Paper Summary:

Real-Time Flying Object Detection with YOLOv8

Paper Link: https://arxiv.org/abs/2305.09972

I) Introduction / Method / Goal

With the development of drones and their possible malicious usage, new challenges are arising such as how to detect them. Drones are hard to detect because of their small size, high maneuverability, and low level of noise which make them a perfect tool for modern warfare and illegal activities. Currently, the proposed detection methods for drones such as radio and acoustic detection aren't reliable enough. The main goal of this paper is to develop a reliable low-inference but high-precision flying object (drone, bird, airplane, helicopter) detector based on a vision camera using the recently released state-of-the-art architecture YOLOv8. This model needs to be refined but still able to generalize and detect a wide variety of classes and features to be useful for further research and flying object detection applications thanks to transfer learning.

II) Model Training

Before training the model, they made their dataset by combining multiple datasets found online with a high number of classes of diverse size and image resolution. The model training phase started with a baseline evaluation of the default YOLOv8 architecture. The two main metrics were mean Average Precision (mAP) which evaluates the accuracy of an object detection model but also the inference which needs to be low to provide a real-time detection system. The medium YOLOv8 architecture was picked for its inference close to the small model (5.7 ms) and with an accuracy nearly as high as the large model (mAP 0.002 lower). The total inference for 1080p HD video was 19.75 ms considering all the pre-process and post-process needed by the model. Because of the lack of computational power, they used a greedy hyperparameter tuning approach giving only 10 epochs per hyperparameter to show a significant increase in performance and then training the first optimized model with 100 epochs, details about the loss function and weight update rule can be found in the paper.

III) Model Evaluation

They validated the model performances during the training thanks to a confusion matrix for all classes and the evolution of the mAP50-95. With 40 different classes, they realized that some low inter-variance classes were most likely to be misclassified by the model. By taking the example of the F-14 and F18 they went through the process of visualizing activation maps to conclude why the mistakes were happening. The goal of an object detection model is to identify the main features specific to a specific task while generalizing to be able to adapt to unknown situations. While the data is going through the different layers of the backbone of the YOLOv8 classifier (CSPDarknet53), features are being extracted from general to specific by progressively down-sampling the input image. This allows the model to generate multiple feature maps with different granularity and on different levels of detail and abstraction for each filter. The similarities in the features map between an F-14 and an F-18 are responsible for the classification errors but overall, the model does very well at distinguishing various types of flying objects despite the high number of classes with slight differences.

IV) Model Refining – Transfer Learning

To make a more specific model, they used another dataset representing the “real world” and applied transfer learning to the model to detect and classify small objects in the appearance of drones, helicopters, airplanes, and birds. Because each object takes a very small portion of the image (around 0.05%), they set the threshold of confidence threshold to 0.2 which is very low for a detector. Another layer-by-layer activation map analysis allowed them to visualize the detection process. The model performance is very good with a mAP of 0.835 across all the classes even though the dataset challenges the model through size, varying objects for the same class, and camouflaging object melding with the background.

V) YOLOv8 overview

At the end of the paper, they do a small recap of YOLO’s: “You Only Look Once” history. YOLOv1 was proposed in 2015 as a single-stage detector with a “refreshingly simple” architecture based on convolutional layers and average pooling layers. This architecture was upgraded over the years and quickly became one of the most popular object detection algorithms.

YOLOv5 achieves state-of-the-art performances on the COCO benchmark dataset. The main change in the model structure is the standardized practice of structuring the model into three components: the backbone, the neck, and the head. On top of that they based the backbone on Darknet53, they also enabled Cross Stage Partial connections, and incorporated anchor boxes with specific size ratios instead of using arbitrary bounding boxes.

YOLOv8 also has numerous improvements compared to its previous versions. It uses both a Feature Pyramid Network (FPN) allowing easier detection of objects at different scales and resolutions and a Path Aggregation Network (PAN) to capture features at multiple scales and resolutions through skip connections. The paper ends with a performance comparison between YOLOv8 and YOLOv5 (YOLOv6 and YOLOv7 being less performant in the benchmark) by using the RF100 where YOLOv8 outperformed YOLOv5 in every single category. On top of the model performances, Ultralytics created a whole environment around YOLOv8 to make it an easy out-of-the-box solution thanks to a Python API, advanced post-processing techniques, and a powerful labeling tool.

Why was the publication interesting to you?

This is not the last publication I’ve read on Computer Vision, but I felt that it was one of the most relevant ones considering the project I was doing at the time. I was working on my master’s thesis (Feasibility study of an Eyes-Out module for recreational airplane pilots) and wanted to look at already existing publications around flying object detection. On top of that, I was already considering YOLOv8 as an architecture to base my model since it is considered state-of-the-art for object detection and has a friendly out-of-the-box solution for training that I used in a previous project.

This paper not only studied a subject similar to mine but also had an overview of the YOLOv8 model although the YOLOv8 paper was not out yet, this gave me important data on YOLOv8 architecture and a comparison to old YOLO versions.

I was interested in the reasoning of the researchers as they explicitly talked about their approach and assumptions while doing their project and also how they tried to explain and understand why the model was behaving a certain way by looking at the activation maps. The transfer learning part was also welcome since it could have been a good starting point for my project.