OBJECT DETECTOR FOR WASTE DETECTION BY MODIFYING FEATURE PYRAMID NETWORKS TO ENHANCE FEATURE FUSION

by

Ocean Monjur, 180041112

Mohammad Galib Shams, 180041130

Faysal Mahmud, 180041117

Supervised By

Dr. Md. Hasanul Kabir, PhD

Professor, Dept. of CSE

Md. Bakhtiar Hasan

Assistant Professor, Dept of CSE

Ahnaf Munir

Assistant Professor, Dept of CSE

A thesis report submitted to the Department of CSE
in partial fulfillment of the requirements for the degree of
Bachelor of Science in CSE

Department of Computer Science and Engineering

Islamic University of Technology
Organization of the Islamic Cooperation (OIC)

Dhaka, Bangladesh

May 19, 2023

Candidate's Declaration

This is to certify that the work presented in this thesis entitled, "Object Detector for Waste Detection by modifying Feature Pyramid Networks to enhance feature fusion", is the outcome of the research carried out by Ocean Monjur, Mohammad Galib Shams, Faysal Mahmud -under the supervision of Dr. Md. Hasanul Kabir, PhD, Professor, Md. Bakhtiar Hasan, Assistant Professor and Ahnaf Munir, Assistant Professor, Dept of Computer Science and Engineering (CSE), Islamic University of Technology(IUT).

It is also declared that neither this thesis nor any part thereof has been submitted anywhere else for the award of any degree, diploma, or other qualifications.

Signature of the Candidate

Ocean Monjur, 180041112

Mohammad Galib Shams, 180041130

Faysal Mahmud, 180041117

Supervised By

Dr. Md. Hasanul Kabir
Professor
Department of Computer Science and Engineering
Islamic University of Technology

Md. Bakhtiar Hasan
Assistant Professor
Department of Computer Science and Engineering
Islamic University of Technology

Ahnah Munir
Assistant Professor
Department of Computer Science and Engineering
Islamic University of Technology

Acknowledgements

We would like to express our gratitude to Professor **Dr. Md. Hasanul Kabir, PhD**, Assistant Professors **Md Bakhtiar Hasan** and **Ahnaf Munir** for being our advisors and mentor. Their insight and guidance significantly shaped our research, providing us with proper direction every step of the way. Their meticulous and thorough process of work helped us detect and correct any inaccuracies in our research early and effectively. They provided constructive criticism for every draft we presented, which greatly helped improve the quality of our research. We are grateful for their patience with our work, it was an essential component in inspiring us to delve deep into the subject matter and keep continuing with our endeavor when faced with challenges in our work.

We would also like to extend our heartfelt gratitude to all the members of the Computer Vision Lab at the Islamic University of Technology who have guided us throughout the endeavor. It is because of their support and commitment which enabled us to conduct this research successfully.

Ocean Monjur thanks his parents Naznin Akter, Md Monjurul Alam, and his Sister Naomi Alam for everything and dedicates this work to them.
Mohammad Galib Shams thanks his parents Hasina Begum and A.Q.M Golam Rabbani for everything and dedicates this work to them.
Faysal Mahmud thanks his parents Md. Shaiful Islam and Fatema Akter for everything and dedicates this work to them.

Contents

List of Figures

List of Tables

Abstract

Waste Detection is of significant importance to the environment. With the lack of a proper detection architecture for waste detection, we are looking to propose an Object Detector specifically designed to deal with the varying attributes required for accurate waste detection to increase the general detection capabilities of the detector. Our approach involves modifying the neck of the object detector to increase performance due to the neck having more influence over the total performance of the object detector than any other component. Building on this hypothesis, we propose our modified recursive feature pyramid neck, called BRPN,Balanced recursive pyramid network. The BRPN involves merging the already established recursive pyramid network with the balanced feature pyramid network. By re-scaling, integrating, refining, and strengthening qualities of the balanced feature pyramid network integrated with the already high-performing recursive feature pyramid, our balanced recursive pyramid network increases the average precision metrics on Zerowaste and Flow data sets by +1.24 and +.91 when compared against the vanilla recursive pyramid network.

Chapter 1

Introduction

1.1 Motivation and Scope

1.1.1 Object Detection

Object detection in computer vision deals with detecting instances of objects of certain classes like cars or bottles in images [1]. The primary objective of this domain is to provide an estimation of the location of the specified objects. Object detection has applications in various real-world situations like autonomous driving, video surveillance, or in waste detection. Typically images consist of a small number of objects. But there are many locations where these objects may be positioned and their scales may vary too [2]. Therein lies the challenge of object detection. Applications of object detection range in various domains; from autonomous driving and video surveillance to waste detection. Our research looks into improving the performance of object detectors, particularly in the waste detection domain, enabling faster and more accurate detection

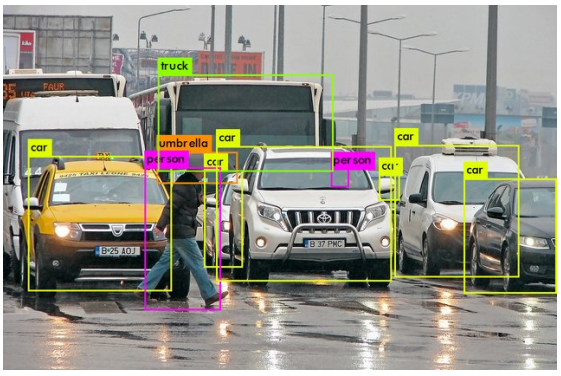

Figure 1.1: Object Detection Example

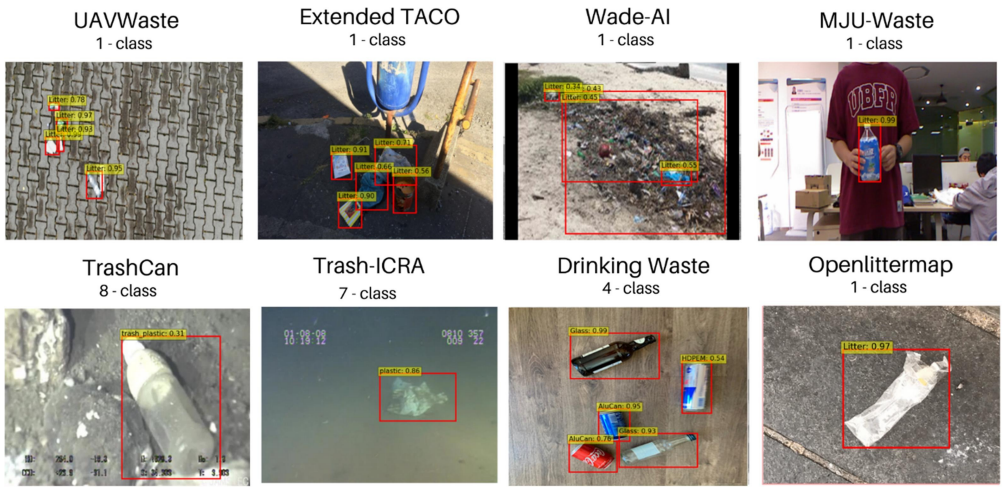

Figure 1.2: Waste Detection Example

of waste objects in different environments. Object detection has gone through a lot of improvement with the development of deep learning techniques in recent years, with it being responsible for a lot of breakthroughs in this domain [3]. Conventionally, Object Detectors involve the backbone module which is responsible for extracting deep latent features, and the neck module, which in turn fuses those features to attain information at different scales [4]. Our research mainly focuses on the neck module and how we can modify some parts of it to increase feature fusion.
Object detection has gone through a lot of improvement with the development of deep learning techniques in recent years, with it being responsible for a lot of breakthroughs in the domain of Object Detection [3].

1.1.2 Waste Detection

According to the world bank, we are expected to reach about 3 billion tonnes of waste by the year 2050 [5, 6]. Analyzing this waste can help in managing it. This is a difficult task to achieve manually. Because of the wide variety of waste objects present, automation in this field is difficult to achieve making waste management a highly labor-intensive task. [7] Improper waste management has several consequences including severe effects on the natural resources, public health, and the overall environment. Here waste detection can not only help with managing waste using automation but can also be used to educate users about throwaway waste [8]. Deep learning can be used to classify and detect waste, which can help with analysis, streamlining the process as a result. Different types of waste require different methods of treatment. Some types of waste such as paper, cardboard, glass, or plastic can be recycled [9, 10]. Some other types like lumber or boxes can be reused or re-manufactured [8, 10]. It is

important to understand these different management techniques, as mismanagement of waste can lead to causing more problems than providing a solution.

Hence detection of waste can be really useful in determining methods of treatment for different kinds of waste. Waste detection can also give useful information about the environment such as the varieties of waste in different locations and their overall effect on the environment. The presence of toxic substances like medical wastes can be identified and measured using deep learning, providing opportunities to assess the situation and make decisions to minimize damage to the ecosystem. Waste detection techniques can be applied to identify waste in locations where it is difficult to reach. In these locations, waste detection techniques can be used by automated systems [11] to collect trash at a much faster rate than if it was done manually. Several waste data sets have been organized and collected in recent years and these data sets provide opportunities for detecting waste in challenging environments. There is also research work on merging some of the data sets to create new and bigger data sets which helps with improving the quality of these data sets.

Convolutional Neural Network algorithms have become the standard for most computer vision tasks including object detection [11]. In our research, we focus on the modification of existing CNN architectures to ensure generalized object detection performance is improved. This is because, in the waste management domain, the types of objects that need to be detected are also generalized.

1.2 Problem Statement

While recent works have introduced various kinds of data sets, due to the environment from which these data sets are formulated such as underwater trash, cluttered trash, or water surface trash, it presents complex challenges to overcome. These challenges include the visibility of waste in these environments due to their size, shape, or the environment itself. For example, underwater trash is difficult to detect due to the background affecting visibility. Some kinds of trash which are small in size are even more difficult to detect in challenging environments like bottles floating in inland waters. All these factors affect the accuracy of the existing models trained by these data sets. Hence a generalized model is required to improve the detection of these wastes in challenging conditions accurately and efficiently. Because of the challenges present in the existing waste datasets, it is important that the focus should be on all levels of feature extraction, from low-level features to high-level features. This will ensure that varieties of object types can be handled efficiently by the same object detector.

Overall it is important that the generalized system we construct will be robust for different types of environments, sizes, and orientations of objects in the images.

The problem statement can be stated as “Modelling a generalized system for waste detection that is robust for various situations which include different environments and different types of waste and efficient enough to be used on an industrial scale.”

1.3 Research Challenge

The first challenge that needs to be dealt with is finding an architecture that performs well for different waste data sets. For different data sets that were published recently, different deep learning architectures were used to train the data, however, generalized architectures have not been established which perform well for different kinds of waste data sets and or merged data sets.

The next challenge would be to ensure the good performance of the architectures while balancing the computation cost. This will be the most difficult challenge considering that waste detection datasets have different challenges like difficult conditions for the background of the images or variation of size, shape, and orientation of objects in images. The performance of recent works using these architectures suggests there can be a lot of things which is possible to improve these architectures.

Another challenge would be to ensure that modification to existing architecture is working as an addition to the existing architecture. This means ensuring that modification does not eliminate any main feature of the existing architecture and that either the modification improves upon those features or adds to the set of features. This ensures that the strength of the architecture is increased in an incremental fashion which will be easier to understand and improve upon.

It is better if any modification on a single architecture can be replicated for any other feature with similar architecture. This means modification needs to be modular. If we modify the structure of the Feature Pyramid Networks of an object detector, then modifications should be such that it can be added to any object detector with a feature pyramid network or a version of the feature pyramid network. This ensures for any future architectures introduced with greater performance, further experimentation can be performed with the modifications introduced in our research.

Other challenges include setting the hyperparameters. This involves finding the best set of parameters and ensuring that they are kept the same for a fair comparison of the performance of the models.

The final Challenge includes training and comparison of results. This heavily depends on the resources available for our research, and the extent of our modifications. It is vital that there is a balance between the increase in performance the model introduces and how computationally expensive the operations become after the modification is introduced. It is also important to ensure that our research takes place within the constraints set in terms of our resources, hence enabling fair comparisons between different architectures.

1.4 Objectives of the Thesis

1. Build an efficient waste detection method that is robust for various backgrounds such as in underwater environments or cluttered environments. This system can be used for automated waste detection with low inference time.
2. The waste detection method should also work well for detecting waste of different sizes and orientations. For example, it should be able to detect small objects with reasonable accuracy compared to objects of bigger sizes.
3. Targeting a specific area of object detector to generalize the improvement of detection. A particular region of interest can be the neck of the object detector which is responsible for the fusion of semantic and spatial features.
4. Comparing the performance of the proposed architecture with appropriate modifications with the architecture without the modifications
5. Analyzing the performance of our proposed method and comparing the results with the existing architectures present in recent works.

1.5 Research Contribution

To summarize the key contributions so far

1. We modified the Recursive Pyramid Network of the neck of DetectoRS by adding the balanced feature pyramid to strengthen multi-level features of the Recursive Pyramid Network
2. The proposed architecture outperforms the original DetectoRs performance on ZeroWaste and Flow

3. All other conditions including the data augmentations were kept same for the experiment.

1.6 Organization

We continue our thesis report from Chapter 2. In chapter 2 we provide the literature review of various dataset papers in the waste detection domain. This is followed by the literature review of the different architectures regarding the neck section of the FPN.
In chapter 3 we discuss the architectures we proposed or worked on so far. In this chapter, we go over in detail about the architecture pipeline providing various diagrams and explanations of the proposed architectures.
In Chapter 4 we go over the results and discussions of our work so far. In this chapter, we analyze the results of our experiments and provide conclusions based on these results.
In chapter 5 we provide the topics we are going to pursue before the final thesis defense based on our recent research and experiments. Lastly, we provide the overall conclusion.

Chapter 2

Background Study

2.1 Waste Detection

[12–16] Waste can be indoor waste or outdoor waste. Small, big, underwater, on land, solid, liquid [12–16]. As each category of waste shows different textures, settings, and sizes the work on waste detection is predominantly focused on creating and maintaining data sets. The paper we have studied created new data sets or merged data sets from existing waste-based data sets. But one common feature in all of the papers was, their proposed methodology is based on deep learning architecture.
Sylwia et al [5] presented a new benchmark data set that utilizes the existing open-source data set to the fullest. The publicly available data sets were merged and filtered to create a unified benchmark data set. As waste identification is an ambiguous process all the existing data sets focused on key criteria. Either the data set is focused on one single class or the data set image count is not enough to create a relevant deduction. Thus the problem arises of creating a benchmark waste detection data set, which will solve the above-mentioned problems and create a merged benchmark data set that will show the best practices of waste detection. Also, it will work as a baseline.
Then the methodology they followed for creating the merged data set, and detect waste is given. Detect waste is based on the Extended TACO and additional waste-based data sets were added thus making the whole data set of over 28000 images and 40000 objects. The key point in the detect waste data set was that it created a diverse waste-based data set featuring indoor, outdoor, and underwater images. Additionally, the authors proposed a two-stage deep learning-based framework for detecting waste in neutral conditions. The proposed framework outperforms the merged data set and also on the individual data set. All of this created a baseline for future waste detection work to be based on.

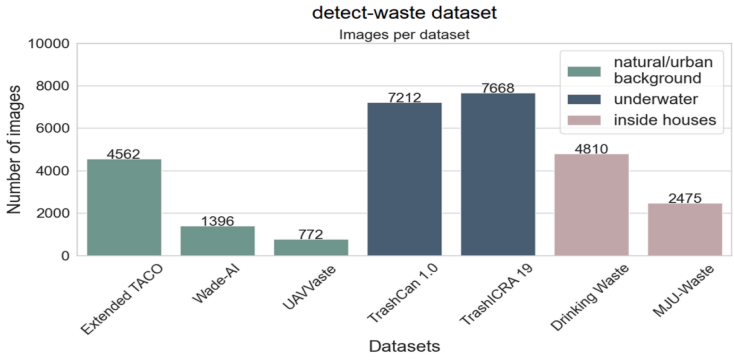

Figure 2.1: Detect waste distribution

Split	#Images	*Cardboard*	*Soft Plastic*	*Rigid Plastic*	*Metal*	#Objects	Domain
Train	3002	12940	4862	1160 +2778	263 +4373	19225 +7151	Real
Validation	572	2167	855	305 +637	60 +1010	3387 +1647	
Test	929	3428	1236	315 +886	63 +990	5042 +1876	
Unlabeled	6212	-	-	-	-	-	
Total	10715	18535	6953	1780 +4301	386 +6370	27744 +10584	
TACO	1499	240	652	1183	506	2581	Real
ReSortIT	16000	8000	8000	8000	8000	32000	Synthetic

Figure 2.2: Zerowaste dataset distribuition

Dina et al [12] presented the largest openly available in-the-wild waste detection dataset ZeroWaste focused on finding waste in cluttered scenes. As the industrial recycling sector processed waste in the MRF conveyor belt, the waste there is usually cluttered in a tiny space. In existing waste datasets the waste images are made of clear backgrounds and the wastes rarely overlap with one another thus creating a necessity for an industrial-grade cluttered waste dataset. Dina et al formulated the dataset to solve the above problem. They proposed a waste dataset that focused on deformable waste in dense areas that fully replicates the recycling facility.

The data necessary for the waste detection were collected from a recycling facility during its regular operational hour. Thus making the dataset similar to the real world. ZeroWaste-f the fully supervised part of the dataset consists of 4661 frames from the video that was recorded in the recycling facility. One limitation that arises from the dataset is as we can see in the figure, there is a class imbalance in it. So the way to fix it was to augment the dataset with the TACO dataset.

The authors also provided a baseline result for the ZeroWaste dataset. The three most popularly used object detection algorithms are used such as Mask R-CNN, Trident Net, and RetinaNet. Results from that experiment showed that TridentNet performed

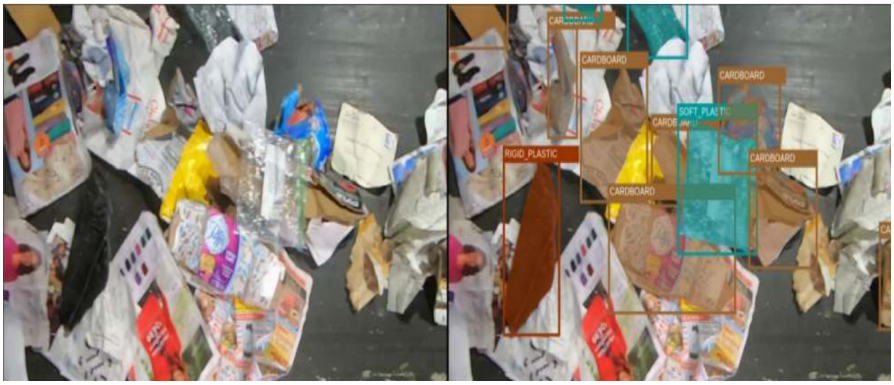

Figure 2.3: Zerowaste dataset

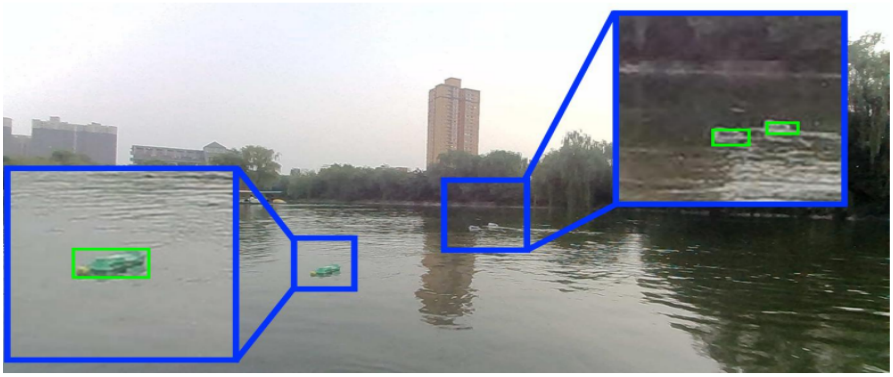

Figure 2.4: Flow dataset images

better out of the three. But one interesting finding from the paper was that all of the methods struggled to identify small objects. Another limitation of the paper that the author described is that the image count the is still smaller than those of the standard large-scale benchmarks. And lastly, the author described that the current state-of-the-art detection method still struggles in detecting cluttered waste.

Yuwei et al [13] presented a floating waste detection dataset FloW that was collected from the point of view of the USVs in real-world inland water in various conditions. The main focus for creating a flow dataset was to enable unmanned vehicles to venture into the marine inland water thus reducing the risk of human injuries and mitigating the pollution present in the inland waters. Flow is the first openly available dataset that is focused on inland waters thus making the waste detection task richer than before. To accurately represent the real-world condition of the waste. Yuwei et al used USV to take the images in the inland waters. Moreover, the images were rescaled and upscale for better object detection functionality. The final version of it, The FloW datasets contain 2000 images with 5271 labels. Then the author benchmarked the dataset with some of the well-used object detection algorithms to find how it dealt with the dataset. The key finding from the result was that two things hindered a better result. One was

light reflection and the other one was water waves. Both of these along with the fact that most of the objects in the waste dataset are small impacted the result. All of it together can make the deduction that accuracy for real-life floating bottles is relatively low.
Another important impact done by the authors Yuwei et al was that more than half the labeled floating waste in the dataset can be regarded as small objects. Thus it paves the path for a waste detection algorithm that focuses more on small waste objects.
Michael et al proposed an underwater waste data set that focuses on marine litter. Marine debris shows a great threat to the environment as it affects the marine ecosystem. Disrupting the natural way of the environment.
The main difficulty in collecting and recycling marine waste is that polluted locations are hazardous for humans to access. Thus creating the need for an autonomous waste detection system. With that view in mind, the authors created the TrashICRA data set. It was made possible by the use of J-EDI data set which consists of video footage underwater. From the video footage, it was sampled at 3 frames per second. After upsampling and correction, the final data set consists of 5720 images.
Afterward, the four most popular state-of-the-art object detection algorithms were used to benchmark the data set and to find out how object detection algorithms work in the case of underwater marine debris. Results from the data set don't give us a clear verdict. As underwater marine debris collection is done by autonomous underwater vehicles there is a need for a balance between inference time and accuracy. Thus concluding that the architecture that one wants to use is purely a subjective choice as all of the frameworks give us near-identical results [17]

2.2 Object Detector Necks

The neck of an object detector contributes the most to the performance of an object detector [4]. Since our approach focuses on improving the neck portion of an object detector in this section we focus on necks and show previous literature on object detector necks.

2.2.1 Feature Pyramid Network

Tsung Yi Lin et al proposed FPN [18], a top-down layer with lateral connection to make use of the pyramid structure of deep convolution architectures. The FPN mixes high semantic information available from deep levels of a convolution network and

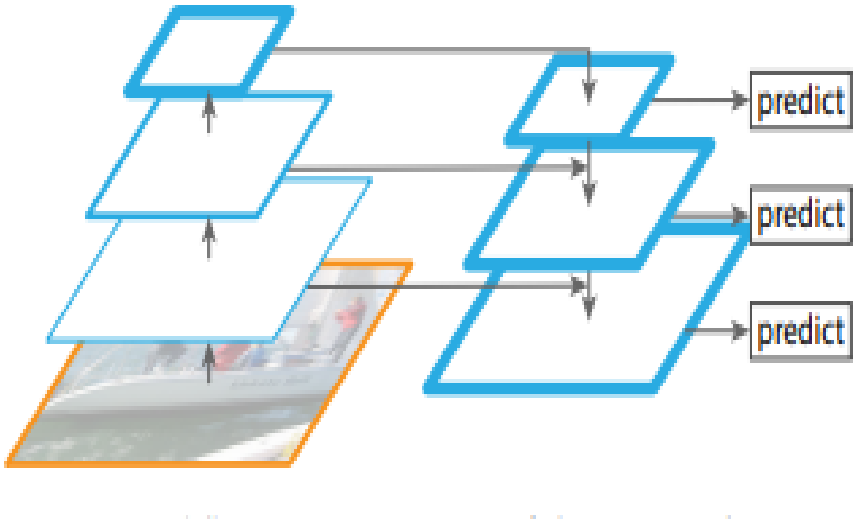

Figure 2.5: Feature Pyramid Network

fuses it with high spatial information available at the lower layers. The fusion of high spatial and semantic information massively increases the performance of object detectors. The primary concept of the Feature Pyramid network is to take the benefits of semantic information found in the processed feature maps and take the high-resolution feature map of earlier layers. The authors couple FPN with the Faster R-CNN architecture and saw SOTA performance on the COCO object detection dataset at the time of publication. The introduction of feature pyramids in deep learning massively influenced further model architectures. The FPN laid the template for a multitude of papers [19–23] to work on and improve upon.

2.2.2 Path Aggregation Network

Shu Liu et al improved upon FPN by introducing PaNet (Path Aggregation Network) [19]. PaNet picks up on a key weakness of the FPN design that the long path from the backbone bottom layer which contains more spatial information is lost. PaNet suggests an extra bottom-up layer after the FPN top-down layer to integrate more spatial information with the already rich semantics by cutting the distance down. The extra top-down layer adds extra computation which adds more semantic information too. Also, the PaNet paper argues that the FPN levels which separately predict from each FPN layer don't take into account that each layer might have information from different levels. Adaptive Feature Polling in the PaNet fixes that by taking a mix of all the Neck output layers. The authors showed results of PaNet on instance segmentation tasks, on the Mask R-CNN which outperformed FPN. Later works have extended PaNet too Object Detection as well

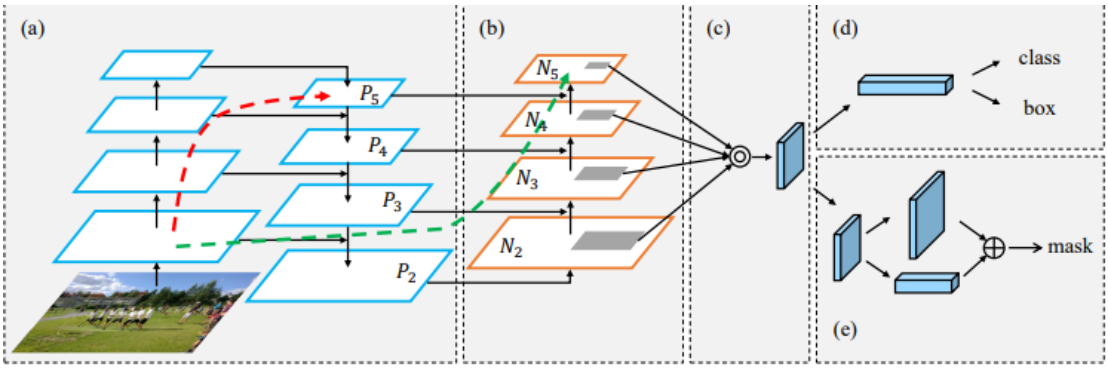

Figure 2.6: Path Aggregation Network

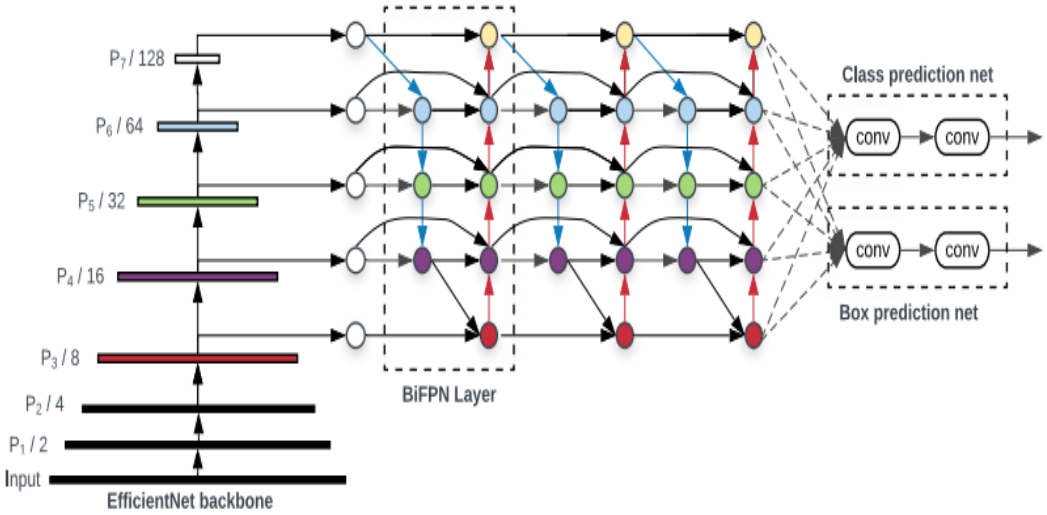

Figure 2.7: EfficientDet

2.2.3 Bidirectional Feature Pyramid Network

Ming Xin et al [20] proposed the EfficientDet object detector. The Detector uses BiFPN which comes as an improvement to PaNet. The authors argued that the layers in PaNet which only had a single input edge weren't as important to the performance of the model compared to the computation they added. So they removed the layers with only the input edge and due to the computational complexity being less could stack more of these BiFPN layers together to make a bigger higher performing neck. The authors also argued that each layer of a neck has a different importance to each task, so they assigned weights to each layer of the neck and let the model figure which layer to focus on for each dataset or task. BiFPN improves upon PaNet by a big margin.

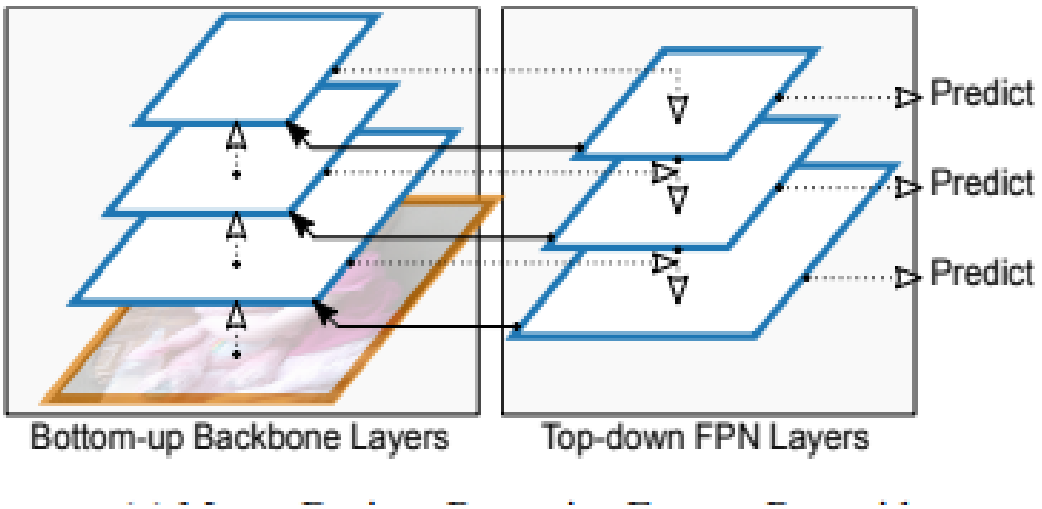

Figure 2.8: Recursive Feature Pyramid

2.2.4 Recursive Feature Pyramid Network

S Qiao et al [24] in their paper proposed RFP (Recursive Feature Pyramid) neck with their object detector Detectors. They took a different approach from their contemporary neck designs. They adopted the design philosophy of looking and thinking twice where a loop between the backbone and neck in this FPN was used. This loop enhances the FPN by making sure the backbone features receive direct gradient updates from the head of the detector. Using the RFP neck and incorporating atrous convolutions the authors show that their architecture achieves SOTA in the COCO object detection dataset.

2.2.5 Generalized Feature Pyramid Network

Yiqi et al [4] start with an intuition that the neck contributes more to the performance of an object detector than any other component. Based on that intuition they proposed to make an extremely heavy neck and compensate for that with an extremely light backbone architecture. The authors proposed a novel S2D chain (Space to depth) backbone and a neck GFPN (Generalized FPN). The GFPN is stacked back to back for further feature fusion. For effective feature transfer to the deep layers, a novel queen fusion is proposed by the author. They also used a log2n skip connection. The queen fusion takes 3 layers of a single FPN top-down and merges them by interpolating the top layer and max-pooling the bottom layer. The authors show that their detection model beats the SOTA map on the COCO dataset by using an inferior backbone to the ResNet but compensating with a heavy neck.

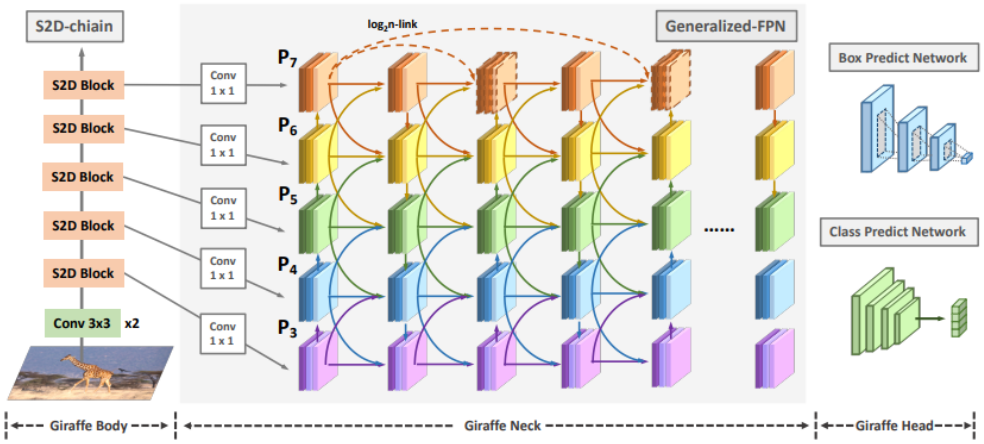

Figure 2.9: GiraffeDet

2.2.6 Parallel Feature Pyramid Networks

The authors try to improve upon the limitation of the original feature pyramid network [21]. The original feature pyramid's different abstraction levels reduce overall performance, especially on smaller objects. It loses a lot of information in this process. The parallel Feature pyramid addresses these lackings by not increasing the depth but increasing the width of the feature pyramid. In order to create a parallel pool of feature maps with various sizes, the authors first use spatial pyramid pooling along with a few extra feature transformations. The additional feature transformation in PFPNet is parallel processing is used to produce feature maps with comparable levels of semantic abstraction across scales. Finally to create the final Feature pyramid levels the feature pool pieces are resized to a uniform size and their contextual data is collected. The concept from this is that computing in parallel and not in-depth retains some spatial information that is required in identifying smaller objects. The authors use an MCSA module similar to that of the inception network to compute and process the feature maps in parallel. The report significant increase in performance when compared to the original feature pyramid network on both the Pascal VOC and COCO object detection data sets.

2.2.7 Extended Feature Pyramid Network

Deng et al [23] propose an Extended Feature Pyramid Network, which includes an additional new module feature texture transfer (FTT), which super-resolves features and extracts specific regional information of the passed on feature map. The motivation for this paper comes from the fact that previous feature pyramids' efforts to detect

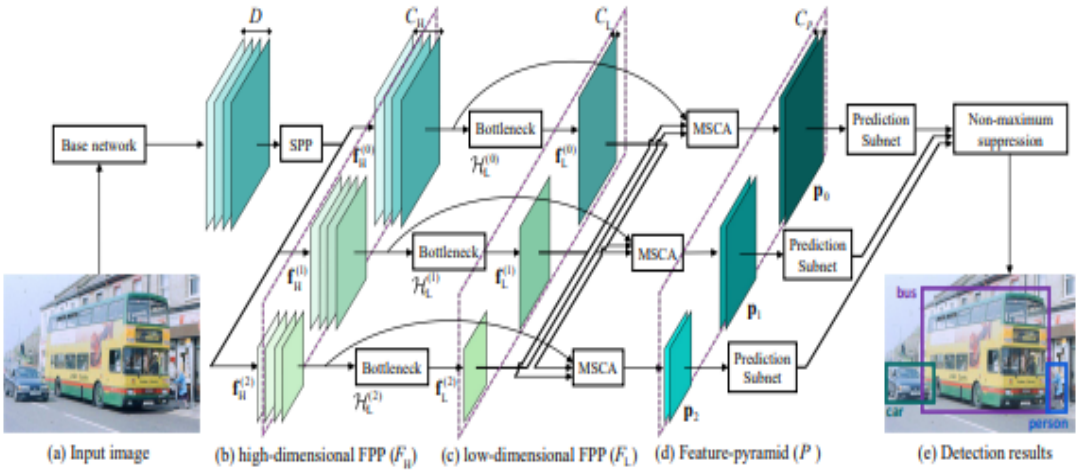

Figure 2.10: Parallel Feature Pyramid Network

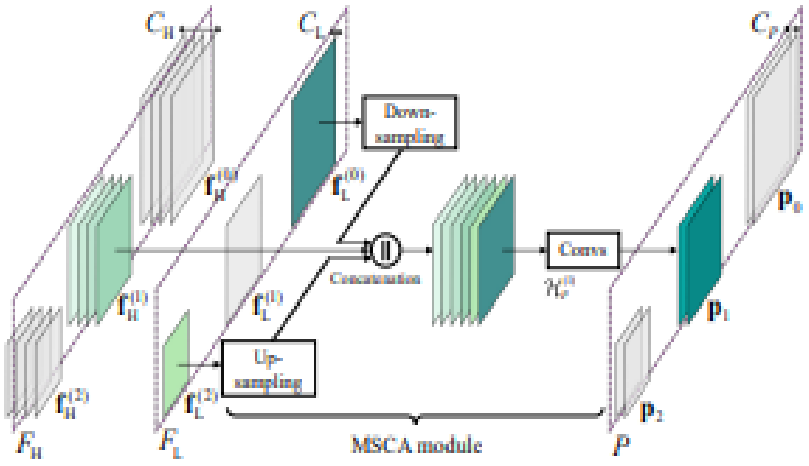

Figure 2.11: MCSA Module

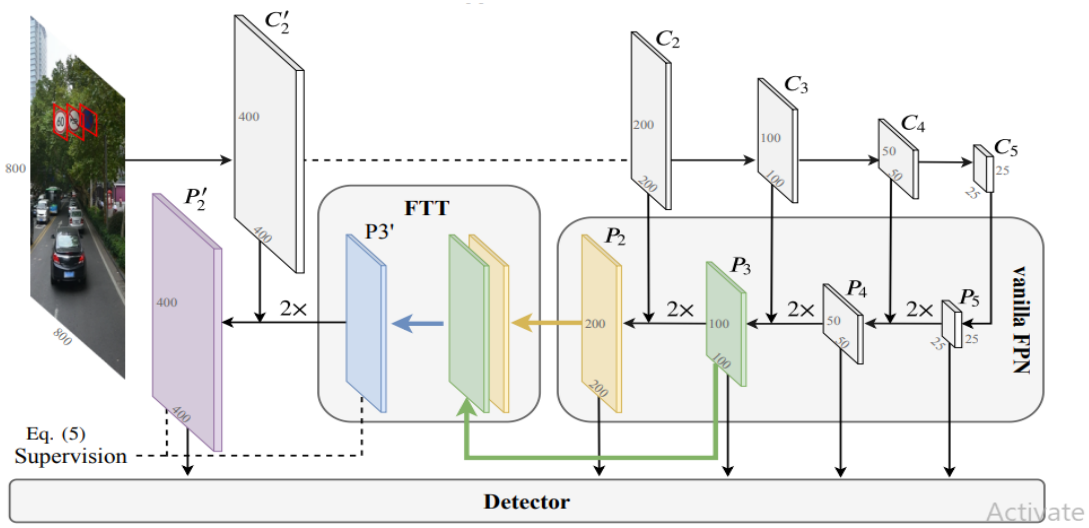

Figure 2.12: Extended Feature Pyramid Network

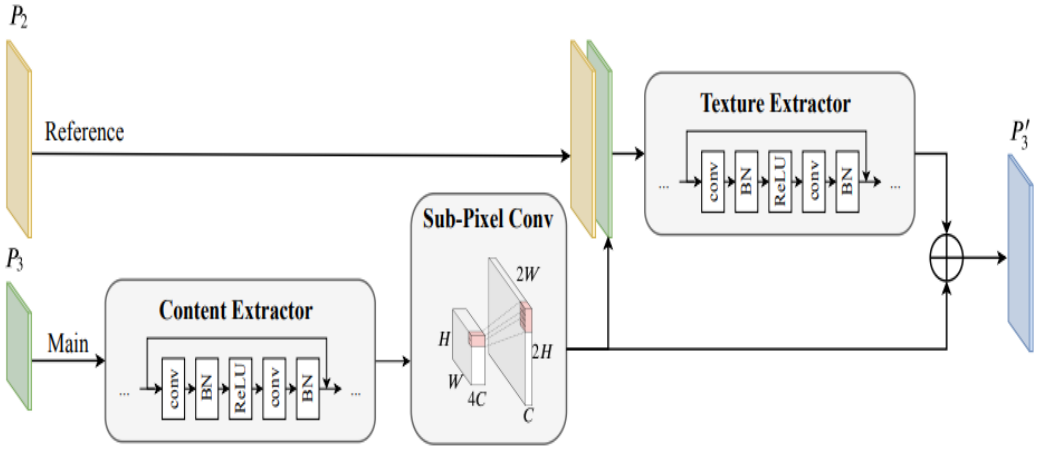

Figure 2.13: FTT Module

small objects have been subpar. The authors argue that merging various scales alleviates the problem the detection of small objects still remains an issue. Along with the neck module, the paper also suggests a loss function that alleviates the area imbalance in the foreground-background. The FTT module allows the combination of low-resolution and high-level semantic features with shallower regional textures. The EFPN enables decoupling of the small and medium-sized objects. The authors find significant improvement using the EFPN on standard data sets compared to the original FPN.

2.2.8 Trident Feature Pyramid Network

C.Picron et al [25] comes to the same conclusion as GFPN [26] that increasing the self-processing or computation in the neck instead of the backbone gives a performance. To this end, the authors propose Trident Pyramid Networks which enables better communication among layers and also better self-processing. The Trident Pyramid allows control over the neck design with hyperparameters allowing easier tuning of communication layers and self-processing layers. The authors report an increase in the performance of the Trident Pyramid Network when compared to BiFPN.

2.2.9 Graph Pyramid Network

G. Zhao et al [26] proposes Graph Feature Pyramid Networks which can change their topological architectures to accommodate support for dynamic feature transfer across all scales and image structures. The authors create a superpixel hierarchy appropriate for each image, this superpixel hierarchy is used as the structure source for the graph feature pyramid network. Local attention of two types is also introduced for this graph feature pyramid network by generalizing global channel attention for CNNs.

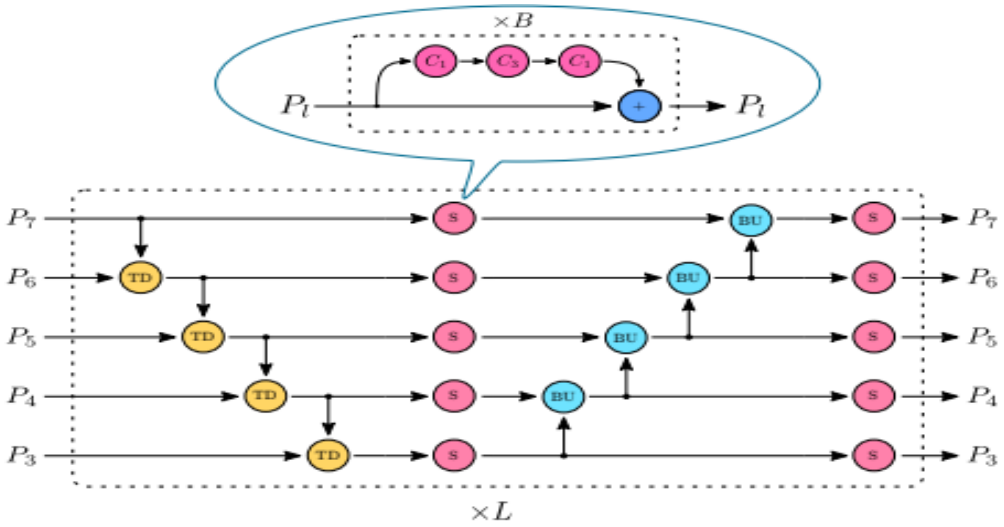

Figure 2.14: Trident Feature Pyramid Network

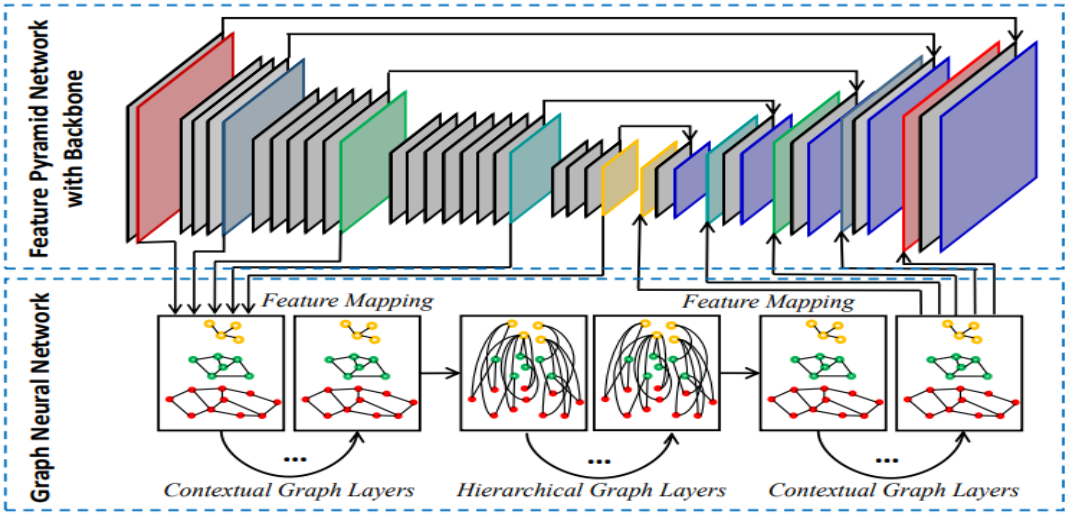

Figure 2.15: Graph Pyramid Network

Chapter 3

Proposed Methodology

The objective of our proposed method is to improve the neck of an object detector so that we can get better performance in the domain of waste detection. To this end so far we have proposed an inception module-based layer connecting the backbone with the neck for better feature aggregation or fusion. In this section we first describe the neck portion of an object detector, followed by the inception module then we explain our intuition for putting the inception module to combine the backbone and the neck. A diagram of our architecture is shown below.

3.1 Balanced Recursive Feature Pyramid Network

We propose an architecture that combines the feature-balancing properties of Libra R CNN [27] with the Recursive feature Pyramid Network of DetectoRs architecture. This is achieved by passing the outputs of the Recursive Feature Pyramid through the feature balancing steps of Libra-R-CNN so that the final output is rescaled, averaged, and refined using a non-local Gaussian attention module. Combining these two architectures ensures that feature extraction not only benefits from a feedback connection to the backbone layers but that the feature level imbalance is also mitigated during the training process. Finally, the non-local Gaussian attention module enhances the final balanced output.

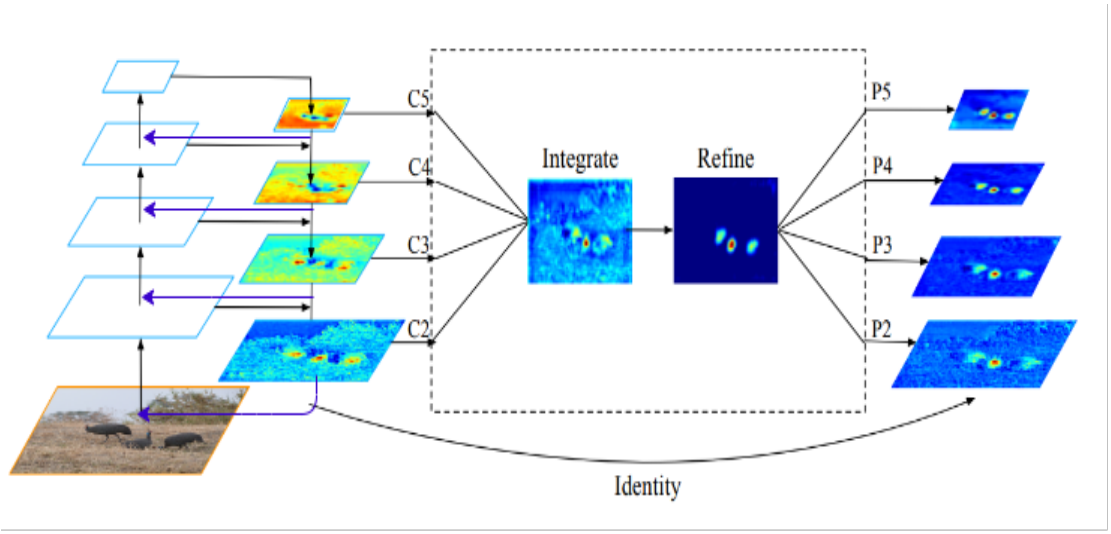

Figure 3.1: Balanced Recursive Feature Pyramid Network

3.2 Components of our Detector

3.2.1 Backbone

Resnet [28]of depth 50 was used as the backbone which is the same backbone that was used for the original Cascade R CNN for comparison. The backbone was kept the same to ensure that this component did not affect the overall results, allowing only the neck to be the main factor. This backbone acts as the bottom-up path of the overall architecture with the topmost layer being the most semantically rich.

3.2.2 Recursive Feature Pyramid

The macro level improvement proposed in the detectors paper called Recursive Feature Pyramid(RFP) is a technique that improves the performance of the Feature Pyramid Network (FPN) by the addition of a new feedback connection from the FPN [21] to the backbone of the architecture. By unrolling this recursive structure to produce an incremental implementation, RFP allows the object detector's backbone to process images several times, resulting in incrementally effective representations. RFP, in essence, iteratively improves FPN to produce more robust and accurate object detection results.

Each Feature Pyramid Networks (FPN) layer is stacked T times in recursive Feature Pyramids. The diagram at Fig 3.2 depicts the two-stage unrolling of Recursive Feature Pyramids, with the blue-shaded region representing the backbone utilized for the first

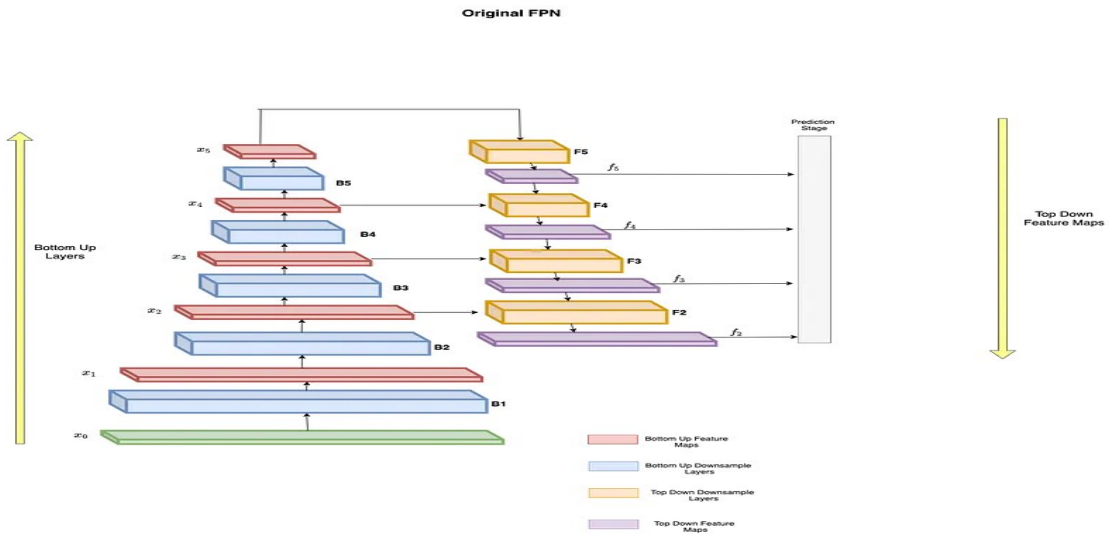

Figure 3.2: Recursive Feature Pyramid Network

stage of bottom-up FPN operation. Before returning the prediction feature mappings f(i) to the backbone, they are converted using the ASPP module, which employs the idea of atrous convolution. R(i) is the name given to this transformation on the feature map f(i). Recursive Feature Pyramids can build more robust representations by adding ASPP, boosting the object detector's accuracy and dependability.

Several feature fusion and attention layer strategies have been shown to be highly effective in deep networks for detecting objects of various scales with a respectable confidence score. The authors have included a Fusion Module to improve the information stored inside the maps and increase object detection performance. This module employs top-down Feature Pyramid Networks (FPN) operation feature maps (f(i)s) from various FPN stages and integrates them in their design with an implicit attention gate. This strategy, as shown in the picture, can considerably increase the detector's accuracy and reliability by successfully integrating data from different sources.

3.2.3 Balanced Feature Pyramid

For balanced learning to take place at the feature level, integration of Deep High-level features and Low-level features should be such that, the end result should possess balanced information across all the available resolutions. However, the sequential manner of integration that architectures like FPN and PaNet have, allows for more focus to be placed upon adjacent resolution than others. Hence the semantic information present in non-adjacent layers gets diluted once per fusion during the flow of information.

To mitigate this imbalance, a balanced feature pyramid works in two stages:

1. Feature Scaling: The output features of the feature pyramid network are first resized to a single feature size, in this case, it is C4. This is achieved via max pooling and interpolation respectively. The next step involves obtaining the balanced semantic features by simple averaging. After balanced semantic features are obtained, the features are then rescaled to their original shape using the same but reverse process. No parameter is involved with this procedure.
2. Refinement: Refinement involves refining the balanced semantic features obtained from the previous step. The end product from refinement will lead to the features being more discriminative. It was found that the non-local Gaussian attention module is more stable in this process compared to refinement using convolutions directly. The refinement process is thus conducted using embedded non-local Gaussian attention. Thereby enhancing the already balanced features.

3.2.4 DetectoRS

DetectoRS was chosen as the detector. We change the neck of this detector to our proposed neck. The authors proposed macro-level and micro-level changes. We only use the micro-level changes.
nd in the micro level of the proposed architecture for object detection, the authors add a new technique named Switchable Atrous Convolution (SAC). SAC uses location-dependent switch functions that operate at different atrous rates. The procedure works on the same input feature. This results in a feature map that has different switches which controls how the output of SAC will look like. The convolutional structure on the backbone in the ResNet [28] was replaced with SAC for the use of SAC in the detector. This change significantly improves the detector's performance, making it more accurate and reliable.

Atrous convolution is an effective method for expanding the field of view of filters in any convolutional layer. What atrous convolution does is, with an atrous rate r, it adds r-1 zeros between the filter values. Thus converting the kernel size of a k x k filter to k = k + (k - 1)(r - 1). This is done without adding more parameters or computation to the overall architecture. The receptive field is an important notion in Convolutional Neural Networks because a bigger kernel size can improve feature map output context information. However, this comes at the expense of more computations. As a result, atrous convolution establishes parity between the parameter count and the computational cost to keep a broad receptive field.
While dilated convolutions with higher atrous rates are good at acquiring contextual information, they may fail to pick up local information from smaller objects on oc-

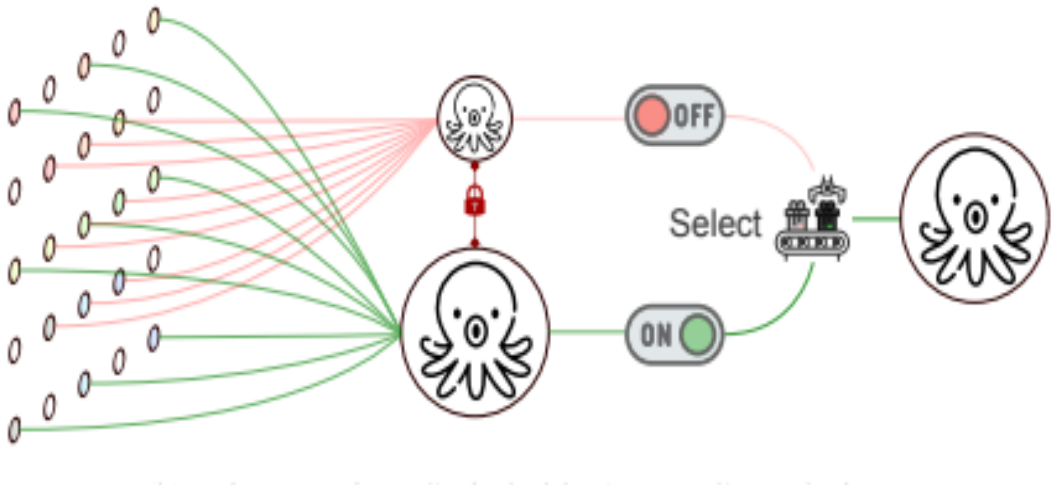

Figure 3.3: Atrous Convolution

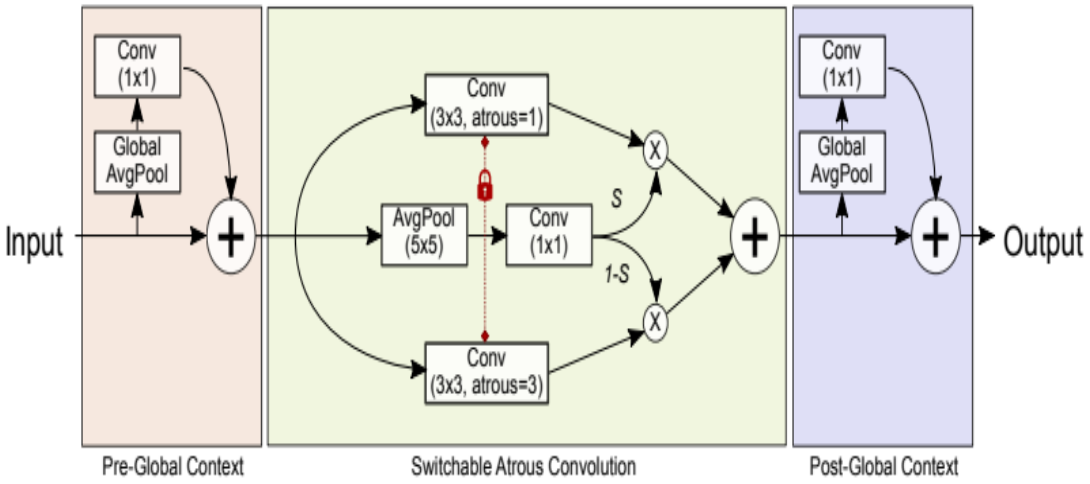

Figure 3.4: Switchable Atrous Convolution

casion. In order to address this issue, the authors developed a novel idea called conditional convolutions, specifically Switchable Atrous convolutions (SAC). SAC is a dependable option for pre-trained backbone networks, allowing for greater flexibility in capturing both local and contextual data. The object detector's performance is improved by adding SAC, making it more accurate and reliable while dealing with objects of varying sizes.
Now to improve the performance of the ResNet Residual blocks, all the standard 3x3 convolution operations were replaced by the dilated convolution which uses SAC structure with two different atrous rates. The figure above shows the modification. This change results in more flexibility in capturing contextual information. Also the authors proposed a switch function S(x), which consists of a 5x5 kernel Average Pooling Layer in addition to 1x1 convolution block. Using pooling layers, this switch can collect statistics on the feature map, which can then be utilized to detect objects at various scales. With this above most modification, the ResNet backbone becomes capable of tackling the challenges of detecting objects in complex scenarios.
As an additional modification to improve performance, the authors have incorporated a global context mechanism before and after each of the 3x3 blocks. These mechanisms consist of lightweight 1x1 convolution kernels and a global average pooling layer. The use of global information is deemed more effective for stabilizing the switch function S(x) that governs the switching between two different convolution operations. By leveraging these mechanisms, the object detection backbone can better capture global contextual information while remaining computationally efficient.

3.3 Training Methodology

We coded our Balanced Recursive Feature Pyramid architecture in the mmdetection library [29]. The base code used for DetectoRs was the default code available in mmdection. The images in Zerowaste and Flow dataset were all resized to 1333*800. We used basic augmentation like horizontal, and vertical flipping. The backbone used in our experiments was the ResNet50, which did not use pre-trained weights. Our model was updated with each higher validation score. We used the Adam optimizer [30] and kept a learning rate of 0.0001 with weight decay.

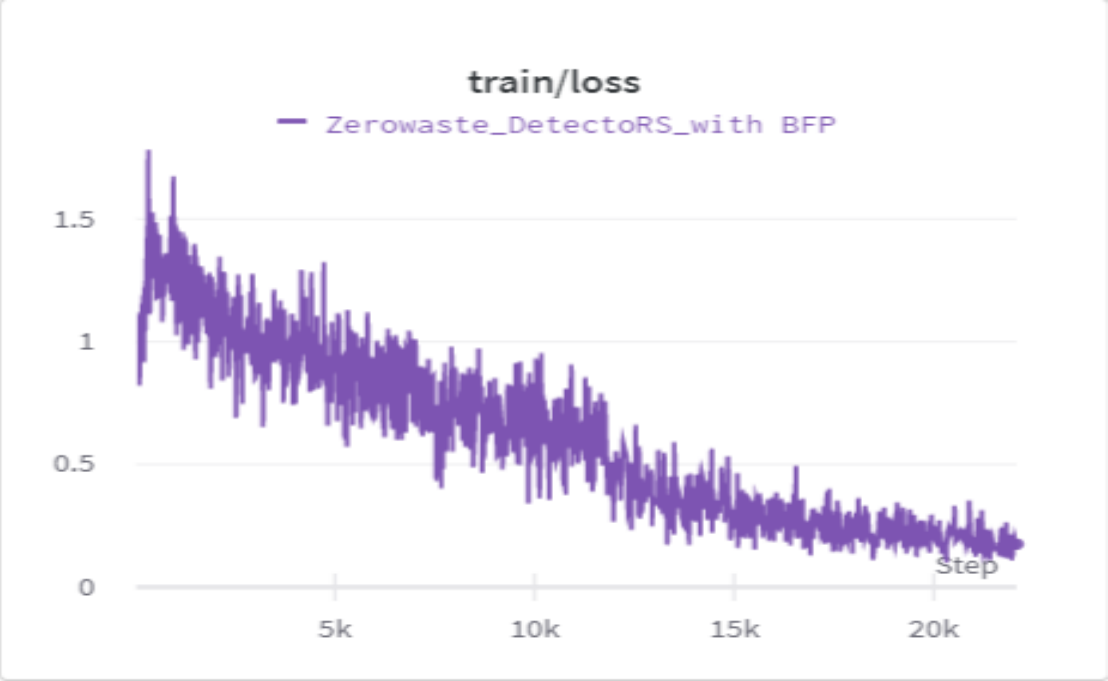

Figure 3.5: Learning Rate Zerowaste

Figure 3.6: Learning Rate Flow

Chapter 4

Results and Discussion

4.1 Datasets

4.1.1 Flow Dataset

Flow data set: First openly available dataset that focuses on inland waste data sets. Consists of images of wastes floating on inland waters. There are 2000 images in total and the class of objects the annotations exist for is 1, bottle. The number of small objects present in the data set is in the majority.

4.1.2 ZeroWaste Dataset

ZeroWaste:The largest openly available cluttered waste dataset.Consists of 4661 images present on a cluttered background. The main objective of this data set is to depict waste in cluttered scenes. There are 4 classes in the data set which include paper, plastic, metal, and glass.

4.2 Our results

Table 4.1 contains the results of base architectures on Flow and ZeroWaste. From a number of models, Cascade R-CNN has the highest mAP 0.5:.95 of 43.4 on Flow. DetectoRs was a close second with 43.0. The other models are significantly lower. When looking at ZeroWaste we can see DetectoRs performing the best with a mAP 0.5:.95 of 30.22, while the other models are around the mid-20s in mAP 0.5:.95. Based on the

Table 4.1: Performance of different models on Flow and ZeroWaste

Model	Flow	ZeroWaste
Cascade R-CNN [13]	43.4	N/A
FPN [13]	33.4	N/A
DSSD [13]	27.5	N/A
RetinaNet [13]	24.9	21.0
Mask R-CNN [12]	N/A	22.8
TridentNet [12]	N/A	24.2
DetectoRs	43.0	**30.22**
DetectoRs-BFP(Ours)	**43.91**	**31.46**

Table 4.2: Comparison of different architectures on FloW and ZeroWaste dataset

Dataset	DetectoRS (mAp 0.5:0.95)	DetectoRS-BFP (Ours) (mAp 0.5:0.95)	change (mAp 0.5:0.95)
Flow-test	43.0	**43.91**	**+.91**
Zerowaset-test (same padding)	30.22	**31.46**	**+1.24**

initial results we decided to pick DetectoRs as our base detector.

Table 4.2 contains the comparison between base DetectoRs and our modified DetectoRs which includes our proposed Balanced Recursive Feature Pyramid. We can see on both datasets our proposed model outperforms the base DetectoRs. On the Flow dataset, we see an increase in mAP 0.5:.95 by +.91 and on the Zerowaste dataset by +1.24.

If we look at the performance of our model when compared with the performance of models in Table 4.1 we can see that it our performs all other models available in the literature on both Flow and Zerowaste dataset.

Chapter 5

Conclusions

5.1 Summary

In our work so far, we have experimented with two waste-based data sets ZeroWaste and Flow. We also benchmarked widely used architectures. Our contribution was to add the Balanced feature pyramid with the neck of DetectoRs, Recursive pyramid network. By adding the Balanced feature pyramid it strengthens the multi-level features of the recursive feature pyramid using the same deeply integrated balanced semantic features. We have seen significant improvement in DetectoRs with the addition of a balanced feature pyramid network. With The new neck-balanced recursive feature pyramid network, DetectoRs outperforms itself by 1.24 percent on ZeroWaste and by .91 percent on the Flow dataset. Our model as per our knowledge gives us the best performance in the ZeroWaste dataset and gives us comparable results with State of the art in the Flow dataset.

5.2 Future Work

Future work can be done on the neck by exploring different ideas with the integration of MHSA (Multi Head Self Attention), or by trying different design choices such as proposing heavier necks altogether.

References

[1] Z. Zou, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," 2019. [Online]. Available: https://arxiv.org/abs/1905.05055

[2] Y. Amit, P. Felzenszwalb, and R. Girshick, "Object detection," *Computer Vision: A Reference Guide*, pp. 1–9, 2020.

[3] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[4] Z. Tan, J. Wang, X. Sun, M. Lin, H. Li *et al.*, "Giraffedet: A heavy-neck paradigm for object detection," in *International Conference on Learning Representations*, 2021.

[5] S. Majchrowska, A. Mikołajczyk, M. Ferlin, Z. Klawikowska, M. A. Plantykow, A. Kwasigroch, and K. Majek, "Deep learning-based waste detection in natural and urban environments," *Waste Management*, vol. 138, pp. 274–284, 2022.

[6] A. B. Stambouli and E. Traversa, "Fuel cells, an alternative to standard sources of energy," *Renewable and sustainable energy reviews*, vol. 6, no. 3, pp. 295–304, 2002.

[7] W.-L. Mao, W.-C. Chen, H. I. K. Fathurrahman, and Y.-H. Lin, "Deep learning networks for real-time regional domestic waste detection," *Journal of Cleaner Production*, vol. 344, p. 131096, 2022.

[8] A. M. King, S. C. Burgess, W. Ijomah, and C. A. McMahon, "Reducing waste: repair, recondition, remanufacture or recycle?" *Sustainable development*, vol. 14, no. 4, pp. 257–267, 2006.

[9] B. Ma, X. Li, Z. Jiang, and J. Jiang, "Recycle more, waste more? when recycling efforts increase resource consumption," *Journal of Cleaner Production*, vol. 206, pp. 870–877, 2019.

[10] A. B. Wahyutama and M. Hwang, "Yolo-based object detection for separate collection of recyclables and capacity monitoring of trash bins," *Electronics*, vol. 11, no. 9, p. 1323, 2022.

[11] A. M. F. Durrani, A. U. Rehman, A. Farooq, J. A. Meo, and M. T. Sadiq, "An automated waste control management system (awcms) by using arduino," in *2019 International Conference on Engineering and Emerging Technologies (ICEET)*. IEEE, 2019, pp. 1–6.

[12] D. Bashkirova, M. Abdelfattah, Z. Zhu, J. Akl, F. Alladkani, P. Hu, V. Ablavsky, B. Calli, S. A. Bargal, and K. Saenko, "Zerowaste dataset: Towards deformable object segmentation in cluttered scenes," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 21 147–21 157.

[13] Y. Cheng, J. Zhu, M. Jiang, J. Fu, C. Pang, P. Wang, K. Sankaran, O. Onabola, Y. Liu, D. Liu *et al.*, "Flow: A dataset and benchmark for floating waste detection in inland waters," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10 953–10 962.

[14] P. F. Proença and P. Simões, "Taco: Trash annotations in context for litter detection," *arXiv preprint arXiv:2003.06975*, 2020.

[15] M. Kraft, M. Piechocki, B. Ptak, and K. Walas, "Autonomous, onboard vision-based trash and litter detection in low altitude aerial images collected by an unmanned aerial vehicle," *Remote Sensing*, vol. 13, no. 5, p. 965, 2021.

[16] M. S. Fulton, J. Hong, and J. Sattar, "Trash-icra19: A bounding box labeled dataset of underwater trash," 2020.

[17] M. Fulton, J. Hong, M. J. Islam, and J. Sattar, "Robotic detection of marine litter using deep visual detection models," in *2019 international conference on robotics and automation (ICRA)*. IEEE, 2019, pp. 5752–5758.

[18] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125.

[19] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8759–8768.

[20] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: Scalable and efficient object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 10 781–10 790.

[21] S.-W. Kim, H.-K. Kook, J.-Y. Sun, M.-C. Kang, and S.-J. Ko, "Parallel feature pyramid network for object detection," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 234–250.

[22] S. Seferbekov, V. Iglovikov, A. Buslaev, and A. Shvets, "Feature pyramid network for multi-class land segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 272–275.

[23] C. Deng, M. Wang, L. Liu, Y. Liu, and Y. Jiang, "Extended feature pyramid network for small object detection," *IEEE Transactions on Multimedia*, vol. 24, pp. 1968–1979, 2021.

[24] S. Qiao, L.-C. Chen, and A. Yuille, "Detectors: Detecting objects with recursive feature pyramid and switchable atrous convolution," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 10 213–10 224.

[25] C. Picron, T. Tuytelaars, and K. ESAT-PSI, "Trident pyramid networks for object detection," 2022.

[26] G. Zhao, W. Ge, and Y. Yu, "Graphfpn: Graph feature pyramid network for object detection," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 2763–2772.

[27] J. Pang, K. Chen, J. Shi, H. Feng, W. Ouyang, and D. Lin, "Libra r-cnn: Towards balanced learning for object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 821–830.

[28] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[29] K. Chen, J. Wang, J. Pang, Y. Cao, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Xu *et al.*, "Mmdetection: Open mmlab detection toolbox and benchmark," *arXiv preprint arXiv:1906.07155*, 2019.

[30] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.