Efficient and Adversarially Robust Object Detection

Anton Liu
Dept. of Computer Science, Western University
aliu467@uwo.ca

Abstract

The abstract goes here.

I. Introduction

Object detection, one of the fundamental tasks in computer vision, is used in numerous real-world applications by enabling machines to identify and locate objects within images or videos. Object detection algorithms aim to address the two challenges of recognizing objects of interest within complex scenes by identifying their spatial extent with bounding boxes, and assigning corresponding class labels. This capability is crucial for a wide range of tasks, including pedestrian detection for autonomous driving, object tracking in video surveillance, and product recognition for inventory management. Moreover, object detection serves as a foundational building block for higher-level computer vision tasks such as scene understanding, semantic segmentation, and action recognition, facilitating deeper insights and decision-making capabilities in intelligent systems.

The emergence of deep learning revolutionized object detection, leading to significant advancements in accuracy and efficiency. Convolutional Neural Networks (CNNs) have demonstrated remarkable potential and have been the main research direction in recent years [1]. While the performance of deep learning-based object detection models has significantly improved, their vulnerability to adversarial attacks poses a significant challenge to their reliability and security [2]. Adversarial attacks involve maliciously crafted perturbations to input data, which can cause deep neural networks to make incorrect predictions or fail to detect objects altogether [3]. In safety-critical applications like autonomous driving, the susceptibility of object detection systems to adversarial attacks can have severe consequences, jeopardizing human safety and privacy [4]. Adversarial defence is a critical area of research focused on enhancing the robustness of deep learning models against adversarial attacks. While deep learning models are typically optimized to maximize performance metrics on clean images, adversarial training aims to increase performance in adversarial scenarios [2]. Despite the importance of adversarial defence techniques, there is a noticeable discrepancy in the number of papers focusing on adversarial attacks compared to defences in the context of object detection. This asymmetry highlights the need for more research efforts dedicated to adversarial defence in object detection, to mitigate the growing threat posed by adversarial attacks and enhance the robustness of deep learning models in practical settings [5].

Despite the recent success, deep learning methods are computationally intensive during training due to the need to process large volumes of data through multiple layers, while simultaneously updating millions of parameters through backpropagation [6]. Therefore, as the models get larger and more complex, efficiency becomes an increasing conern. Efficiency can be measured as any type of reduction of computational resources, such as through memory, processing power, and energy consumption. In this

work, the efficiency of the model will be measured in inference time, which is advantageous because it directly reflects the model's performance in real-world applications, particularly those requiring real-time or low-latency processing. Inference time indicates how quickly a trained model can make predictions, which is critical for tasks like autonomous driving, video-based object detection, or any time-sensitive application. A model with shorter inference time can handle more data in a given period, improving overall throughput and user experience.

Various methods can be employed to enhance CNN efficiency, such as transfer learning, quantization, weight sharing, and pruning. Pruning is a key technique for improving efficiency and comes in two forms: unstructured pruning and structured pruning. Unstructured pruning removes individual weights that contribute little to the network's performance, leading to sparser weight matrices. While effective in reducing the number of parameters, unstructured pruning can result in irregular memory access patterns, making it harder to optimize on certain hardware. In contrast, structured pruning removes entire filters, channels, or layers, resulting in a more compact architecture. This type of pruning not only reduces the parameter count but also decreases the computational load, making it more suitable for real-time deployment on hardware like GPUs or mobile devices. Structured pruning maintains the integrity of the network structure, allowing for faster inference times while maintaining accuracy.

II. Background

A. Object Detection

To train a CNN model for object detection, a large labeled dataset is required, where each image contains annotations for the objects present, including their class labels and bounding box coordinates. The model is trained by passing input images through a network that extracts features and learns patterns, with the help of techniques like backpropagation and gradient descent to minimize the error between the predicted and true labels. During training, the CNN learns to detect objects by adjusting its weights to improve both classification accuracy and localization precision, ultimately producing a model capable of accurately identifying and locating objects in unseen images.

1) Model: CNN models for object detection generally consist of a feature extraction backbone and a detection head. The backbone, often a pre-trained deep CNN such as ResNet or VGG, extracts high-level features from images, which are then fed into the detection head to perform localization and classification tasks. Some models, like Faster R-CNN, use a region proposal mechanism, while others like SSD (Single Shot Multibox Detector) predict bounding boxes and class labels in a single pass. YOLO (You Only Look Once) models take a unique approach by dividing the image into a grid and predicting bounding boxes and class probabilities simultaneously, making them highly efficient for real-time applications. YOLOv3, in particular, improves upon earlier versions with better detection at multiple scales, making it an excellent model for balancing speed and accuracy in object detection tasks.

2) Dataset: Datasets play a pivotal role in advancing the field of object detection by providing annotated images for model training and evaluation. Many openly available datasets exist online, such as Pascal VOC, ImageNet, and Open Images [7]. Among these datasets, the Microsoft Common Objects in Context (MS COCO) dataset stands out as one of the most comprehensive and widely used benchmarks [8]. Its superiority stems from several factors. Firstly, COCO has a diverse range of object categories, encompassing common everyday objects across 80 distinct classes, including people, animals, vehicles, and household items. This diversity ensures that models trained on COCO generalize well to a wide array of real-world scenarios. Secondly, each image in the dataset is annotated with instance-level bounding

box coordinates, and class labels for precise training and evaluation of detection algorithms. Additionally, COCO provides a large-scale dataset comprising over 200,000 images split into training, validation, and test sets, facilitating robust model training and unbiased evaluation [8].

B. Efficient Adversarial Robustness

Many papers have attempted to tackle the problem of making CNNs robust to adversarial attacks, but also making them more efficient in the process.

1) Towards Compact and Robust Deep Neural Networks: Introduced by Sehwag et al. in 2019, the work "Towards Compact and Robust Deep Neural Networks" evaluated the robustness of CNNs under both structured and unstructured pruning. The authors provide a formal definition of the pruning procedure, encompassing pre-training, weight pruning, and fine-tuning, which clarifies the methodology's effectiveness in achieving compact networks without compromising performance. Empirical results demonstrate that the proposed method can maintain, on average, 93% benign accuracy, 92.5% empirical robust accuracy, and 85% verifiable robust accuracy while achieving a compression ratio of 10×.Their experimental data showed that their proposed method is effective for both pruning methods, but especially for unstructured pruning.

2) HYDRA: In a follow up paper, HYDRA: Pruning Adversarially Robust Neural Networks, is a seminal work that has taken another step to improving the performance of machine learning models. Sehwag et al. developed a pruning method that is aware of the robust training objective. This is achieved by formulating the pruning process as an empirical risk minimization (ERM) problem combined with a robust training objective, which is solved efficiently using Stochastic Gradient Descent (SGD). The authors also introduced importance score based optimization, with every weight being initialized with a importance value that is proportional to pre-trained network weights. This change has shown to help with the final performance and speed of SGD convergence over random initialization. Extensive experiments were performed by testing with CIFAR-10, SVHN, and ImageNet datasets, and with four robust training techniques: iterative adversarial training, randomized smoothing, MixTrain, and CROWN-IBP. HYDRA achieves state-of-the-art performance in both benign and robust accuracy, even at high pruning ratios (up to 99%). The method shows significant gains in robust accuracy while also improving benign accuracy compared to previous works.

The flow of HYDRA can be described in three steps:

1) Pre-training (train with adversarial, without pruning)
2) Pruning (freeze weights and only update importance score, generate pruning mask)
3) Fine-tuning (update non pruned weights but freeze importance scores)

The paper was chosen as the foundation of this work, from its excellent performance and extensive documentation. The performance of the pruning algorithm was strong enough to earn second place in the auto-attack robustness benchmark near the time the paper was published, and the paper has been cited 100+ time since the time of publication.

III. Method

A. Adapting hydra to object detection

The HYDRA method is developed for the task of image classification, where only the type of object in the image is recoganized. Often it is useful to learn the location of the objects inside the images as well, which leads to the task of object detection. Therefore, the CNN model was replaced by YOLOv3, and the dataset was replaced by COCO.

B. Implementing unstructured and structured pruning

Subsection text here.

C. New initialization technique

DeepLIFT

IV. Experiments

In the context of object detection using the COCO dataset, mAP serves as a fundamental metric for evaluating the performance of detection models. mAP quantifies the precision-recall trade-off by calculating the average precision across different object categories. COCO dataset's mAP is typically computed at various Intersection over Union (IoU) thresholds such as 0.5, which is the threshold used in all experiments in the work. Also, a subset of 50,000 images from MS COCO is used to reduce training time.

A. Training time penalty

B. Inference performance

Figure 1

insert ANOVA test results

C. unstructured vs structured

Figure 1

D. effect of greyscale on performance

Figure 2

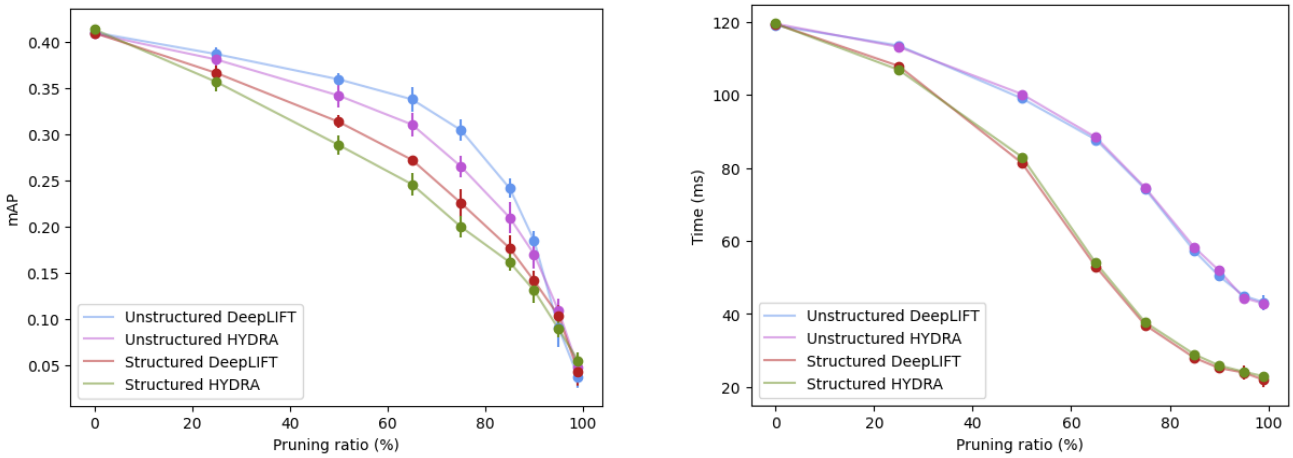

Fig. 1: Comparison of proposed approach versus HYDRA, through both unstructured and structured pruning. The mAP performance is shown on the left, and time on the right

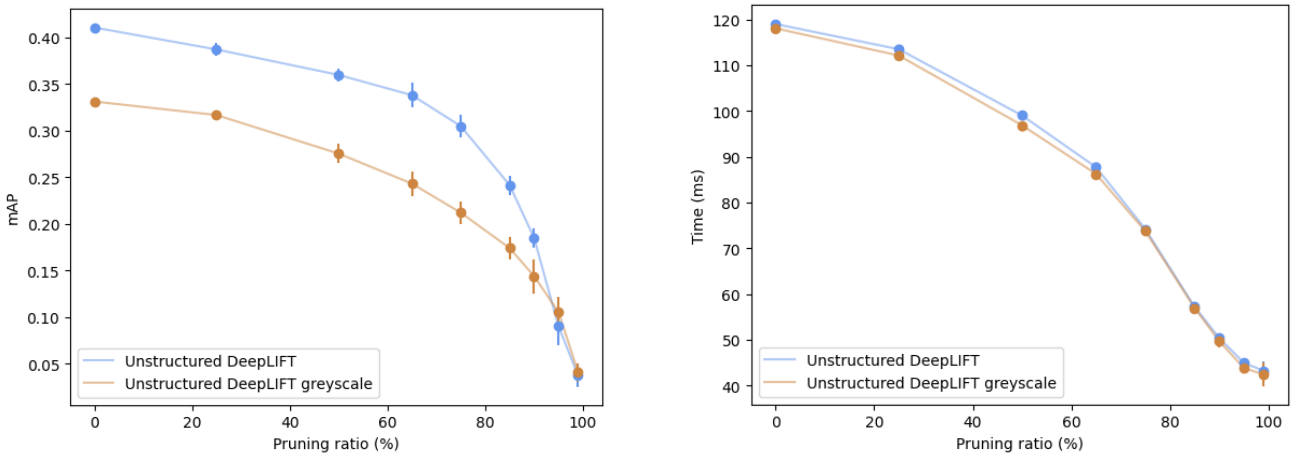

Fig. 2: Comparison of performance between color and greyscale images on the proposed method. The mAP performance is shown on the left, and time on the right

E. example pictures

Subsection text here.

V. Conclusion

Conclusion here.

Appendix A
Proof of the First Zonklar Equation

Appendix one text goes here.

APPENDIX B

Appendix two text goes here.

ACKNOWLEDGMENT

The authors would like to thank...

REFERENCES

[1] Z. Zou, K. Chen, Z. Shi, Y. Guo, and J. Ye, *Object detection in 20 years: A survey*, 2023. arXiv: 1905.05055 [cs.CV].

[2] J. C. Costa, T. Roxo, H. Proença, and P. R. M. Inácio, *How deep learning sees the world: A survey on adversarial attacks & defenses*, 2023. arXiv: 2305.10862 [cs.CV].

[3] H. Li, G. Li, and Y. Yu, *Rosa: Robust salient object detection against adversarial attacks*, 2019. arXiv: 1905.03434 [cs.CV].

[4] H. Zhang and J. Wang, *Towards adversarially robust object detection*, 2019. arXiv: 1907.10310 [cs.CV].

[5] T. Bai, J. Luo, J. Zhao, B. Wen, and Q. Wang, *Recent advances in adversarial training for adversarial robustness*, 2021. arXiv: 2102.01356 [cs.LG].

[6] G. K. Erabati, N. Gonçalves, and H. Araujo, "Object detection in traffic scenarios - a comparison of traditional and deep learning approaches," Jul. 2020, pp. 225–237. DOI: 10.5121/csit.2020.100918.

[7] S. S. A. Zaidi, M. S. Ansari, A. Aslam, N. Kanwal, M. Asghar, and B. Lee, *A survey of modern deep learning based object detection models*, 2021. arXiv: 2104.11892 [cs.CV].

[8] T.-Y. Lin *et al.*, *Microsoft coco: Common objects in context*, 2015. arXiv: 1405.0312 [cs.CV].