep learning," Jun. 2017. [Online]. Available: https://towardsdatascience.com/image-augmentation-for-deep-learning-histogram-equalization-a71387f609b2.

[37] P. Soni, "Data augmentation: Techniques, benefits and applications," Jan. 2022. [Online]. Available: https://www.analyticssteps.com/blogs/data-augmentation-techniques-benefits-and-applications.