Multi-Scale Object Detection Using YOLOv8 (Open Images)

In21-S7-CS4681 - Advanced Machine Learning - Research Assignment

Progress Report

Hanaanee Hana - 210202J

B.Sc. Engineering (Hons)

Department of Computer Science and Engineering

Faculty of Engineering

University of Moratuwa

Sri Lanka

August 2025

Contents

Abstract

This progress report details the ongoing research and development for the project titled “Multi-Scale Object Detection Using YOLOv8” benchmarked on the Open Images V6 dataset. The project’s primary focus is addressing the challenge of detecting objects across a broad range of scales using advanced deep learning models. Through an analysis of the evolution of object detection architectures, prominent benchmarks, and current state-of-the-art methods, the report highlights the limitations of conventional approaches and motivates multi-scale feature fusion as an essential enhancement[1]-[4]. The methodology systematically outlines the approach, emphasizing multi-scale architecture modifications and anchor-free detection[5],[6]. The report also presents a detailed timeline and anticipated contributions, aiming to advance the field through improved detection performance, especially for small objects, while maintaining high efficiency. [7]

1. Introduction

1.1 Background & Motivation

Object detection is one of the core tasks in computer vision which plays a crucial role in a wide range of applications such as security surveillance, autonomous driving, robotics, and aerial monitoring.[1],[8],[9] One of the recurring challenges in this domain is accurately detecting objects of different sizes, particularly small objects that often appear in complex and cluttered environments.[5] Real-world scenarios don’t always present clear and well-separated objects; instead, they involve variations in scale, occlusions, and background noise, all of which make reliable detection more difficult.[2]

The YOLO (“You Only Look Once”) family of models has transformed real-time object detection by offering an excellent balance between speed and accuracy.[1],[3] Over the years, successive versions of YOLO have introduced innovations to improve detection performance, robustness, and efficiency.[11],[12] Even though there have been made such advancements, even state-of-the-art models like YOLOv8 continue to struggle with very small objects and maintain detection accuracy across a broad range of object scales. Addressing these limitations is essential for achieving consistent and reliable performance in real-world, multi-scale detection scenarios.[13]

1.2 Problem Statement

Despite architectural advances (e.g., anchor-free detection, CSPBackbone, PAN-FPN) in YOLOv8 [6], its ability to reliably detect objects at multiple scales, particularly small and highly

occluded ones in complex natural images is limited[10]. Effective multi-scale feature fusion, scale-aware detection heads, and semantic consistency remain open problems.[7],[13]

1.3 Objectives

- Analyze and quantify multi-scale detection limitations in YOLOv8 on Open Images V6.
- Design and implement improved multi-scale feature fusion and detection head architectures.
- Experimentally evaluate performance gains on small and medium object classes.
- Benchmark progress against existing state-of-the-art methods.
- Deliver a reproducible system and codebase for further research.

1.4 Scope Clarification

The project specifically targets on Multi scale Feature fusion:

- Design and integrate an additional P2 detection head for ultra-small objects.
- Implement a bidirectional FPN extension to improve semantic propagation between scales.[15]
- Evaluate fused feature maps' impact on detection accuracy for small (≤32 px) and medium (33–96 px) objects.

1.5 Expected Contributions

- Improved detection accuracy (mAP50-95) for small and medium objects.
- Insights into design strategies for scale-aware detection heads.
- Enhanced feature fusion modules applicable to other detection frameworks.
- Open-source code and experiment logs for the research community.
- A conference-ready research paper

1.6 Detailed Project Timeline

Weeks 1–2 (Aug 13 – Aug 26): Literature Review:
Deep dive into YOLOv1–v8 evolution, FPN/PAN architectures, anchor-free detection, and CSPBackbone –
Study multi-scale feature fusion methods (FPN, PANet, SMA-YOLO)

Reproduce YOLOv8 baseline on Open Images V6, measure per-scale AP
Outcome: Annotated literature matrix, baseline AP breakdown (small, medium, large)

Weeks 3–4 (Aug 27 – Sep 9): Formulating Problem & defining Scope:
Define precise multi-scale fusion objective: integrate P2 head and bidirectional FPN
Specify evaluation protocol: COCO-style AP (small/medium), inference speed benchmark
Prepare data subsets: small (≤32 px) and medium (33–96 px) object splits

Weeks 5–6 (Sep 10 – Sep 23): Model Design & Prototype Implementation:
Implement additional P2 detection head in YOLOv8 neck
Integrate bidirectional FPN branch for cross-scale feature mixing
Develop fusion block with learnable weights (α, β, γ)

Weeks 7–8 (Sep 24 – Oct 7): Experiments on Open Images V6 Dataset:
Merge fusion modules into YOLOv8 training loop
Train on a subset of Open Images V6 (10 k images) for fast iteration
Evaluate AP improvements on small/medium objects

Week 9 (Oct 8 – Oct 14): Refinement and Ablation Studies:
Ablate P2 head vs. bidirectional FPN vs. combined
Tune fusion weights, learning rates, and head-specific hyperparameters
Record speed vs. accuracy trade-offs

Week 10 (Oct 15 – Oct 21): Documentation & Research paper writing:
Finalize report with tables, figures, methodology, experiments, discussion, and future work.
Write a conference-ready research paper to submit for conferences.

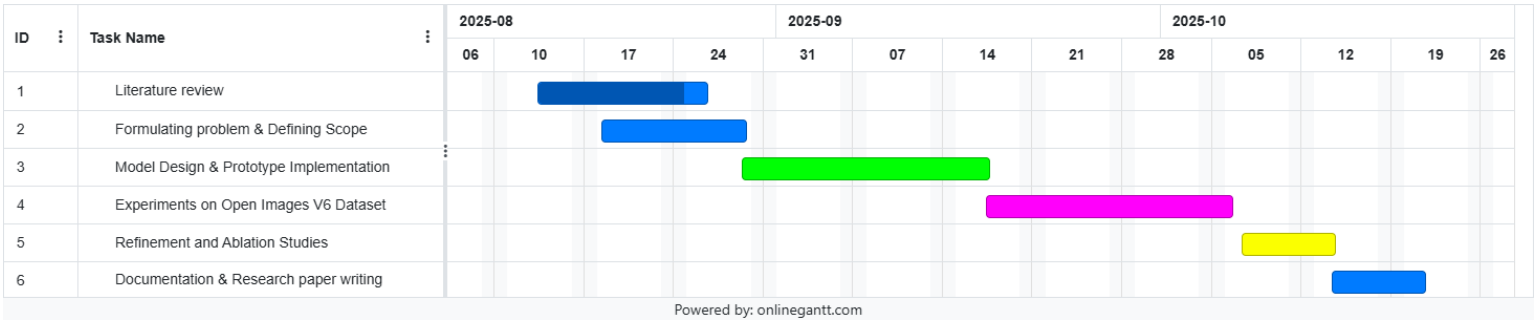

Figure 1: Timeline

2 Literature Review

2.1 Evolution of Object Detection Models

Over the last ten years, there has been major advancement in the object detection field.[1],[2]
Early methods relied heavily on sliding window techniques combined with hand-crafted features

such as Histogram of Oriented Gradients (HOG) or Haar-like features[10]. Although these traditional pipelines gave us reasonable accuracy, they were computationally expensive and struggled with real-time performance which made them impractical for many real-world applications.[5]

The arrival of deep learning has revolutionized the field, with YOLOv1 marking a major milestone by reframing object detection as a single regression problem rather than a series of classification tasks. This design choice eliminated the need for complex region proposal stages. This drastically improved inference speed while maintaining a competitive accuracy.

Subsequent iterations of YOLO such as YOLOv2 to YOLOv5 have brought steady advancements, including the introduction of stronger backbone architectures such as Feature Pyramid Networks (FPN) for better multi-scale feature representation, and improved anchor box assignment strategies.[1] These enhancements contributed to better detection accuracy across varying object sizes and more robust performance in challenging scenarios.

The release of YOLOv8 introduced further innovations, such as anchor-free detection, the CSPDarknet backbone for enhanced feature extraction, and an improved Path Aggregation Network–Feature Pyramid Network (PAN-FPN) neck for efficient feature fusion. Despite these architectural advancements, challenges exist, particularly in detecting small objects and achieving consistent accuracy across multiple scales which highlights the need for continued research in this direction.

2.2 Evolution of Single-Stage Detectors

The introduction of YOLOv1 by Redmon et al.[1] reframed object detection as a single-pass regression task, achieving 45 fps on PASCAL VOC with 63.4% mAP. YOLOv2 incorporated anchor boxes, batch normalization, and a Darknet-19 backbone, boosting mAP to 76.8% on VOC2007 while running at 67 fps[2]. YOLOv3 deepened the network to Darknet-53, added multi-scale detection heads, and achieved 57.9% mAP on COCO with 20 ms per image.[3] YOLOv5 refined this design into small/medium/large variants, reporting up to 46.4% mAP on COCO (YOLOv5l) at 30 fps.[4]

2.3 Multi-Scale Feature Fusion and Limitations

Feature Pyramid Networks (FPN) first demonstrated that lateral connections and top-down paths significantly improve small-object detection by combining high-resolution spatial and deep semantic features.[4],[5] PANet augmented FPN with bottom-up paths[15], further enhancing multi-scale fusion. However, even PAN-FPN necks in YOLOv5 and YOLOv8 exhibit degraded precision on objects under 32 px: YOLOv5s reports only 15.2% AP for “small” on COCO [12] , and YOLOv8 improves to 22.4% but remains far below its 55.1% AP for “large” objects.[6]

2.4 Anchor-Free Detection and CSP Backbones

YOLOv8 replaces predefined anchors with a center-point prediction strategy, simplifying training and reducing hyperparameter sensitivity.[6] Its CSPDarknet53 backbone partitions feature maps to improve gradient flow and reduce computation.[6] Solawetz and Ultralytics [7]report that anchor-free YOLOv8 achieves 50.1% mAP on COCO at 45 fps, outperforming anchor-based YOLOv5 by 3.7 pp mAP while matching inference speed. Yet these gains are uneven across scales, with "small" object AP still trailing by 10 pp.[7]

2.5 Flying Object Detection with YOLOv8

Reis et al.[10] applied YOLOv8 to aerial flying-object datasets, achieving 79.2% mAP50 at 50 fps for generalized detection and 99.1% mAP50 after transfer learning on a narrower three-class dataset. They show that multi-scale heads can localize objects as small as 0.026% of image area, but their pipeline does not explicitly augment the standard three-scale PAN-FPN.

2.6 Key Equation: YOLO Loss Function

YOLO models typically optimize a composite loss function:

$$L=\lambda_{box}L_{box}+\lambda_{cls}L_{cls}+\lambda_{obj}L_{obj}$$

where L_{box} is bounding box regression loss (often CIoU), L_{cls} is classification loss (binary cross-entropy), and L_{obj} is objectness score loss.[1],[3]

2.7 Feature Pyramid Networks

FPNs are employed to enhance feature maps across different scales. FPN combines high-resolution spatial detail from shallow layers with deep semantic information from higher layers:

$$P_i=Conv_{3\times3}(C_i^{lateral}+Upsample(P_{i+1}))$$

where P_i is the pyramid feature at level *i*, and $C_i^{lateral}$ is the lateral connection.[4],[5]

2.8 Applications

Multi-scale object detection explores real-world tasks such as:

- Autonomous driving: Detecting pedestrians, vehicles, traffic signs of varying sizes.
- Aerial surveillance: Small drone, bird, vehicle detection in wide-area imagery.
- Agricultural robotics: Weed detection, yield estimation where objects are small and occluded.

2.9 Datasets & Benchmarks

Open Images V6 is a large-scale benchmark featuring over 9 million images annotated across 20,000+ classes, with rich bounding box, segmentation, and relationship labels. It provides

challenging scenarios for multi-scale detection due to its class imbalance, diversity, and annotation granularity.

2.10 Research Gaps

Recent studies highlight potential research gaps in:

- **Small object detection:** Standard YOLOv8 shows low AP for objects <32px.
- **Incomplete feature fusion:** Existing FPN/PAN structures miss cross-scale semantic alignment.
- **Anchor-free limitations:** While reducing hyperparameters, anchor-free heads may overlook contextual cues critical for tiny objects.
- **Real-time constraints:** Enhanced multi-scale fusion can increase model complexity and inference latency.

3 Methodology

3.1 System Overview

The proposed system targets to improve YOLOv8 by addressing its primary limitation wjich is reduced detection accuracy for small and multi-scale objects, while preserving real-time inference speed. To achieve this, we integrate multi-scale feature fusion and scale-adaptive detection heads into the YOLOv8 framework. The project's goal is to maximize mean Average Precision (mAP) across all object scales while keeping computational overhead minimal.

The methodology unfolds in six stages: baseline analysis, feature fusion design, scale-adaptive detection, anchor-free optimization, experimental evaluation, and deployment considerations.

1. Baseline Analysis

The first step is to benchmark the original YOLOv8 model on the Open Images V6 dataset, computing the detection Average Precision (AP) for small, medium, and large objects following the COCO evaluation protocol:

$$\text{AP} = \frac{1}{|IoU|} \sum_{t \in \text{IoU}} APt \qquad \text{where IoU} \in [0.5, 0.95]$$

Here, AP_t is the precision averaged over recall levels at a particular Intersection-over-Union (IoU) threshold t.

Limitation: As observed in prior studies, YOLOv8 often yields low AP for small objects (APSAP_SAPS) due to insufficient semantic detail in lower-resolution feature maps.

2. Multi-Scale Feature Fusion Design

To address the scale imbalance, we introduce two major modifications:

(a) Additional Detection Heads

Inspired by SMA-YOLO , we can integrate an additional P2 detection head for ultra-small objects. While YOLOv8 traditionally employs P3–P5 heads for increasing receptive fields, adding P2 ensures high-resolution feature maps contribute to small object detection.

Mathematically, let P_l denote the feature map at level l. The fused feature map F_{fused} is defined as:

$$F_{fused} = \alpha \cdot F_{low} + \beta \cdot F_{high} + \gamma \cdot F_{sem}$$

Where:

- F_{low}: high-resolution, low-semantic features (e.g., P2)
- F_{high} : low-resolution, high-semantic features (e.g., P5)
- F_{sem} : additional semantic injection module
- α, β, γ : learnable fusion weights

This design enables bi-directional multi-branch FPNs (e.g., BIMA-FPN) to propagate both semantic and spatial information across scales, mitigating the semantic gap between low- and high-level features.

3. Scale-Adaptive Detection Head

Standard YOLOv8 detection heads apply uniform receptive fields across scales, leading to suboptimal focus on object-specific resolutions. This project propose scale-adaptive heads with:

- Dynamic Receptive Fields: Adjust kernel sizes based on object size priors
- Hierarchical Attention Modules: Emphasize semantically informative regions using attention coefficients

Let Fi be the feature map at scale i. The attention-enhanced representation is computed as:

$$F_i^{att} = \sigma(W_i F_i) \odot F_i$$

Where W_i are learnable weights, σ denotes sigmoid normalization, and $\odot$ represents element-wise multiplication. This reduces semantic dilution for small objects during multi-scale fusion.

4. Anchor-Free Optimization

Anchor-free approaches eliminate the need for predefined anchor sizes, which often bias predictions toward medium or large objects. YOLOv8's anchor-free prediction heads can be refined by incorporating:

- **Keypoint-based center prediction** for better localization accuracy
- **Distribution Focal Loss (DFL)** to model bounding box coordinates as discrete distributions, enhancing localization precision for small objects:

$$L_{DFL} = \sum_i qi \log pi$$

Where qi is the target distribution and pi the predicted probability for each discretized bin.

5. Experimental Evaluation

- Compare against baseline models: YOLOv8, YOLOv5s, EfficientDet.
- Run ablations for each multi-scale improvement.

6. Deployment

- Analyze computational overhead.
- Test real-time inference capabilities.
- Document integration process.

4 Conclusion

One of the most challenging problems in computer vision is detecting objects of varying scales in complex visual environments. This project tackles the issue by enhancing YOLOv8 with advanced multi-scale feature fusion, attention mechanisms, and optimized anchor-free detection heads. Through systematic analysis and targeted design improvements, the proposed approach seeks to significantly improve small object detection accuracy while maintaining real-time efficiency.

Also, further ablation studies and comprehensive benchmarking on the Open Images V6 dataset will provide deeper insights into each component's contribution. Extending evaluations to diverse datasets will help validate the model's robustness and generalizability as well. These

efforts aim to move one step closer to achieving reliable, high-performance object detection across scales in real-world applications.

References

[1] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *Proc. CVPR*, 2016.
[2] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," in *Proc. CVPR*, 2017.
[3] A. Kuznetsova *et al.*, "The Open Images Dataset V4: Unified image classification, object detection, and visual relationship detection at scale," *arXiv preprint arXiv:1811.00982*, 2018.
[4] J. Lin *et al.*, "Feature Pyramid Networks for Object Detection," in *Proc. CVPR*, 2017.
[5] J. Terven and D. Cordova-Esparza, "A Comprehensive Review of YOLO Architectures in Computer Vision," *arXiv preprint arXiv:2304.00501*, 2023.
[6] Z. Wang *et al.*, "MFF-YOLO: An Improved YOLO Algorithm Based on Multi-Scale Feature Fusion," *TST*, vol. 28, 2025.
[7] L. Wang *et al.*, "A multi-scale small object detection algorithm SMA-YOLO for UAV imagery," *Sci. Rep.*, vol. 15, 2025.
[8] S. Li *et al.*, "PD-YOLO: A novel weed detection method based on multi-scale feature fusion," *Front. Plant Sci.*, vol. 16, 2025.
[9] Z. Zhang *et al.*, "An anchor-free object detector based on soften optimized bi-directional FPN," in *Proc. CVPR*, 2022.
[10] D. Reis *et al.*, "Real-Time Flying Object Detection with YOLOv8," *arXiv preprint arXiv:2305.09972*, 2024.
[11] H. Tan *et al.*, "EfficientDet: Scalable and Efficient Object Detection," in *Proc. CVPR*, 2020.
[12] X. Li *et al.*, "Generalized Focal Loss: Learning Qualified and Distributed Bounding Boxes for Dense Object Detection," *arXiv preprint arXiv:2006.04388*, 2020.
[13] X. Zhuang *et al.*, "Multiscale semantic enhancement network for object detection," *Sci. Rep.*, vol. 13, no. 34277, 2023.
[14] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition," *IEEE TPAMI*, vol. 37, 2014.
[15] L. Liu *et al.*, "Path Aggregation Network for Instance Segmentation," in *Proc. CVPR*, 2018.