Spot the Ad! - An analysis of object detection models for identifying harmful advertising exposures in the Kids Online project.

Gabby Hawke

February 23, 2025

Abstract

As commercialism and over-consumption are continuously promoted and encouraged, children are increasingly at risk of exposure to these efforts and the adverse problems they cause. The Kids Online project aims to analyse children's real-life online experiences to quantify their exposure to harmful advertising. The project relies on the highly labour-intensive manual annotation of hundreds of hours of video content. This paper explores applying deep learning-based object detection models to automate this process. In recent years, object detection has seen tremendous strides in terms of accuracy and speed, particularly in YOLO and DETR architectures. Through comparative analysis, the YOLOv8 model demonstrated the strongest performance in identifying and quantifying exposures, balancing speed and accuracy while requiring minimal parameter optimisation. This research highlights the benefits of transfer learning, allowing models to generalise well to new datasets with relatively small amounts of labelled data. Furthermore, this study demonstrates how frame-level detection can be leveraged to analyse dynamic video data in a way that provides meaningful insights for public health researchers. The pre-processing and post-processing methods introduced in this paper cut down on processing time and create structured exposure intervals to accurately measure exposure count and duration to understand the extent of harmful advertising. Despite these promising results, certain limitations remain. The model's performance is still far from human capacity, heavily reliant on the dataset's quality and failing to interpret contextual information. Nonetheless, this work proves that an automated approach to detecting exposures is possible, opening the door for future research and improvements for Kids Online.

Acknowledgements

I would like to express my sincere gratitude to my academic supervisor, Jeremiah Deng, for his invaluable support, guidance, and encouragement throughout this project. His insights and expertise have been pivotal in shaping and developing my research. I deeply appreciate and admire the time and dedication he invests in all his students. I am also grateful to the Kids Online team for their unwavering support and collaboration. In particular, I would like to thank Louise Signal, Libby McNaughton, Moira Smith, and Ryan Gage for their valuable feedback, thoughtful discussions, and ongoing assistance, which have greatly contributed to the development of this work.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Problem Overview

Consumerism and over-consumption are continuously promoted and encouraged in our highly commercialised society. We are increasingly exposed to marketing efforts, bound to sway our subconscious favourably for any company. Ethics and advertising standards are questioned when children are exposed to these efforts. According to the advertising standards authority, "occasional food or beverage products must not target children or be placed in any media where children are likely to be a significant proportion of the expected average audience" [1]. Occasional food and beverage products are consumables that do not meet the Food Standards Australia and New Zealand guidelines [1]. The Kids Online project has been developed to observe "children's real-life experiences" [2] and examine the extent of the harmful advertising content children are being exposed to online. This project has accumulated hundreds of hours of video data that needs to be manually analysed to understand the extent of this issue. This manual process has proven to be highly time-consuming and resource-draining, which begs the question of whether there is an object detection model that can carry out this process instead.

As humans, we can see an image and depict all its contents instantly, allowing us to navigate life with minimal conscious thought [3]. Object detection in machine learning strives to match this accuracy and pace to give computers human-like abilities. It is defined as locating and identifying large and various objects in a given image [3]. Since the popularisation of convolu-

tional neural networks (CNN), object detection has seen tremendous strides in accuracy and speed. In particular, a You Only Look Once (YOLO) model and a real-time detection transformer (RT-DETR) model can balance this trade-off and produce accurate, real-time results ideal for a dataset of such a large volume.

1.2 Research Aims

This project aims to evaluate the current deep learning object detectors (especially YOLO and DETR architectures) and their performance in object detection from the Kids Online video data and find potential solutions for performance improvement. Specifically, we intend to answer the following questions:

- What are the performance metrics of current object detection models?
- What benefits can be gained from carrying out transfer learning? What are the implications or requirements for using collected data for fine-tuning?
- What statistical information can be derived from object detection to inform Public Health researchers about the extent of harmful advertising content?

1.3 Organisation of the Thesis

This research design adopts a goal-driven approach. First, in Chapter 2, a literature review will cover Kids Online's precursor work and the project's significance from a marketing perspective. Additionally, previous object detection work will be reviewed. Chapter 3 is the methodology for developing the models, which is how transfer learning can be utilised and how it affects the performance of these models. Chapter 4 examines how object detection models can be adapted to produce meaningful results for the Kids Online project, including being able to input video data and produce exposure intervals. Chapter 5 summarises and discusses this project's findings and potential future works.

Chapter 2

Literature Review

2.1 The Significance of Kids Online

2.1.1 The impact of harmful advertising

All marketing towards children has the potential to harm their health [4], and the prevalence of unhealthy food options continues to be a particularly damaging influence. The World Health Organisation has identified that childhood obesity rates are becoming a more serious threat worldwide, noting its alignment with the United Nations Sustainable Development Goals set in 2015, prioritising the prevention of non-communicable diseases [5]. A core recommendation by the World Health Organisation to address this issue is to promote healthier food options [5]. There is a significant disparity in advertising between the promoted food and beverages and the recommended food intake for a balanced diet [6]. Research revealed that while food advertising constitutes a substantial proportion of the total advertising children are exposed to, a tiny proportion of these promoted healthy food options [5]. Many studies have made direct causal links between exposure to unhealthy food advertising and children's food behaviour, knowledge, culture, psychological well-being and dietary-related outcomes [7][8][9][4]. Many other factors influence children's dietary habits, such as parental influence, lifestyle factors and social and economic status. However, advertising is an equally significant factor [8], with evidence consistently showing that promoting unhealthy foods is a modifiable risk for non-communicable diseases [9].

This new generation of children has much greater spending power with

higher disposable incomes and more significant influence over their household's purchasing decisions [9][10][11]. Investments in marketing towards children rose 150-fold in only 20 years [12]. Marketers increasingly view children as a profitable market segment, employing emotional appeals of fun, maturity and success while promoting materialism to express wealth and social status [12][8]. These tactics aim to shape how children view themselves, their relationships and the world at large, which can be detrimental during their developmental stages [10]. This evidence reinforces the unfavourable perceptions associated with commercialisation, namely its prioritisation of profit above all else [10]. Consumer culture is being forced upon children at earlier ages through advertising, distorting reality and eroding childhood experiences [13]. Children who spend more time in front of a screen are shown to possess more materialistic traits, have lower self-esteem and report that they would be happier if they had the money to buy more things for themselves. The lines between entertainment and advertising are blurred, with unhealthy commodities being promoted even in programs and websites specifically designed for children [14][15]. Even if a particular child does not see the advertisement directly, it will inevitably reach them through peer pressure [16].

2.1.2 Prior Work

Research in this area has mainly relied on self-reports or observational studies conducted in controlled environments [17], often not accurate representations of actual advertising exposures. The Kids Cam Project [17] was the first study of this kind to collect real-life data. One hundred sixty-eight children, aged 11 to 14, from schools in the Wellington region, wore cameras around their necks for four consecutive days (Thursday to Sunday), taking pictures every 7 seconds and capturing 136° of whatever was in front of them. This research showed that the children were exposed to 'non-core' foods 27.3 times a day on average, where 'non-core' foods are classified as not recommended to be marketed to children. Most of the exposures from this group were made up of sugary drinks and fast food advertisements [17]. Watkins et al. [18] found that children in this study were exposed to a brand nearly every minute, emphasising the highly commercialised world we live in and how these advertisements seriously threaten children's psychology. Although the data from Kids Cam provided valuable insights, it recognised that even though children accessed the internet during the observed period, the cameras

could not capture sufficient content from their devices [17].

2.1.3 Advertising in Digital Media

Previous research is mainly centred around television advertising, which was the primary media source for children in the early 2000s. However, as we shift our attention to online media, we must also focus on how digital media affects children's behaviours [19]. It is reported that children aged between 5 and 15 spend an average of 2 hours each weekday online and 3 hours each weekend day online [14]. The internet has allowed marketers to reach children more frequently and meaningfully. 98% of websites created for children allow for advertising, and two-thirds of these websites solely rely on this as their primary source of revenue [15]. One study that looked at internet traffic data showed that marketing techniques were more frequent on websites targeted at children [20]. Online advertising allows for the unique ability to create content that users can interact with, inducing high levels of involvement and fostering intimate brand relationships that encourage the purchase decision to be made at an unconscious level [11]. Further, user-generated content (UGC) is a strategy marketers use to encourage and extend their market reach and capitalise on perceived positive word of mouth. Digital media allows for seamlessly woven together promotional events, which are no longer seen as interruptions but rather entertainment [11]. Similar work in this area shows strong evidence linking online advertising to children's food preferences and intent for food purchases [15].

2.1.4 Kids Online Project

The Kids Online study has been established to examine real-life exposure to unhealthy food commodities in digital media [2]. One hundred fifty-six students across the Wellington region have been selected and observed for four consecutive days (Thursday to Sunday). Participants are encouraged to use Zoom to share their screen content whenever they access the internet on their phone, laptop and/or tablet. This data is the first of its kind to be collected. Further information about the data collection, sampling methods and ethical approvals for Kids Online is detailed elsewhere [2]. This project has accumulated 256 hours of content, which takes months to analyse manually, consuming significant time and resources. Additionally, manual evaluation is subject to human error, which poses the question of whether a more ef-

ficient and automated method for analysing this data can be developed to accurately inform about the extent of digital advertising.

2.2 Object Detection

Object detection is defined as locating and identifying various predefined objects within an image [21], which is not an easy task considering the vast amount of lighting, quality and viewpoints possible for any given object [22]. It is an area of machine learning which has attracted much attention from scholars over the years. Many real-world applications employ object detection methods, such as surveillance applications, robot vision, and, for our purposes, brand detection.

2.2.1 Performance Metrics

Object detection is measured in accuracy and speed [22]. The term "accuracy" includes a model's ability to classify an object into the correct class and identify the object's position within the image, referred to as localisation [23]. A model's prediction outcome is the location or bounding box where the object exists, the predicted class, and a confidence level [24]. Average precision describes the model's overall accuracy [22]. Localisation is calculated using the intersection over union (IoU) method, which measures the difference between the predicted bounding box B_p and the ground truth bounding box B_{gt},

$$J(B_p, B_{gt}) = \text{IOU} = \frac{\text{area}(B_p \cup B_{gt})}{\text{area}(B_p \cap B_{gt})}$$

[24].

If these scores are above a predefined threshold, for example, if IOU$\geq t$, then the image is correct; otherwise, it is classified as incorrect [22][24]. These predictions are then classified into true positives, false negatives or false positives. In this context, it is impossible to classify true negatives because there are an infinite number of true negative locations. Therefore, measures such as precision and recall are the typical choices for performance metrics [24]. The average precision is then calculated by measuring the area under the curve (AUC) of the precision and recall metrics. The mean average precision is the average precision over all classes,

$$\text{mAP} = \frac{1}{N} \sum_{i=1}^{N} AP_i$$

[24].

The mAP is also often analysed over multiple thresholds, for example, between 0.5 and 0.95 (mAP50:95). This promotes more precise localisation, making it more suited to real-world applications [22]. In addition, specific datasets are used to train object detection models, enabling performance comparisons and benchmarks. Pascal VOC07 [25] was commonly used from 2005 to 2015. It consisted of 5,000 training images and 12,0000 annotated objects with 20 classes. MS-COCO datasets [26], such as COCOval2017 [27], have since become the new standard as they are much more challenging, containing 160,000 images, 897,000 annotated images, and 80 different classes [22].

2.2.2 Object Detection Overview

The history of object detection can be divided into two distinct eras: traditional methods, which occurred before 2014, and the integration of deep learning techniques beginning in 2014 [22]. Traditional object detection can be broken into three stages: Informative region selection, feature extraction and classification [3]. These shallowly trained models relied on manually handcrafted features, which were inherently influenced by human perception of objects [22][28]. Traditional object detection models utilised a sliding window feature, which was exhaustive and redundant, creating unnecessary time complexities [3]. Other disadvantages of these models include poor portability, low robustness and lack of flexibility [29]. The idea of artificial intelligence and neural networks was first developed in the mid-1900s. However, it did not become popular until the 2000s due to the emergence of large-scale data, high-power computational systems and advances in machine learning algorithms [3][29]. The emergence of deep neural networks allowed for the introduction of convolutional neural networks (CNN) in the image recognition space, differing greatly from traditional approaches [3][30]. The application of these algorithms can be roughly divided into two categories; the first is a two-stage detection algorithm that generates regional proposals and then classifies these proposals. The second is a single-stage algorithm that turns object detection into a single regression problem, inheriting a

unified framework that classifies and locates objects [3][29].

2.2.3 Two-stage Object Detectors

R-CNN, a two-stage detection model, was the first to incorporate deep convolutional neural networks (CNNs) into object detection [28]. This model was first proposed in 2014 by Girshick [31] and followed a similar structure to traditional methods as shown in Figure 2.1. Regional proposals are extracted through selective search, then the images are resized to fit into the CNN, and finally, a linear support vector machine (SVM) classifier is used to predict the presence of a predefined object within each proposal [22][29]. This model enabled deeper training and eliminated the need for manual feature design; however, it still relied on a multistage pipeline with redundant regional proposals, leading to high time complexities and additionally required fixed input sizes, resulting in image distortion [3][29]. SPP-Net was introduced in 2014 [32], using a spatial pyramid pooling layer to eliminate the need for fixed-sized inputs. This model avoids redundant calculations and achieves improved results but still has several limitations, including a multistage pipeline with high overheads [29][22].

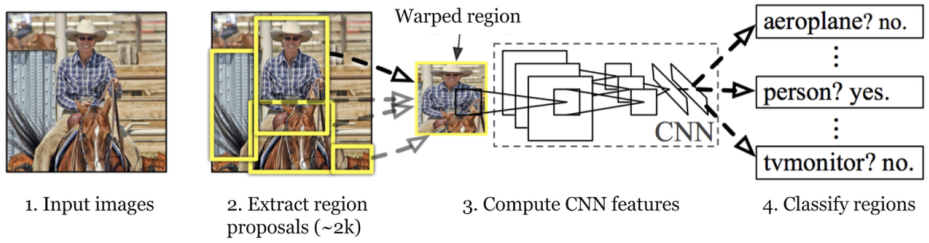


Figure 2.1: R-CNN architecture [31]

In 2015, Fast R-CNN [33] was introduced, which incorporated a pyramid pooling layer, replaced SVM with a Softmax function for classification, utilised a region of interest (ROI) pooling layer, and enabled simultaneous training of the detector and bounding box regressor [29][22]. This model ran over 200 times faster than R-CNN and achieved a mAP50 of 70% on VOC07 [29]. However, it still had the same limitations as traditional-based methods [28]. Faster R-CNN introduces a regional proposal network (RPN), which replaces selective search and eliminates the need for external region proposal generation [28][22]. The Faster R-CNN model architecture, displayed

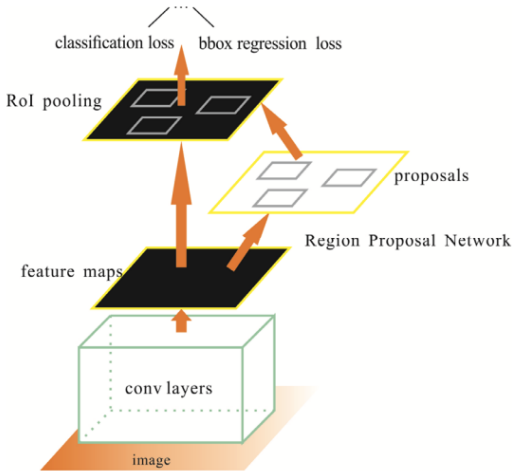


Figure 2.2: Faster R-CNN architecture [29]

in Figure 2.2, shows a clear improvement from the earlier R-CNN architecture with a much simpler structure; however, the distinct model stages still limit this model's speed capabilities [22]. The final significant algorithm in two-stage detection is FPN, which was introduced in 2017 [34]. This method uses a multi-scale feature map, which not only leverages the feature map of the network's top layer but also detects features at all levels, capturing a broader range of information. In conjunction with Faster R-CNN, this method achieved mAP50:95 of 36.2% on the MS-COCO dataset [27] and has become a building block for many future models [22].

2.2.4 One-stage Object Detectors

Although two-stage detectors can reach high accuracies, they are rarely used in real-world applications because of their slow speed and high computational complexities [22]. For instance, Fast R-CNN processes at seven frames per second (fps), while Faster R-CNN processes at five fps [29]. On the other hand, one-stage detectors map straight from the image pixels to the bounding box, significantly reducing the processing time [3]. A You Only Look Once (YOLO) model was the first one-stage detector introduced in 2016 by Redmon [35]. The core idea of this model is that the entire detection occurs within one simple CNN structure, turning object detection into a single regression problem [36]. Figure 2.3 displays the architecture of the first

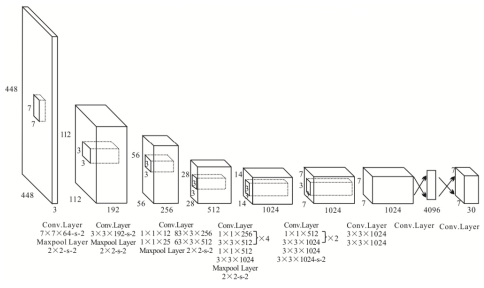


Figure 2.3: YOLOv1 architecture [29]

YOLO model. The first layers of the network turn the image into an S*S grid, and each grid cell then predicts B bounding boxes for contained objects and their corresponding probabilities [29][23]. This model processed at 45 fps but lagged behind other models significantly in terms of accuracy [29]. SSD was released later the same year, which uses the concept of an anchor box and detects using multiple layers to capture multi-scaled objects [22][28]. This model achieved a mAP50:95 of 31.2% [24] at 46 fps [29], a much more comparable result to two-stage detectors.

Since 2016, several new versions of YOLO have been released, which address its earlier limitations [29]. YOLOv2 [37], released in 2017, utilises an improved Darknet-19 backbone architecture, has a more complex CNN structure with pooling layers, and enables batch normalisation, multi-scale training and the anchor box mechanism. [38][29]. Redmon additionally proposed YOLOv3 in 2018 [39], which used an updated Darknet-53 backbone architecture, replaced the Softmax function with a binary cross entropy loss function, and employed an upsampling function method drawing from FPN to adopt a multi-scale framework [29][28][36]. 2020 saw the release of the fourth YOLO model [40], whose main contributions included a new MISH activation function, a closer integration of spatial pyramid pooling, a new generalised IoU loss function and data enhancement through Mosaic data augmentation [36][29]. This model achieved mAP50:95 of 43.5% at 65.7 fps on the MS-COCO dataset [27], outperforming all previous models [29]. Later, in 2020, Glen Jocher (Ultralytics) released an open-source YOLOv5 [41], which was the first to use a PyTorch framework, gaining much popularity [42]. This model further enhanced data augmentation, optimised training strategies, and uses an automated anchor box mechanism leveraging k-means clustering and genetic evolution [38][43]. YOLOv6 [44] and YOLOv7 [45]were

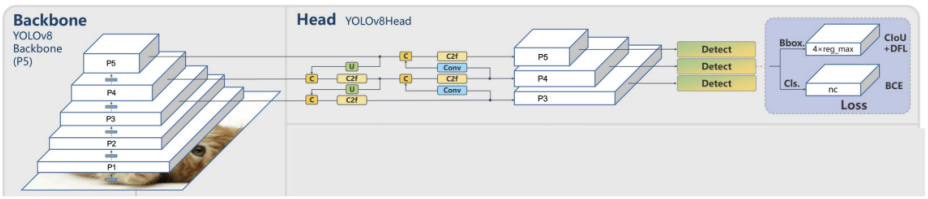


Figure 2.4: YOLOv8 architecture [47]

released in 2022, both introducing new network architectures called EfficientNet and EfficientRep [43]. Both these models improved speed and accuracy and cut costs and computational overheads [43]. Lastly, in 2023, Glen Jocher released YOLOv8 [46]. The architecture of this model is displayed in Figure 2.4. The YOLOv8 model has an improved MISH activation function, does not use anchor boxes, instead predicts the object centres directly and optimises aggregation features through Mosaic augmentation [43]. This model combines and leverages FPN and a Path Aggregation network (PAN) to reduce the spatial resolution, increase the number of feature channels and aggregate through different network layers [47]. The YOLOv8 model performs at a mAP50:95 of 52.9% and 71 fps on the pre-training dataset [48].

Another recent significant contribution to one-stage object detection models is a detection transformer (DETR). This model does not use a traditional CNN structure, but instead, a new transformer-based architecture [49]. This architecture uses an encoder and decoder structure heavily based on self-attention to highlight an image's essential features, enhance generalisation and allow for parallel computations [49]. However, this model renders higher computational costs with a much more complex architecture, meaning they cannot process in real time. An example is the DINODeformable-DETR-R50 model, which reaches a mAP50:95 of 50.9% on the pre-training dataset, processing at five fps [48]. A recently proposed real-time detection transformer (RT-DETR), displayed in Figure 2.5, can overcome this limitation. Although this model is still very complex, it uses a hybrid encoder/decoder to process multi-scale features and increase the model's speed. Additionally, it improves accuracy using an uncertainty-minimal query selection mechanism and supports a flexible number of decoder layers [48]. The RT-DETR-R50 model reaches a mAP50:90 of 53.1% on the pertaining dataset at a speed of 108 fps, outperforming the DETR and YOLO models.

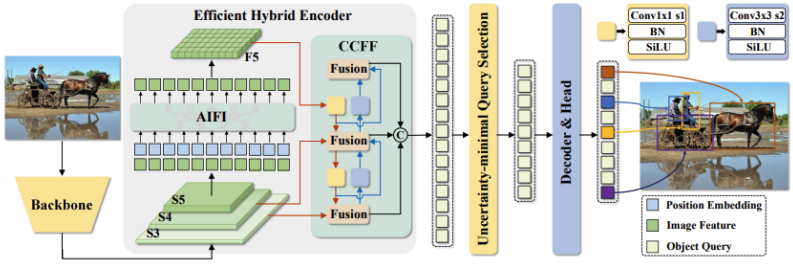


Figure 2.5: RT-DETR architecture [48]

2.3 Similar Work

In 2020, Kuntsche (et al.) [50] created an Alcohol Beverage Identification Deep Learning Algorithm (ABIDLA). This model took a different approach from previous object detection models and focused on classifying the image as a whole rather than using bounding boxes to find specific images. A refined version of this model was also released in 2022 [51]. Without the need to manually annotate the training images, the dataset could be a lot larger, and the later version of the model demonstrated the power of this tactic by employing the use of 'weak labels'. This approach uses keywords attached to the image location to classify the object, eliminating the need for human evaluation. However, because the images were sourced only from Google and Bing, the model is limited in generalising to new, unseen data. Without identifying the features of the object, the model does not understand what it is looking for, resulting in poor performance from unfamiliar images. Additionally, this model cannot quantify the number of exposures within the frame and their location, limiting the ability to perform post-processing and provide valuable statistical information.

In 2021, Palmer (et al.) trained an object detection model to detect harmful advertising, such as fast food and alcohol, in street view images [52]. The proposed method used an Inception V3 architecture, a two-stage architecture released in 2015. Although the accuracy was extremely high at the time of release, it has been overtaken by subsequent models, such as YOLOv4, in terms of speed and accuracy [53]. The data comprised images taken every 0.5 seconds, which results in duplicate advertising exposures in consecutive frames. To address this, they measured the level of similarity in Euclidean distance of the exposure and removed any that were too similar.

During training, the model reached a precision of 0.85, a recall of 0.72 and an F1 score of 0.76. The model identified and extracted significant harmful advertising in street-view imagery [52].

In 2018, ADNet was proposed to detect advertising content in video frames focusing on outdoor scenes [54]. AdNet splits a video into frames and identifies the images containing an advertisement. This model is inspired by a VGG network architecture, which, similarly to the Inception V3 architecture, has been outperformed by YOLO [53]. The proposed ADNet model achieves a 94% accuracy on the video dataset [54]. However, this paper fails to mention how duplicates were handled and the rate at which frames from the videos were extracted to achieve this accuracy. More recently, Ha (et al.) [55] used both YOLOv5 and Faster R-CNN for surveillance in digital media. This analysis showed that YOLOv5 consistently outperformed Faster R-CNN in all test cases. This work demonstrated the effectiveness of transfer learning in object detection to detect harmful advertising; however, similarly to previous work, it focused on the performance of image datasets and could not interpret video content efficiently and meaningfully [55].

Chapter 3

Methodology

3.1 Overview

Fine-tuning is refining the parameters of a pre-trained model to adapt it for a different task [56]. This is an example of transfer learning as it leverages existing knowledge, eliminating the need to build a new model from scratch. Speed is an important consideration for the Kids Online Project because the model needs to keep up with the rapidly changing digital landscape and dynamic nature of online advertising. Many factors influence fine-tuning performance, including resource capacity and the training dataset's quality. Previous work in this area has identified that the choice of model architecture and fine-tuning parameters is very context-specific. Three models have been selected for a comparative analysis. YOLOv5 and YOLOv8 have been chosen as they share the same authorship. YOLOv5 [57] is particularly noteworthy for being the first open-source version of the YOLO series, while YOLOv8 [58] represents one of the latest advancements. Additionally, an RT-DETR model [59] will be included in the analysis to compare the transformer-based architecture to YOLO's traditional CNN-based design. Various parameters will be tested to understand how these models can be optimised for Kids Online. The performance will be analysed in terms of precision and recall, which describe the model's ability to detect and correctly classify all relevant objects in the image. However, in the case of Kids Online, it is also important to distinguish between frames with and without brand exposure, no matter how many objects, so that the total exposure time can be quantified. For this reason, the overall frame detection will also be a consideration. It is

important to note that, in this case, it is better to have false negatives than false positives to prevent unwarranted accusations.

3.2 Data Collection

3.2.1 Class Selection

To quantify the amount of harmful advertising children are exposed to, it is essential to identify the brands responsible. Brands leverage logos and other visual elements to reinforce recognition and create relationships with their audience [16]. These elements are consistent among all marketing channels, making it much easier for models to recognise when a brand is present. Conversely, identifying generic items, such as a burger, is significantly more challenging due to their visual variability. The Kids Cam results revealed significant exposure of children to non-core foods, with sugary drinks and fast-food brands being the most frequently encountered categories[17]. Previous research shows that brands such as Coca-Cola, McDonald's, Burger King and KFC have dominated the advertising space in digital media[11]. To understand the benefits that can be obtained from transfer learning, we first look at soft drinks as they not only have a brand logo but also an identifiable product. For example, in figure 3.1, although the Coca-Cola brand logo is not in the frame, the image is still identifiable as a Coca-Cola bottle. Other soft drinks with a significant online presence include Pepsi, Sprite and Fanta [60]. These four soft drink brands will make up the test case for the previously described models.

3.2.2 Dataset Creation

Transfer learning in object detection involves training an existing model on a custom dataset to recognise new objects of interest. A custom dataset of images needs to be compiled and manually annotated so that the model understands the new object's visual characteristics. Ultralytics [61] recommends 1500 images per class to achieve the best training results. However, this is not feasible due to the resource limitations of this project. Three different datasets will be created to measure the impact of dataset quality: A small dataset where each class has 20-30 instances, a medium-sized dataset with 80-90 cases within each class, and a large dataset with 140-150

Figure 3.1: Validation set image displaying a Coca-Cola product with no logo

instances within each class. Images have been scraped from Google searches and social media websites such as TikTok, Instagram and YouTube. A combination of advertising content and UGC has been collected to reflect the likely characteristics of the Kids Online Data. Roboflow [62] is a tool to support the development of computer vision applications and has been utilised to annotate images. Different classes have been created depending on the brand's identifiable categories. Figure 3.2 shows the four different categories for the brand Coca-Cola: Coca-Cola logo, Coca-Cola bottle, Coca-Cola can, and Coke logo. This approach ensures optimised accuracy by distinguishing between objects with unique features. The same approach has been taken for the other three soft drink brands. After dataset compilation, the small, medium and large datasets have 446, 1142, and 1984 images, respectively. The datasets have been split in three ways: 70% training, 20% validation and 10% testing. Roboflow [62] offers a feature to resize and augment training images, enhancing the diversity of the dataset and delaying over-fitting [61]. For the three datasets, images have been resized to 640x640 (with black borders to fit). Some images have been flipped horizontally or vertically, rotated between +8° and -8°, and applied a grayscale of 10% or a saturation between +20° and -20°. After image augmentation, the small, medium and large datasets have 949, 2619 and 4533 images, respectively.

Figure 3.2: Annotated images from the small dataset showing the four different Coca-Cola classes

3.3 Model Development

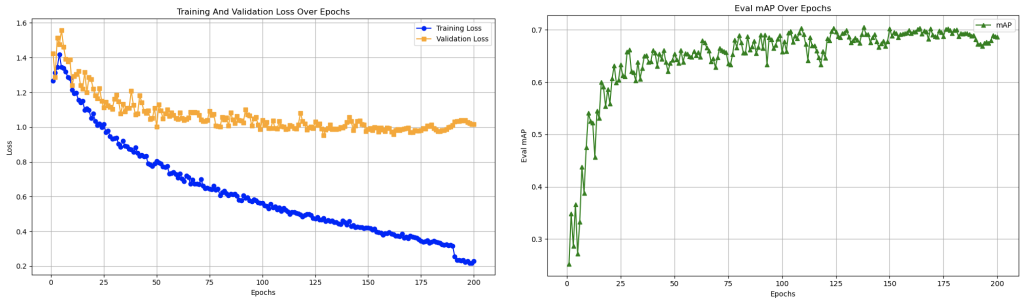
3.3.1 Model Selection

All models come with different-sized pre-trained models that trade-off speed and accuracy and have been trained on the COCO val2017 dataset [27]. Ultralytics recommends using a small or medium-sized model for mobile deployments and a large or extra-large model for cloud deployments [57]. The YOLOv5 models were trained over 300 epochs, achieving mean average precision (mAP050:95) scores of 37.4%, 45.4%, 49.0%, and 50.7% for the small, medium, large, and extra-large models, respectively. The corresponding processing speeds per image for CPU b1 are 0.9ms, 1.7ms, 2.7ms, and 4.8ms using a batch size of 32 [57]. The large YOLOv5 model has been selected for its balanced performance, achieving a relatively high mAP with a fast processing time. YOLOv8 [58] similarly offers the same model sizing and has trained the models on 300 epochs. The small, medium, large and extra large models achieved a mean average precision (mAP50:95) of 44.9%, 50.2%, 52.9% and 53.9%, respectively. The corresponding speeds of these models for A100 Tensor RT are 1.20ms, 1.83ms, 2.39ms and 3.53ms [58]. For consistency, the large model for YOLOv8 has also been chosen. The RT-DETR model [59] also provides four different model size options: R18, R32, R50, and R101 pre-trained over 72 epochs. The mAP50:95 for these models is 46.5%, 48.5%, 53.1% and 54.3%. The speed, measured in frames per second, is 217, 172, 108 and 74, respectively [59]. Again, the large (R50)

model has been chosen.

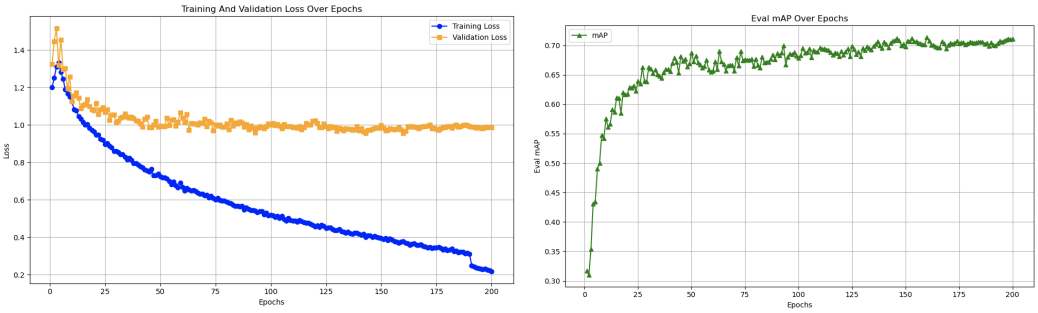
3.3.2 Fine-tuning

To perform transfer learning on YOLOv8, the Ultralytics package [58] is imported. There are 30 hyper-parameters, including learning rate, weight decay, and image size. These parameters have been optimised for the COCO val2017 training dataset [27] and left unchanged. Utilising an NVIDIA A100 with 20 GPU cores, the pre-trained large model weights are fine-tuned three times on the small, medium, and large datasets, each on 200 epochs. Figures 3.3, 3.4, and 3.5 illustrate the mAP50:95 validation and training bounding box loss across each epoch. The small model achieved a mAP50:95 of 0.705 at epoch 138, the medium model reached 0.713 at epoch 160, and the large model reached 0.734 at epoch 189. The model trained on the small dataset took 7 minutes to train, the medium took 24 minutes, and the large took 45 minutes. The performance of these models is all relatively similar; the training loss appears to continue to decrease while the validation loss plateaus between 50-100 epochs. There is no clear evidence of overfitting in these models; however, it may be beneficial to experiment with fewer iterations as the validation loss shows limited improvement beyond 100 epochs. The results show an increasing mAP, highlighting the effectiveness of fine-tuning for YOLOv8. The models successfully learn and adapt to detect valuable classes for the Kids Online Project.



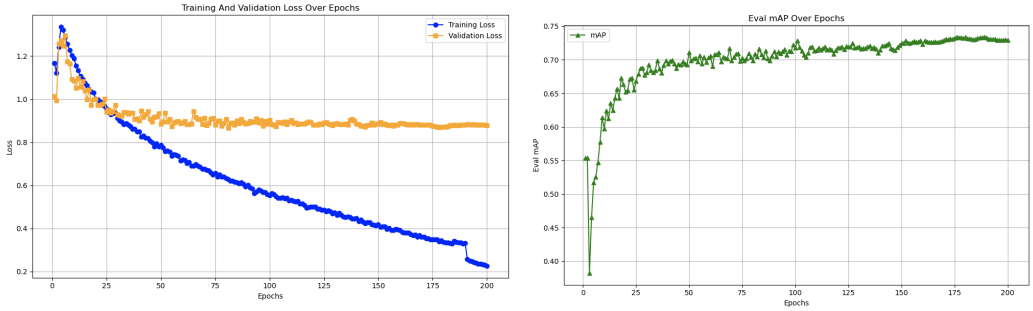
(a) Training and Validation Loss (b) mAP50-90

Figure 3.3: Training results of YOLOv8 using the Small Dataset



(a) Training and Validation Loss (b) mAP50-90

Figure 3.4: Training results of YOLOv8 using the Medium Dataset



(a) Training and Validation Loss (b) mAP50-90

Figure 3.5: Training results of YOLOv8 using the Large Dataset

Transfer learning is also performed on YOLOv5 by cloning the respective Git repository [57] and importing the Torch package. Like YOLOv8, the default hyperparameters were used, as they were optimised during the pre-training. Utilising an NVIDIA A100 with 20 GPU cores, the pre-trained large model weights are loaded and further fine-tuned three times using the small, medium, and large datasets, each on 200 epochs. Figures A.1, A.2, and A.3 illustrate the mAP50:95, validation and training bounding box loss across each epoch for each dataset size. The small model took 9 minutes to train, the medium took 30 minutes, and the large took 52 minutes. For the small model, an optimal value of 0.684 was achieved at epoch 200; for the medium model, 0.672 at epoch 196; and for the large model, 0.697 at epoch 139. The training loss continues to decrease throughout the 200 epochs, and the validation loss plateaus at around 50-100 epochs. The validation loss of the large model shows a slight increase around epoch 125, suggesting potential overfitting; however, this increase is minimal. Similarly to YOLOv8, the results demonstrate that fine-tuning has succeeded, and each model effectively identifies and categorises relevant content.

Finally, transfer learning is also applied to the RT-DETR model. The Torch, Requests, and Transformer packages were utilised, along with the checkpoint from the R50 model [59]. The hyperparameters of the DETR were altered slightly; the medium model required a max gradient norm of 1 compared to the default 0.1, and the large model required a max gradient norm value of 10 and a learning rate of 1e-6. The pre-trained weights were fine-tuned separately for the small, medium, and large models, each over 200 epochs. Figures A.4, A.5, A.6 show the mAP50:95, validation and training loss for each test. Utilising an A100 GPU and a batch size of 20, the model trained on the small dataset took 24 minutes to train, the medium took 90 minutes, and the large took 150 minutes. The validation and training loss combines various parameters, including bounding box loss, varifocal loss and generalised IoU loss, with bounding box loss contributing most of the weighting. The small model reached a maximal mAP of 0.684 at epoch 76, the medium model reached 0.671 at epoch 36, and the large model reached an mAP of 0.563 at epoch 59. The performance of these models differs significantly from the YOLO models. They reach their peak performance much earlier in the training process, after which they experience overfitting. Compared to YOLO, these models struggled to perform well on their default metrics, failing to achieve comparable accuracy. Additional computational resources would be required to optimise these model parameters and prevent

overfitting.

3.4 Model Validation

3.4.1 Data and Performance Metrics

A validation set was created to further evaluate these models, consisting of five 20-minute videos: four recorded on a mobile device and one on a computer. Screen recordings on social media were taken to simulate the Kids Online data, which has short, fast-paced clips. The videos have been purposefully created to have a high frequency of brand exposure. Frames and null occurrences have been extracted for each soft drink group, with 25 to 30 frames selected for each. This validation set evaluates the selected models on a consistent dataset that closely resembles the actual data. The best models for YOLOv5 and YOLOv8 have been selected based on a weighted evaluation, combining 10% mAP50 and 90% mAP50:95 scores. The best RT-DETR model has been chosen based solely on the mAP50:95 score. The nine selected models were tested for each brand across 20 confidence intervals, ranging from 0% to 95%. The raw results are displayed in table A.1. The first performance assessment task analyses whether a brand was present and detected within the frame, regardless of the number or location of prediction.

$$\hat{Y} = 1\left(\sum_{i=1}^{N} D(i) > 0\right), \quad Y = 1\left(\sum_{i=1}^{N} G(i) > 0\right)$$

Where D is the function that returns 1 if a brand is detected in frame i and G is the ground-truth function that returns 1 if a brand is actually within frame i.

$$\text{TPR} = \frac{\sum(Y = 1 \wedge \hat{Y} = 1)}{\sum(Y = 1)}$$

$$\text{FPR} = \frac{\sum(Y = 0 \wedge \hat{Y} = 1)}{\sum(Y = 0)}$$

Since the whole frame is the prediction, it allows for quantifying true negatives, as there is no longer an infinite amount. The TPR and FPR are presented using an ROC curve, and the AUC statistic is calculated. The equal error rate (EER) was used to find the optimal threshold. The second

performance assessment task evaluates the number of brand detections compared to the actual occurrences, providing precision and recall metrics and an F1 score to compare these results.

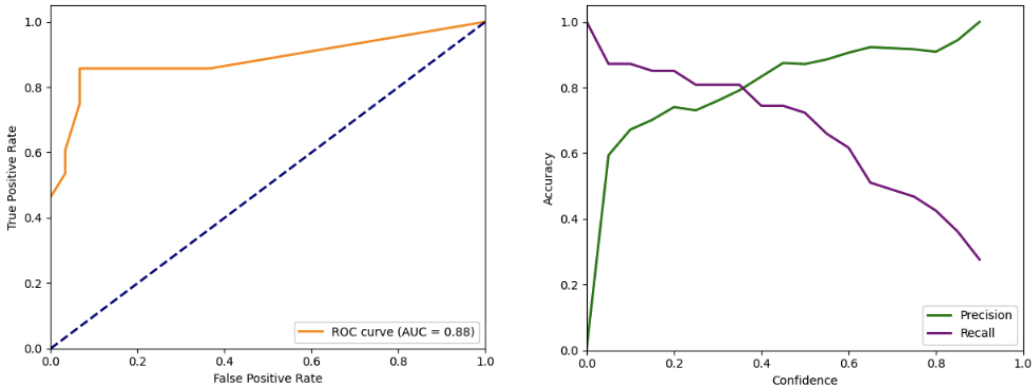
$$\text{Precision} = \frac{TP}{TP + FP}, \quad \text{Recall} = \frac{TP}{TP + FN}$$

$$F1 = 2 \cdot \frac{P \cdot R}{P + R}$$

3.4.2 Results

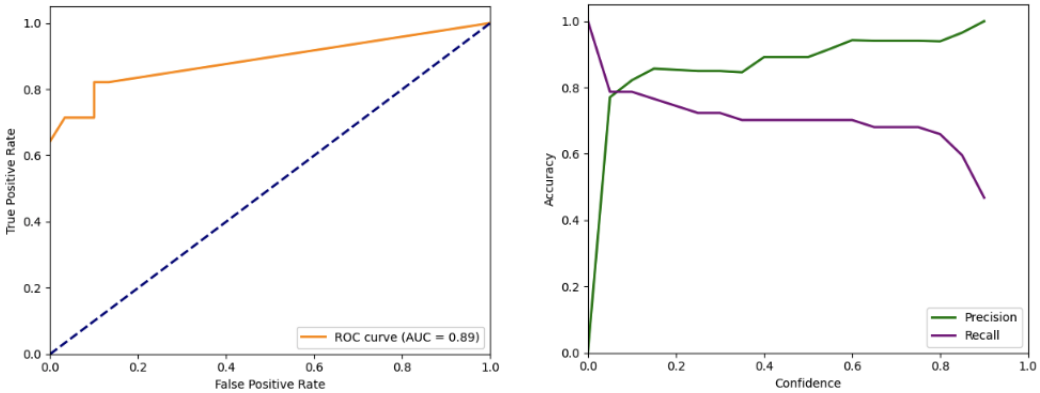
Figure 3.6 shows the results of the YOLOv8 model trained on the small dataset, figure 3.7 presents the model trained on the medium dataset, and figure 3.8 displays the results for the large dataset. All figures focus on detecting Coca-Cola. The three YOLOv8 models exhibit similar performance, with the AUC for the small and large datasets at 0.88 and the medium dataset at 0.89. The AUC values indicate that all models perform relatively well at distinguishing between frames with and without Coca-Cola, with values closer to 1 showing a stronger ability to differentiate. Over the 137 validation images, the small model took an average of 7.1ms for pre-processing, inference, and post-processing on GPU A100. Both the medium and large took 7.2ms. All three model datasets achieve an efficient balance between precision and recall at an estimated 0.8, demonstrating the model's successful performance. The lower confidence threshold used by the large and medium models suggests that the models are more conservative, minimising false positives and ensuring more reliable performance. Although the medium model performs slightly better, the difference is not substantial enough to confidently conclude it is not due to chance. Additional evaluation of more complex datasets and further refinement of model parameters are needed to assess whether this dataset size is optimal.

Figures A.7, A.8, A.9 display the results of the small, medium and large YOLOv5 models for detecting Coca-Cola. As expected, these results are worse than YOLOv8, particularly in the first performance assessment task. Conversely, to YOLOv8, the medium YOLOv5 model appears to outperform the large and small datasets more substantially, particularly in its ability to count the number of objects within each frame. The F1 score for the medium model is just below 0.8 compared to the small and medium models' F1 scores of around 0.7. All three models took an average of 5.3ms per



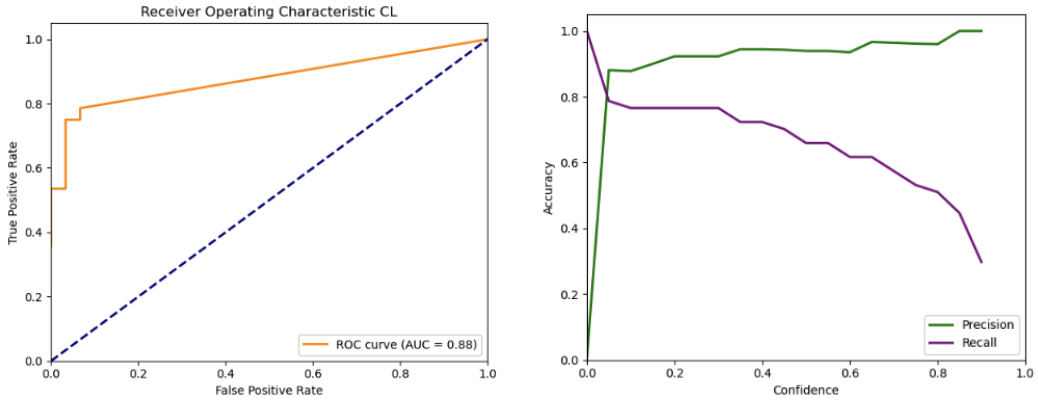
(a) Receiver Operating Statistic (b) Precision and Recall

Figure 3.6: Performance metrics for YOLOv8 Small Dataset for detecting Coca-Cola



(a) Receiver Operating Statistic (b) Precision and Recall

Figure 3.7: Performance metrics for YOLOv8 Medium Dataset for detecting Coca-Cola



(a) Receiver Operating Statistic (b) Precision and Recall

Figure 3.8: Performance metrics for YOLOv8 Large Dataset for detecting Coca-Cola

image. Finally, figures A.10, A.11, A.12 show the results of the small, medium and large RT-DETR models for detecting Coca-Cola. These results show a decrease in performance compared to YOLOv5 and YOLOv8, which was anticipated based on their training outcomes. The medium model seems to have a much better ROC curve; however, the precision and recall curve show poor performance. This discrepancy suggests that the model's ability to distinguish between frames with and without objects of interest may be due to chance. The average speed of these models was 9.1ms per image on GPU A100. The speed of DETR models is much slower than YOLO due to its extensive transformer-based architecture.

The model validation results have been summarised in Table 3.1, which displays the average across all four brands. Both performance assessment tasks appear to show a similar result, despite the first being based on the presence of the brand in a given frame and the second focusing on the actual count of detected objects. The medium model outperforms the large and small models, achieving a higher AUC statistic and F1 scores across all three models. This difference is more pronounced when considering all classes rather than just Coca-Cola, particularly for YOLOv5 and RT-DETR. The improved performance of the medium model suggests that, given the default parameters, the medium-sized dataset was a much better balance of complexity and variety, allowing the model to learn more effectively and achieve better generalisation than small and large datasets. All results indicate a low number of false positives, hence a low optimal threshold of 0.10. The

Model	Size	Av. AUC	Av. Max F1	Av. Speed (ms)
YOLOv8	Small	0.925	0.821	7.1
	Medium	0.948	0.869	7.1
	Large	0.908	0.841	7.2
YOLOv5	Small	0.878	0.717	5.3
	Medium	0.915	0.848	5.3
	Large	0.890	0.788	5.3
RT-DETR	Small	0.893	0.711	9.2
	Medium	0.940	0.770	9.1
	Large	0.820	0.622	9.1

Table 3.1: Summary of Model Validation Results

YOLOv8 model performed the best on both performance assessment tasks for each dataset size.

3.5 Discussion

The validation results highlight the effectiveness of transfer learning of object detection models for Kids Online. All models showed improved performance metrics as they adapted to detect new classes for the Kids Online Project. While the results do not show strong evidence of overfitting, the plateau in validation loss suggests that they have reached their generalisation limit and can no longer improve their performance. Larger datasets took significantly longer to train than smaller datasets; however, the YOLO models trained on the larger datasets reached higher mAP values during training. The RT-DETR model conversely shows a decrease in the mAP as the dataset became more complex, with an mAP of 0.684 for the small model, suggesting that the model may require further parameter optimisation to handle more complex datasets. Compared to the default model parameters, the refined models achieve a higher mAP during pre-training, which is expected because the fine-tuning datasets are much smaller and less complex than the pre-training dataset. Nonetheless, it demonstrates that transfer learning has been successfully applied.

The validation set results are directly comparable to real-world data and enable an objective evaluation of different-sized models under the same con-

ditions, allowing for a more precise assessment of how dataset size impacts performance. The variability in the performance of the small, medium, and large datasets shows that the quality of the datasets significantly impacts the performance of the model. The medium models outperformed the large and small models in both performance tasks on all three occasions. Given the model parameters, these models efficiently trade off complexity and generalisation, achieving a balance that allows for strong detection. The small dataset likely does not provide sufficient information about the distinction between object characteristics, while the large dataset contributes to overfitting. The results were not as expected because the recommendation was to use 1500 images, which suggested that increased dataset size would lead to better performance. To build on this analysis, we would need to test with a much larger quantity of images and optimise more model parameters so that they can better handle the complexity of the dataset. In terms of the model architectures, the YOLOv8 outperformed YOLOv5, demonstrating how performance has improved from earlier YOLO versions. The superior performance of the YOLO models over RT-DETR suggests that the CNN structure is more effective and adaptable for transfer learning, requiring less parameter tuning to achieve strong performance. In contrast, RT-DETR would need further trials and optimisation to match the efficiency and accuracy of YOLOv8, making it a less optimal choice for rapid fine-tuning and deployment. The YOLOv8 model will be used to apply these models to the Kids Online data moving forward. The medium dataset has also been selected as it can most efficiently balance model complexity and generalisation using the default model parameters while maintaining a reasonable training and processing speed.

Chapter 4

Application to Kids Online

4.1 Overview

The Kids Online Project aims to understand the extent of harmful online advertising targeted at children, focusing mainly on the visibility of brands. Quantifying advertisements through exposure counts and duration allows Public Health researchers to examine the impact of advertising on children's health. Object detection models output frame-by-frame results, which are not meaningful as they do not provide exposure intervals and contain duplicates in adjacent frames. To generate meaningful statistics, the model output must align with manual annotations and provide an exposure frequency and duration, giving critical insights into evaluating the extent of harmful advertising in digital media.

4.2 Data Preparation

4.2.1 Pre-processing

Ultralytics [61] provides robust tools for processing video content. However, an analysis of the Kids Online dataset reveals significant redundant footage. Examples include videos in which participants did not share their screen, remained on the same screen for a prolonged duration, or turned off their device for an extended period, resulting in a lengthy blank screen segment. Processing such repeated content would lead to inefficient use of resources. Instead, the torch and CSV libraries split the videos, capturing several frames

every second. Redundant frames have been filtered by removing those with a grayscale value below five or an absolute difference of less than 2000 from the previous frame. These thresholds were deliberately set low to ensure that only redundant frames were filtered out while preserving meaningful content. A frame extraction rate must be chosen to balance resource efficiency and adequate data capture; therefore, various frame rates will be examined. Although these models adjust image quality to match the training data, preliminary experiments indicated that altering the input frame size impacted the consistency of the results throughout the exposure interval. Therefore, three different resolutions will be tested by setting the long side of the frame to 700, 800, and 900 pixels while maintaining the current aspect ratio. This test will determine the optimal input frame quality.

4.2.2 Post-processing

Interval Creation

After filtering, the remaining frames are predicted using the chosen model. The model output includes the timestamp, prediction label, confidence level, and location. With these results, the frame-level predictions can be grouped based on their timestamp and location using the Pandas and Numpy packages. Due to the nature of this content, the distance between predictions is not a significant factor when creating intervals because it is highly likely they will vary significantly, such as when a user is scrolling through content. However, if multiple predictions of the same type are in the frame, they are matched to an interval using the predicted location. If there are numerous predictions with similar locations or multiple possible intervals, the closest one is chosen based on their Euclidean distance. These intervals also depend on a time threshold, which must be carefully balanced. Given a threshold of, say, 2 seconds, if the gap between two of the same predictions is longer than 2 seconds, then this interval is ended, and a new one begins. If the threshold is too short, there is a risk that the model will provide a false negative for a short duration, and then a new interval will be created, even though the object remains on screen. On the other hand, if the threshold is too long, it could result in the object being mistakenly counted as the same exposure despite the object moving off-screen and then re-entering. The trade-off here is between the possibility of missing detections and incorrectly counting re-entries as the same exposure, making finding an optimal threshold to

minimise both errors essential. Three different thresholds will be tested: 1 second, 2 seconds and 3 seconds. To ensure that each interval is processed by the model, at least one image within these thresholds needs to be taken, regardless of its absolute difference from the previous frame. The frame at the end of each interval must also be captured to ensure the entire time interval is taken. These adjustments are made to the pre-processing stage.

Interval Overlap

Manual annotations classify each distinct object as a single exposure; however, since the model currently detects the product and its logo, a method to handle overlapping bounding boxes is required to ensure each object is identified as a single exposure. One possible solution involved filtering overlaps at the frame level before grouping them into intervals. At each frame prediction, overlapping boxes were identified using the bounding box coordinates and the prediction with the smaller confidence score was removed. However, this approach introduced several challenges. If the confidence score of an object fluctuated throughout the interval, then two overlapping bounding boxes could switch between having the higher confidence, which could allow both intervals to be created despite them overlapping. Additionally, removing lower-confidence predictions inflated the average confidence of incorrect exposures, further reducing accuracy. To overcome these challenges, overlaps were filtered after intervals were generated. This approach requires an aggregated location for the intervals, so the median has been selected as the most suitable metric. Intervals with overlapping timestamps and median bounding box locations were identified, and the interval with the lower average confidence was removed. If predictions were from the same brand, the intervals were merged, retaining the higher confidence. Although some overlapping boxes persisted due to differing medians, most were resolved, resulting in more accurate exposure counts.

Final Output

Once the exposure intervals are established, each interval's average confidence score and start and end times are generated. These exposures are then classified as true positive, false positive and false negative. Similarly to the model development phase, an infinite number of true negative intervals makes this an unsuitable metric. The categorisation of these intervals is based on

the ground truth data obtained through manual annotation, following the same procedures used for the Kids Online dataset. Instances of re-exposures to the same object were excluded from the analysis. The code for these methods can be found at https://github.com/hawkeg02/SpotTheAd.git.

4.3 Performance Analysis and Optimisation

4.3.1 Image Resolution

Precision and recall metrics were computed based on the post-processing results to assess the impact of different image resolutions. After resizing and filtering during pre-processing, the remaining frames were predicted using the YOLOv8 medium model at a confidence threshold of 0.2. This threshold was set slightly above the average optimal threshold identified during model validation, as a higher number of false positives is expected in a dataset with a more significant proportion of null occurrences. A range of thresholds for the average interval confidence was selected from 0.2 to 0.9. Figure 4.1 displays the result of a frame input size of 800 pixels. This graph shows

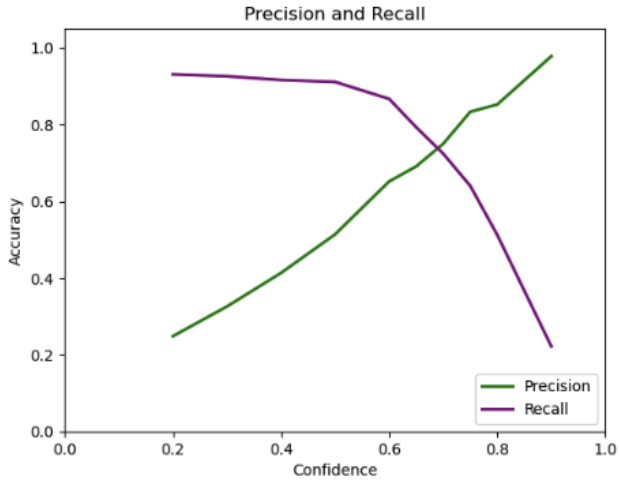


Figure 4.1: Performance metrics for an image resolution size of 800

the trade-off between precision and recall. At a lower threshold, the model tends to classify more intervals as positive, leading to higher precision in its ability to identify positive classes but lower recall, indicating a high number

of false positives. A higher threshold, on the other hand, will be much more conservative in its predictions, leading to lower false positives and higher recall but a lower precision as it is less likely to identify the positive cases. The F1 score balances this trade-off. The optimal F1 score for this trial is 0.744 at a confidence of 0.6. Figures A.13 and A.14 display the results of the trials for 700 pixels and 900 pixels. An image resolution of 900 has a noticeable drop off in performance compared to 700 and 800. An image resolution of 700 achieved an optimal F1 score of 0.736 at a confidence of 0.65, slightly worse than 800 pixels.

4.3.2 Frame Rate

A noticeable aspect of these results is that many more false positives exist, and a significant portion of these exposures had only a single frame within the interval. After removing all intervals with only a single frame at an image resolution of 800, the results showed an improved F1 score of 0.793 at a confidence of 0.60, as shown in Figure 4.2. This model lost six true positives but eliminated significantly more false positives, increasing the overall F1 score.

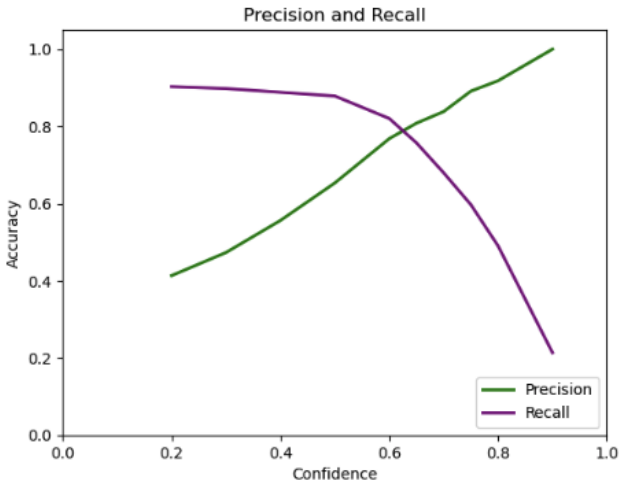


Figure 4.2: Performance metrics for an image resolution size of 800 and removing single framed intervals

The previous image resolution tests used a frame extraction rate of 4fps. However, it is also necessary to compare the performance at 5fps to determine

the optimal frame rate, which trades off accuracy and speed. The image input size has been set to 800 pixels, and single-frame intervals have been removed, as determined by earlier tests. A frame rate of 5fps increased the number of true positives as expected. However, it also increased the false positives and resulted in a slightly decreased F1 score of 0.781 at a confidence of 0.65 compared to an F1 score of 0.793 at 4fps. Hence, a frame rate of 4 fps has been selected to balance resource consumption and accuracy more efficiently. A smaller frame rate was not tested because the reduced resource use does not justify the potential loss of crucial details in the fast-paced Kids Online data.

4.3.3 Interval Length

To identify the optimal interval length, thresholds of 1 second, 2 seconds, and 3 seconds were evaluated in post-processing. The previously selected conditions have remained constant to test these three different thresholds. For a 3-second interval, there were 40 false positives, 33 false negatives, and an overall F1 score of 0.775. Compared to manual annotations, seven exposures were merged with other intervals, while two intervals were split into multiple exposures. At a 2-second time threshold, there were 42 false positives, 31 false negatives, and an F1 score of 0.782, with five merged and six split exposures. A 1-second time threshold resulted in 44 false positives, 29 false negatives, and an F1 score of 0.787, with seven merged and three split exposures. The 1-second interval achieved the highest F1 score by capturing more frames, however, it also increased the number of false positives. Given that the priority for the Kids Online Project is to minimise false positives rather than false negatives, the 2-second interval was selected as it achieves a better balance of false positives and false negatives and has a more optimal balance between merging and splitting exposures, resulting in more reliable exposure counts.

4.3.4 Final Results

To further reduce the high false positive rate, the confidence threshold was increased to 0.62. Intervals with a confidence score below 0.7 and with three or fewer frames were also removed. Filtering these intervals resulted in 30 false positives, 32 false negatives, a precision of 0.81, a recall of 0.8, and an overall F1 score of 0.81.

The model is also evaluated on its accuracy in measuring exposure interval duration. The ground truths were used to calculate the accuracy of exposure counts; however, significant discrepancies have been observed between the ground-truth intervals and the model's intervals due to differing time precisions and the handling of partial object visibility within intervals. As a result, the ground truths have not been used. Instead, the previous exposure count results have been used to categorise intervals into true positive, false positive, and false negative time intervals. The total object on-screen time recorded was 10.08 minutes, with 8.34 minutes identified as true positive occurrences and 1.43 minutes classified as false negative intervals. These results correspond to a precision of 0.85, a recall of 0.83, and an F1 score of 0.84.

4.4 Discussion

These results show that the model's output has been effectively aligned with the manual annotations. The model can detect objects of interest and contribute meaningful information to the Kids Online project. An interesting discovery from this analysis was the strong dependence of the results on image quality. Despite the model resizing the image before each prediction, different image input sizes drastically changed the model's performance, which suggests that the original image quality and resolution still play a significant role in detection accuracy. Lower-quality images may lose critical details during resizing, while higher-quality images are too dissimilar from the training images, so the model struggles to generalise. Another limitation of this analysis emerged during post-processing. Due to the dynamic nature of the data, creating accurate intervals remains a challenge, and the results will always differentiate from manual annotations, particularly in how they handle partial visibility and quick re-entries. Manual annotations often include intervals where less than 50% of an object is visible. For instance, in Figure 4.3, a black can with a visible Pepsi logo became obscured and reappeared later. While manual annotations treat this as a single interval, the model splits it into two exposures, as it cannot detect the object when only a small portion is visible. In another example, a Sprite bottle was visible in one clip, followed by an immediate cut to another featuring the same bottle without any null occurrences. The model counted this as a single exposure, whereas manual annotations recorded it as two separate exposures due to the change in clips.

Re-entries of exposures displayed the same behaviour or contained only one frame, so they were later removed.

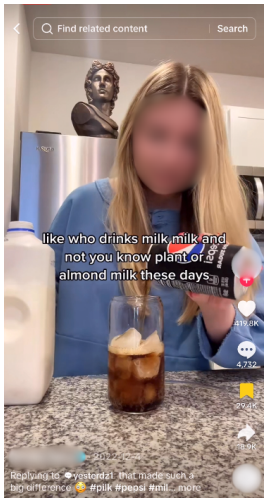


Figure 4.3: Validation set image displaying a product where less than 50% of the logo is visible

Other notable differences between the model's results and manual annotations are due to the object detection accuracy. For example, the model struggled with detecting small objects that are common in real-life data, as illustrated in Figure 4.4, where the model failed to detect the Coca-Cola logo. As humans, we can interpret the context of the image more effectively; for example, even though the Coca-Cola logo is small and almost unreadable, it is evident that the cup is from a fast-food restaurant and based on our knowledge, we can infer that the red circle with white linked writing on the cup is a Coca-Cola logo. This limitation highlights the challenge that all object detection models face in recognising visually diminished or obscured objects despite contextual clues that humans can quickly identify. Additionally, the model often detected objects with high similarity, as shown in Figure 4.5, where an object was incorrectly identified as a 'Coca-Cola Can'. The training process identified that the model bordered on overfitting; however, the inability to differentiate this object from Coca-Cola suggests that the model may require more training and diverse data to inform the model better. Other model parameters must be optimised to handle a more complex dataset and prevent overfitting. Additionally, as technology advances, the accuracy of the models will also improve. Despite these limitations, the model occasionally

Figure 4.4: Validation set image displaying a Coca-Cola Logo that is too small for the model to detect

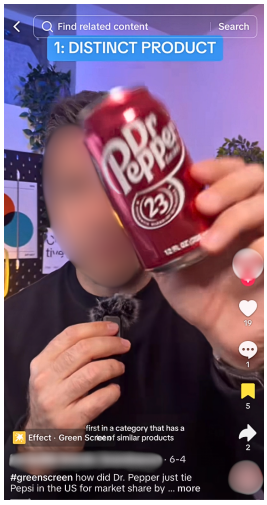


Figure 4.5: Validation set image displaying a product which was incorrectly classified as Coca-Cola

detected objects missing from the manual annotations, highlighting their potential strengths. While manual annotations took 9 hours to complete, the model, taking advantage of parallel processing, took only 40 minutes, further underscoring the significance of this approach, especially as the scale of the data becomes much larger. Overall, the object detection results were successfully transformed into meaningful statistical insights for the Kids Online Project. Frame-level predictions have been filtered and combined into intervals to accurately represent exposures, providing count and total duration and robust evidence about brand exposures in children's media.

4.5 Real-World Application

The manual annotations for the Kids Online project are incomplete; however, to understand how this information can inform the Kids Online project, the model was run on 24 students with a total of 306 videos. In addition to the four soft drink brands included in this analysis, three fast food brands, McDonald's, Burger King, and KFC, have been included in the custom dataset, following the same process outlined in this paper for consistency. The validation set on which the data was tested, although similar, does differ from the Kids Online data in regards to the frequency of exposures and range of activities performed on the device. The validation set consisted of scrolls through social media, whereas participants in this study are likely doing a much broader range of things online. As there are no ground truths, we are unable to find the optimal confidence threshold to balance precision and recall for this data. Since we expect a much lower frequency of exposures and a higher false positive, the confidence threshold is increased to 0.8 to account for this. The model observed 166 exposures with a total duration of 11.48 minutes. Although we cannot validate this number, it is higher than what we would expect for only a sample size of 24 students. Figure 4.6 displays 2 example exposures from this Kids Online Data.

The output provides Public Health researchers with critical insights into the extent of harmful advertising in children's media. Figure 4.7 displays the exposure count and duration by device. Phones exhibited a significantly higher rate of exposure to predefined brands compared to laptops and desktops. This aligns with expectations because children are more likely to engage with social media on their phones, increasing their likelihood of exposure to advertising content. In contrast, laptop/desktop usage is more commonly as-

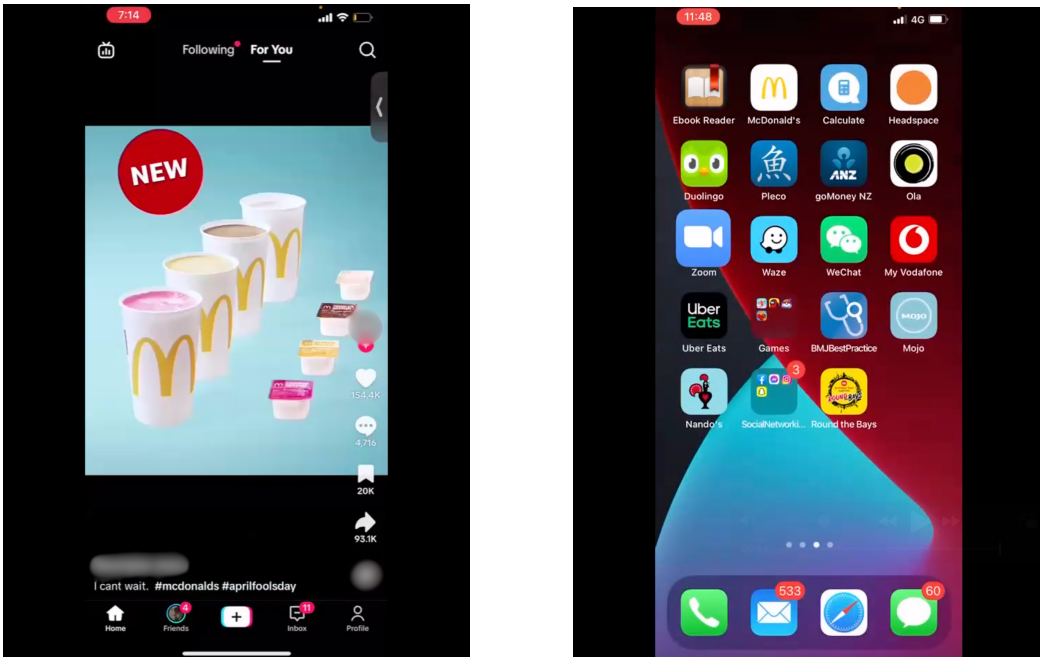
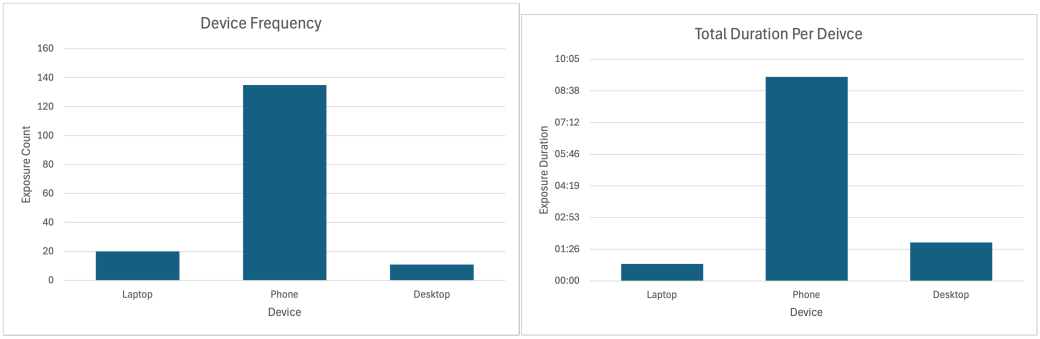


Figure 4.6: Example exposures from Kids Online Data

sociated with gaming, which typically features fewer advertisements. Figure 4.8 displays the exposure count by hour for each day. Saturday and Sunday show a more consistent pattern throughout the day, while Thursday and Friday peak during out-of-school hours. Interestingly, Thursday and Friday saw higher exposure rates compared to the weekend days.



(a) Total Exposure Count by Device (b) Total Duration by Device

Figure 4.7: Exposure Count and Duration by Device

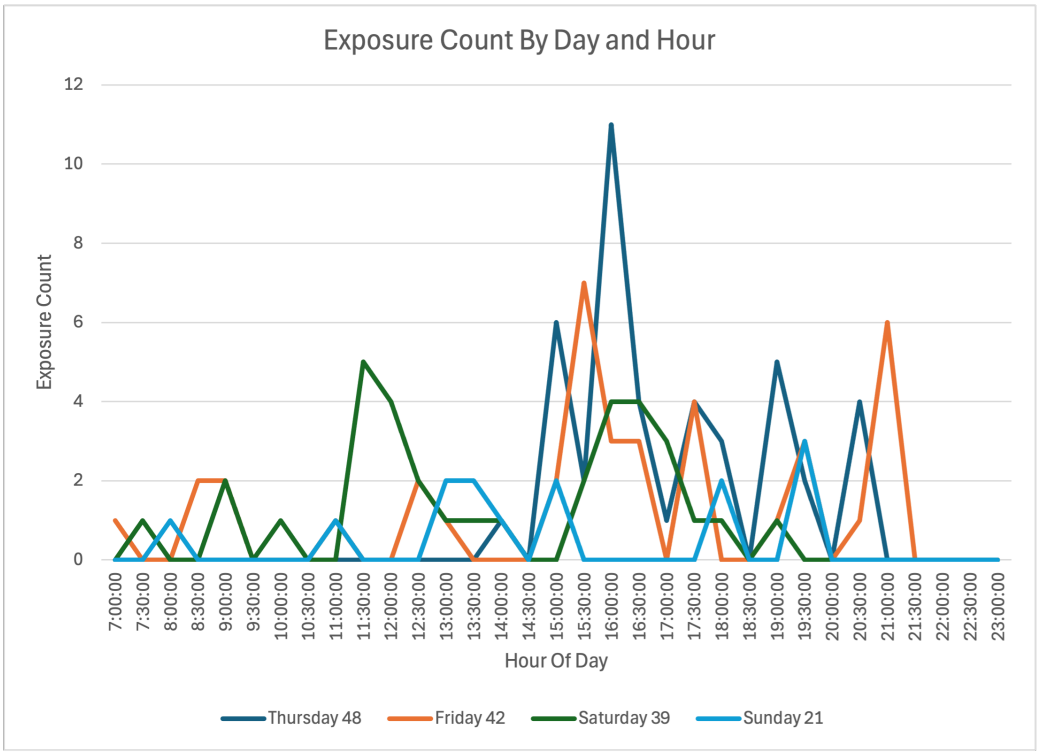


Figure 4.8: Total Exposure Count by Hour and Day

Chapter 5

Conclusion

The strong causal links between the advertising of unhealthy commodities and adverse health outcomes in children and adolescents make it essential to quantify the prevalence of such advertising that children are exposed to online. Previous work in this area has often focused on traditional means of media or has utilised self-reports and experiments in controlled settings. The Kids Online Project, however, enables a more accurate analysis of children's real-life online experiences. An automated approach to detect harmful advertising within this content is required to process this data more efficiently. The literature review revealed two distinct model architectures that have seen huge improvements and promising results regarding accuracy and speed. A YOLOv8 model trained on a dataset containing 80–90 instances per class proved to be the most efficient at distinguishing between frames with and without the object of interest and accurately estimating the number of exposures within each frame. This analysis had limitations, particularly regarding the size of the datasets analysed and limited parameter optimisation. However, a key objective for the Kids Online Project is to ensure time efficiency in model development and the YOLOv8 has performed well on the default parameters, fitting these constraints. This analysis has highlighted the advantages of transfer learning by detecting various new objects of interest without needing to retrain the model from scratch, further cutting down on time costs. The YOLOv8 model was also tested on a validation set similar to the Kids Online Data. This further proved the model's ability to perform efficiently on real-life applications, with a high accuracy and quick processing time. The pre-processing and post-processing methods demonstrated how

an object detection model can efficiently process videos and aggregate the results into intervals to provide valuable insights for the public health sector. This information provides researchers with a clear understanding of the extent of harmful advertising in the Kid's Online data.

Despite the success of the application of these models, several key limitations emerged regarding object detection as a whole. One of the key challenges is how heavily object detection performance relies on the dataset's quality. A custom dataset must be large enough to contain sufficient information about the visual aspects of each object. However, creating a custom dataset is labour-intensive due to images needing to be collected and annotated. Handling this amount of data without overfitting the model also requires more parameter optimisation, which is also a time-consuming process. In the case of Kids Online, the classification of "harmful" brands requires a predefined list. In our increasingly commercialised society, the number of unhealthy commodities is consistently growing, so maintaining an up-to-date and comprehensive dataset is extremely difficult. Additionally, unhealthy food commodities are not always advertised with a brand logo present, especially in UGC. Without the distinctive logo, the object detection model cannot recognise and quantify these exposures, further limiting the analysis. Object detection models cannot fully understand the contextual information humans can interpret, limiting their capabilities. While the models demonstrated promising results, there is still considerable room for improvement.

5.1 Future Work

As object detection models evolve alongside technological advancements, the potential avenues for future research will also change and grow. Since the beginning of this project, multiple new versions of the YOLO architecture, such as YOLOv9, YOLOv10, and YOLOv11, have been released, each demonstrating improvements in speed and accuracy. The improved architecture of these new versions has the potential to enhance the accuracy of detecting harmful advertising exposures in the Kids Online data. As previously discussed, this analysis could also be enhanced through changes and improvements in the dataset quality and optimising the model parameters. Techniques such as automated annotation techniques or something similar to 'weak labels' introduced by ABIDLA [51] would speed up the custom dataset creation and allow it to increase in size. The annotation technique

itself also presents an opportunity for improvement. Currently, the custom dataset uses a simple bounding box. However, another way would be to use polygon annotation, which precisely traces around the object, providing the model with an understanding of each object's visual features. This technique is more time-consuming but provides much more accurate results. Finally, integrating a text-based model could also aid in brand recognition. Many brands display their names without the logo, so a model such as an Optical Character Recognition (OCR) model could transform image-text to text, increasing brand detection accuracy. OCR models can also help beyond brand recognition and identify other associations to unhealthy commodities that children are exposed to online.

Bibliography

[1] *Advertising Standards Authority*. URL: https://www.asa.co.nz/codes/codes/children-and-young-people/.

[2] Marcus Gurtner et al. "Objective assessment of the nature and extent of children's internet-based world: protocol for the kids online Aotearoa study". In: *JMIR Research Protocols* 11.10 (2022), e39017.

[3] Zhong-Qiu Zhao et al. "Object detection with deep learning: A review". In: *IEEE transactions on neural networks and learning systems* 30.11 (2019), pp. 3212–3232.

[4] Darren Powell. "Harmful marketing to children". In: *The Lancet* 396.10264 (2020), pp. 1734–1735.

[5] World Health Organization et al. *Report of the commission on ending childhood obesity*. World Health Organization, 2016.

[6] Bridget Kelly et al. "Television food advertising to children: the extent and nature of exposure". In: *Public Health Nutrition* 10.11 (2007), pp. 1234–1240.

[7] G Hastings. "Review of research on the effects of food promotion to children". In: *University of Strathclyde* (2003).

[8] Gerard Hastings et al. "The extent, nature and effects of food promotion to children: a review of the evidence". In: *Geneva: World Health Organization* 20 (2006).

[9] Georgina Cairns et al. "Systematic reviews of the evidence on the nature, extent and effects of food marketing to children. A retrospective summary". In: *Appetite* 62 (2013), pp. 209–215.

[10] Alex Molnar et al. “Schoolhouse Commercialism Leaves Policymakers Behind–The Sixteenth Annual Report on Schoolhouse Commercializing Trends: 2012-2013.” In: *National Education Policy Center* (2014).

[11] Kathryn C Montgomery and Jeff Chester. “Interactive food and beverage marketing: targeting adolescents in the digital age”. In: *Journal of adolescent health* 45.3 (2009), S18–S29.

[12] Juan Francisco Dávila and Mònica Casabayó. “Influences in children’s materialism: a conceptual framework”. In: *Young Consumers* 14.4 (2013), pp. 297–311.

[13] Agnes Nairn, Jo Ormrod, and Paul Andrew Bottomley. “Watching, wanting and wellbeing: Exploring the links-a study of 9 to 13 year-olds”. In: (2007).

[14] Helen Clark et al. “A future for the world’s children? A WHO–UNICEF–Lancet Commission”. In: *The Lancet* 395.10224 (2020), pp. 605–658.

[15] Helena Sandberg, Kerstin Gidlöf, and Nils Holmberg. “Children’s exposure to and perceptions of online advertising”. In: *International Journal of Communication* 5 (2011), p. 30.

[16] Reg Bailey. “Letting children be children: Report of an independent review of the commercialisation and sexualisation of childhood”. In: (2011).

[17] Louise N Signal et al. “Children’s everyday exposure to food marketing: an objective analysis using wearable cameras”. In: *International Journal of Behavioral Nutrition and Physical Activity* 14 (2017), pp. 1–11.

[18] Leah Watkins et al. “An objective assessment of children’s exposure to brand marketing in New Zealand (Kids’ Cam): a cross-sectional study”. In: *The Lancet Planetary Health* 6.2 (2022), e132–e138.

[19] Mariya Stoilova, Sonia Livingstone, Rana Khazbak, et al. “Investigating Risks and Opportunities for Children in a Digital World: A rapid review of the evidence on children’s internet use and outcomes”. In: (2021).

[20] Stefanie Vandevijvere et al. “Unhealthy food marketing to New Zealand children and adolescents through the internet”. In: (2017).

[21] Yali Amit, Pedro Felzenszwalb, and Ross Girshick. “Object detection”. In: *Computer Vision: A Reference Guide*. Springer, 2021, pp. 875–883.

[22] Zhengxia Zou et al. “Object detection in 20 years: A survey”. In: *Proceedings of the IEEE* 111.3 (2023), pp. 257–276.

[23] Tausif Diwan, G Anirudh, and Jitendra V Tembhurne. “Object detection using YOLO: Challenges, architectural successors, datasets and applications”. In: *multimedia Tools and Applications* 82.6 (2023), pp. 9243–9275.

[24] Rafael Padilla, Sergio L Netto, and Eduardo AB Da Silva. “A survey on performance metrics for object-detection algorithms”. In: *2020 international conference on systems, signals and image processing (IWSSIP)*. IEEE. 2020, pp. 237–242.

[25] Mark Everingham et al. “The pascal visual object classes (voc) challenge”. In: *International journal of computer vision* 88 (2010), pp. 303–338.

[26] Tsung-Yi Lin et al. “Microsoft coco: Common objects in context”. In: *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*. Springer. 2014, pp. 740–755.

[27] Tsung-Yi et al. *COCO 2017: Common Objects in Context 2017*. 2017. URL: `https://cocodataset.org/#home`.

[28] Yang Liu et al. “A survey and performance evaluation of deep learning methods for small object detection”. In: *Expert Systems with Applications* 172 (2021), p. 114602.

[29] Jun Deng et al. “A review of research on object detection based on deep learning”. In: *Journal of Physics: Conference Series*. Vol. 1684. 1. IOP Publishing. 2020, p. 012028.

[30] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. “Imagenet classification with deep convolutional neural networks”. In: *Advances in neural information processing systems* 25 (2012).

[31] Ross Girshick et al. “Rich feature hierarchies for accurate object detection and semantic segmentation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014, pp. 580–587.

[32] Kaiming He et al. “Spatial pyramid pooling in deep convolutional networks for visual recognition”. In: *IEEE transactions on pattern analysis and machine intelligence* 37.9 (2015), pp. 1904–1916.

[33] Ross Girshick. "Fast r-cnn". In: *arXiv preprint arXiv:1504.08083* (2015).

[34] Tsung-Yi Lin et al. "Feature pyramid networks for object detection". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 2117–2125.

[35] J Redmon. "You only look once: Unified, real-time object detection". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.

[36] Peiyuan Jiang et al. "A Review of Yolo algorithm developments". In: *Procedia computer science* 199 (2022), pp. 1066–1073.

[37] Joseph Redmon and Ali Farhadi. "YOLO9000: better, faster, stronger". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 7263–7271.

[38] Juan Terven, Diana-Margarita Córdova-Esparza, and Julio-Alejandro Romero-González. "A comprehensive review of yolo architectures in computer vision: From yolov1 to yolov8 and yolo-nas". In: *Machine Learning and Knowledge Extraction* 5.4 (2023), pp. 1680–1716.

[39] Joseph Redmon. "Yolov3: An incremental improvement". In: *arXiv preprint arXiv:1804.02767* (2018).

[40] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. "Yolov4: Optimal speed and accuracy of object detection". In: *arXiv preprint arXiv:2004.10934* (2020).

[41] Glenn Jocher et al. *YOLOv5 by Ultralytics (Version 7.0)[Computer software]*. 2020.

[42] Muhammad Hussain. "YOLO-v1 to YOLO-v8, the rise of YOLO and its complementary nature toward digital manufacturing and industrial defect detection". In: *Machines* 11.7 (2023), p. 677.

[43] Mohammed Gamal Ragab et al. "A Comprehensive Systematic Review of YOLO for Medical Object Detection (2018 to 2023)". In: *IEEE Access* (2024).

[44] Chuyi Li et al. "YOLOv6: A single-stage object detection framework for industrial applications". In: *arXiv preprint arXiv:2209.02976* (2022).

[45] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2023, pp. 7464–7475.

[46] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. "YOLO by Ultralytics". In: (2023).

[47] Dillon Reis et al. "Real-time flying object detection with YOLOv8". In: *arXiv preprint arXiv:2305.09972* (2023).

[48] Yian Zhao et al. "Detrs beat yolos on real-time object detection". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024, pp. 16965–16974.

[49] Yong Li et al. "Transformer for object detection: Review and benchmark". In: *Engineering Applications of Artificial Intelligence* 126 (2023), p. 107021.

[50] Emmanuel Kuntsche et al. "How much are we exposed to alcohol in electronic media? Development of the Alcoholic Beverage Identification Deep Learning Algorithm (ABIDLA)". In: *Drug and alcohol dependence* 208 (2020), p. 107841.

[51] Abraham Albert Bonela et al. "Development and validation of the Alcoholic Beverage Identification Deep Learning Algorithm version 2 for quantifying alcohol exposure in electronic images". In: *Alcoholism: Clinical and Experimental Research* 46.10 (2022), pp. 1837–1845.

[52] Gregory Palmer et al. "A deep learning approach to identify unhealthy advertisements in street view images". In: *Scientific reports* 11.1 (2021), p. 4884.

[53] M Vickya Ramadhan et al. "Comparative analysis of deep learning models for detecting face mask". In: *Procedia Computer Science* 216 (2023), pp. 48–56.

[54] Murhaf Hossari et al. "ADNet: A deep network for detecting adverts". In: *arXiv preprint arXiv:1811.04115* (2018).

[55] Eungyeom Ha, Heemook Kim, and Dongbin Na. "HOD: New Harmful Object Detection Benchmarks for Robust Surveillance". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2024, pp. 183–192.

[56] Wanli Ouyang et al. “Factors in finetuning deep model for object detection with long-tail distribution”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 864–873.

[57] Glenn Jocher. *Ultralytics YOLOv5*. Version 7.0. 2020. DOI: 10.5281/zenodo.3908559. URL: https://github.com/ultralytics/yolov5.

[58] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. *Ultralytics YOLOv8*. Version 8.0.0. 2023. URL: https://github.com/ultralytics/ultralytics.

[59] Yian Zhao et al. *DETRs Beat YOLOs on Real-time Object Detection*. 2023. arXiv: 2304.08069 [cs.CV].

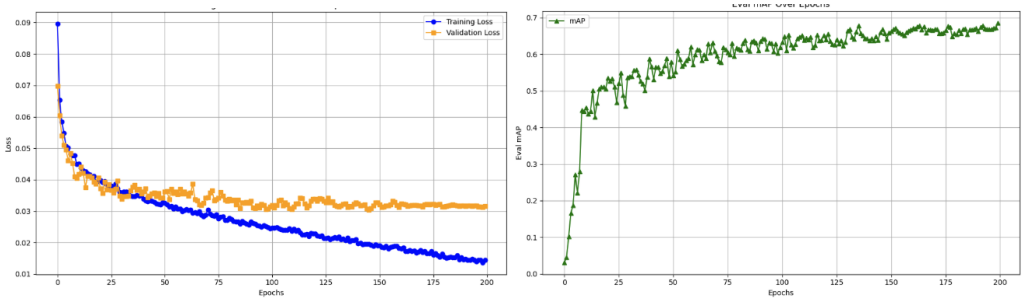
[60] Umair Bashir. *Soft Drinks Brand Awareness KPI ranking UK 2024*. Nov. 2024. URL: https://www.statista.com/statistics/1346279/most-well-known-soft-drink-brands-in-the-uk/.

[61] *Ultralytics*. URL: https://www.ultralytics.com/.

[62] *Roboflow*. URL: https://roboflow.com/.

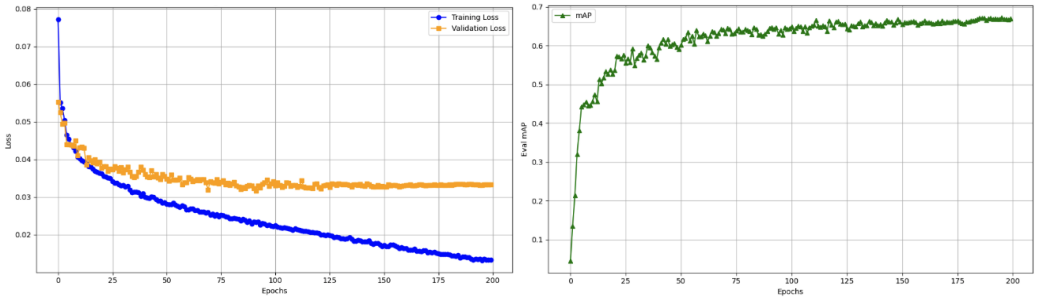
Appendix A

A.1 Appendix For Chapter 3, Subsection 3.3 - Model Development



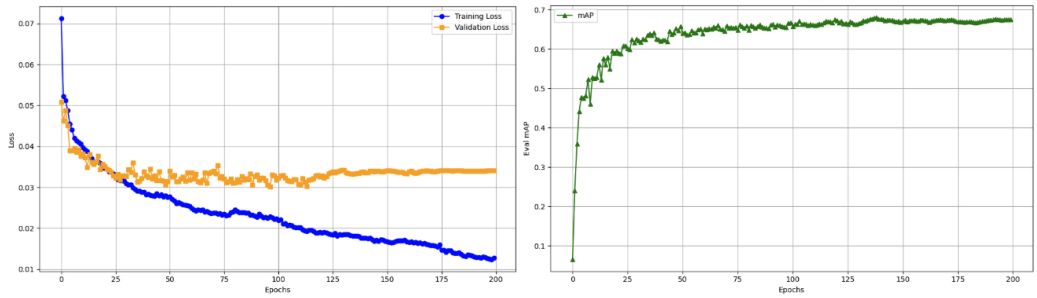
(a) Training and Validation Loss (b) mAP50-90

Figure A.1: Training results of YOLOv5 using the Small Dataset



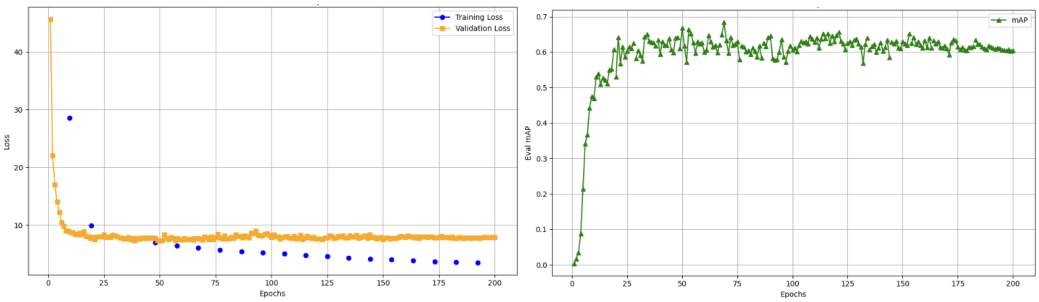
(a) Training and Validation Loss (b) mAP50-90

Figure A.2: Training results of YOLOv5 using the Medium Dataset



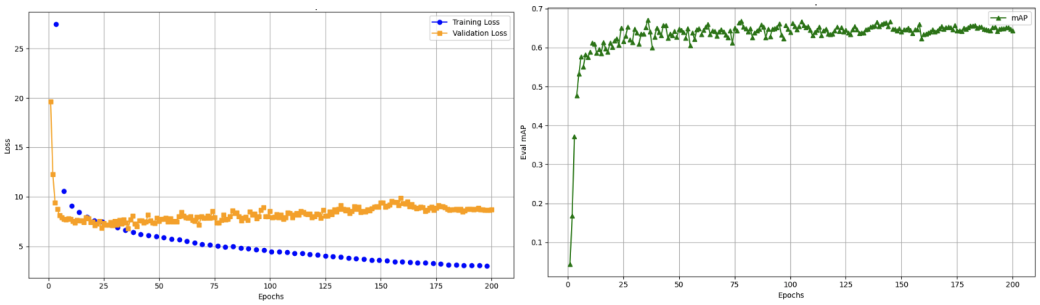
(a) Training and Validation Loss (b) mAP50-90

Figure A.3: Training results of YOLOv5 using the Large Dataset



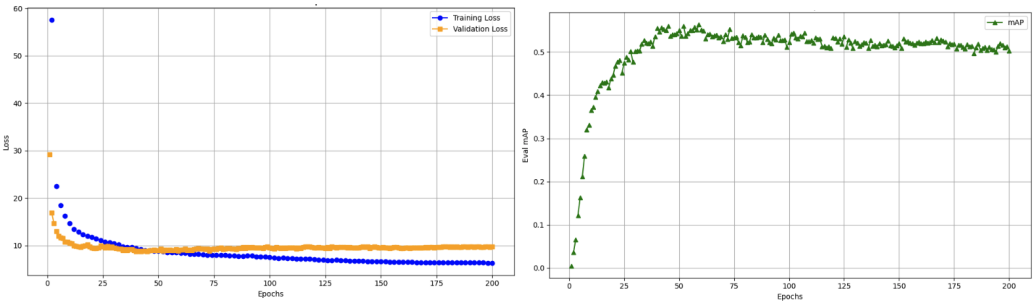
(a) Training and Validation Loss (b) mAP50-90

Figure A.4: Training results of RT-DETR using the Small Dataset



(a) Training and Validation Loss (b) mAP50-90

Figure A.5: Training results of RT-DETR using the Medium Dataset



(a) Training and Validation Loss (b) mAP50-90

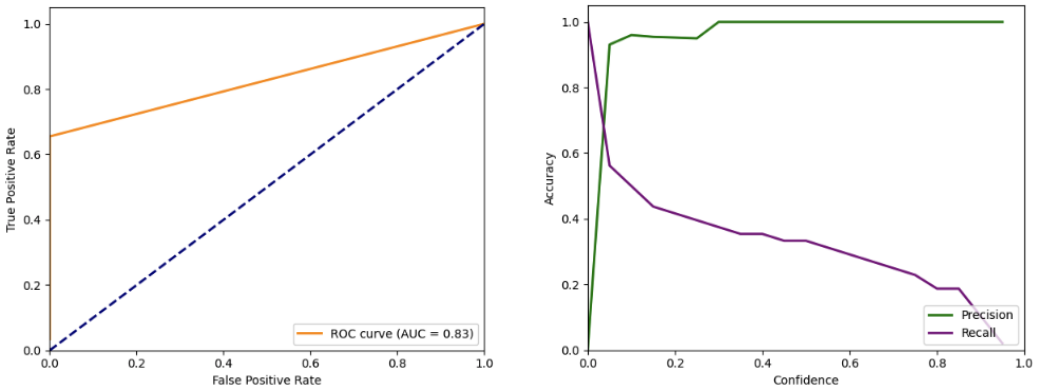
Figure A.6: Training results of RT-DETR using the Large Dataset

A.2 Appendix For Chapter 3, Subsection 3.4 - Model Validation

Model	Size	Brand	AUC	Min EER Threshold	Max F1
YOLOv5	Small	Coca-Cola	0.83	0.05	0.701
		Sprite	0.83	0.05	0.667
		Pepsi	0.95	0.05	0.759
		Fanta	0.90	0.05	0.742
	Medium	Coca-Cola	0.83	0.05	0.783
		Sprite	0.96	0.20	0.860
		Pepsi	0.95	0.05	0.912
		Fanta	0.92	0.05	0.835
	Large	Coca-Cola	0.84	0.05	0.750
		Sprite	0.87	0.05	0.810
		Pepsi	0.91	0.05	0.782
		Fanta	0.94	0.05	0.810
YOLOv8	Small	Coca-Cola	0.88	0.35	0.805
		Sprite	0.94	0.15	0.833
		Pepsi	0.94	0.05	0.815
		Fanta	0.94	0.05	0.833
	Medium	Coca-Cola	0.89	0.05	0.809
		Sprite	0.96	0.20	0.860
		Pepsi	0.98	0.05	0.934
		Fanta	0.96	0.05	0.874
	Large	Coca-Cola	0.88	0.05	0.837
		Sprite	0.92	0.05	0.864
		Pepsi	0.93	0.05	0.819
		Fanta	0.90	0.05	0.844
RT-DETR	Small	Coca-Cola	0.77	0.10	0.605
		Sprite	0.98	0.15	0.835
		Pepsi	0.89	0.30	0.661
		Fanta	0.93	0.10	0.744
	Medium	Coca-Cola	0.87	0.25	0.667
		Sprite	0.98	0.10	0.860
		Pepsi	0.93	0.25	0.707
		Fanta	0.98	0.20	0.844
	Large	Coca-Cola	0.67	0.15	0.487

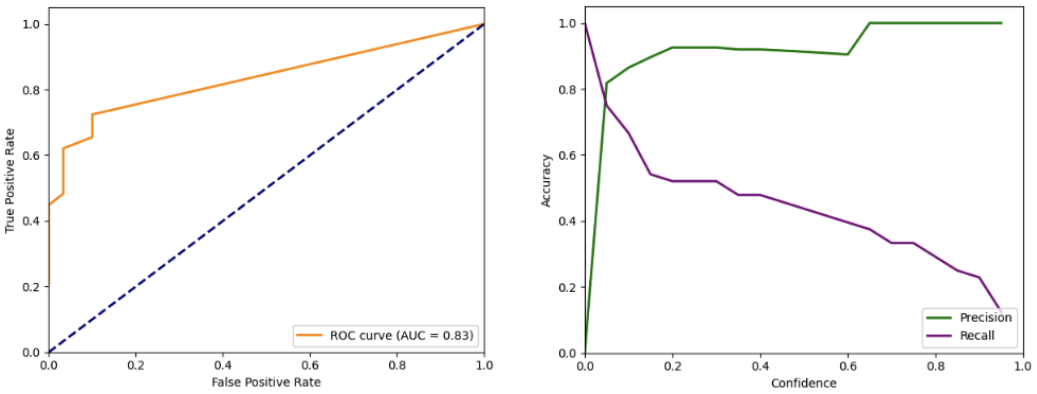
Model	Size	Brand	AUC	Min EER Threshold	Max F1
		Sprite	0.87	0.15	0.649
		Pepsi	0.82	0.15	0.569
		Fanta	0.92	0.10	0.783

Table A.1: Raw Model Validation Results



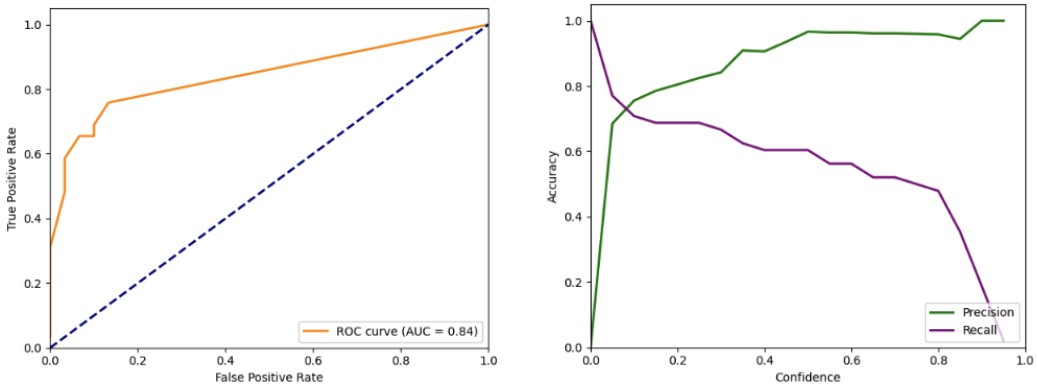
(a) Receiver Operating Statistic (b) Precision and Recall

Figure A.7: Performance metrics for YOLOv5 Small Dataset for detecting Coca-Cola



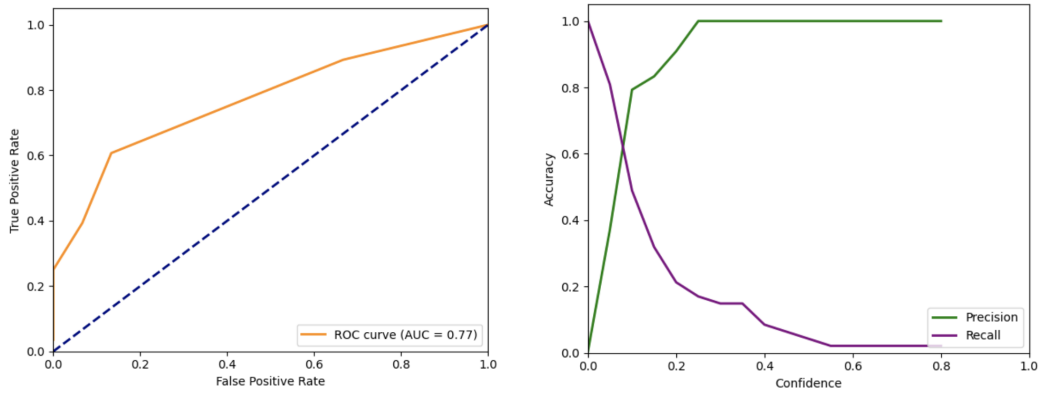
(a) Receiver Operating Statistic (b) Precision and Recall

Figure A.8: Performance metrics for YOLOv5 Medium Dataset for detecting Coca-Cola



(a) Receiver Operating Statistic (b) Precision and Recall

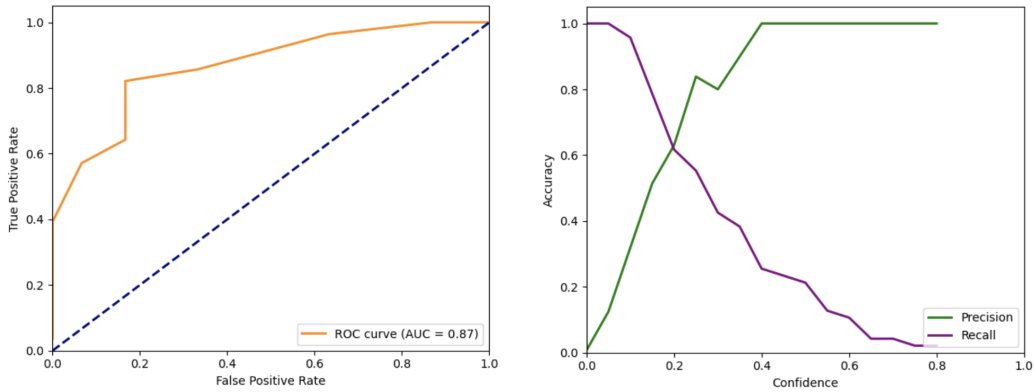
Figure A.9: Performance metrics for YOLOv5 Large Dataset for detecting Coca-Cola



(a) Receiver Operating Statistic (b) Precision and Recall

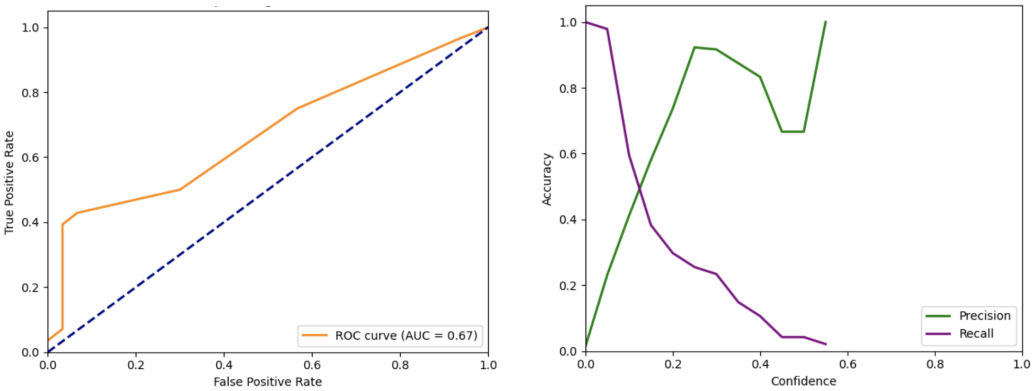
Figure A.10: Performance metrics for RT-DETR Small Dataset for detecting Coca-Cola

A.3 Appendix For Chapter 4, Subsection 4.3 - Performance Analysis and Optimisation



(a) Receiver Operating Statistic (b) Precision and Recall

Figure A.11: Performance metrics for RT-DETR Medium Dataset for detecting Coca-Cola



(a) Receiver Operating Statistic (b) Precision and Recall

Figure A.12: Performance metrics for RT-DETR Large Dataset for detecting Coca-Cola

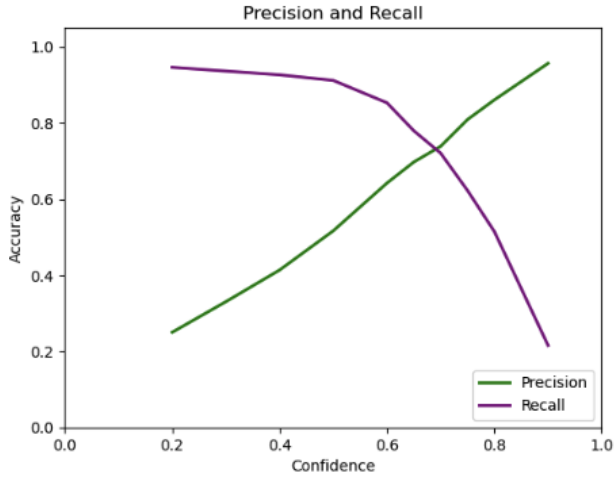


Figure A.13: Performance metrics for an image resolution size of 700

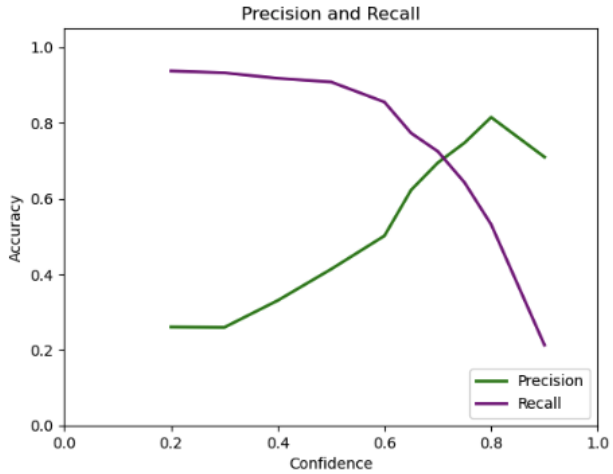


Figure A.14: Performance metrics for an image resolution size of 900