IDENTIFICATION/DETECTION OF NON-OBJECT AREAS FOR DATA MINING IN IMAGES USING YOLOV8

A Project Based Learning Report Submitted in partial fulfilment of the requirements for the award of the degree

of

Bachelor of Technology

MULTIMODAL INFORMATION PROCESSING

23ALT3102E

Submitted by

2310080022: JAGGAVARAPU AKHIL SAI REDDY
2310080007: MOHAMMED SHOAIB HANNAN

Under the guidance of

Dr. Gangamohan Paidi

(DEEMED TO BE UNIVERSITY)

Department of Electronics and Communication Engineering

Koneru Lakshmaiah Education Foundation, Aziz Nagar

Aziz Nagar – 500075

FEB - 2025.

Introduction

YOLOv8 is an advanced object detection algorithm that improves speed, accuracy, and robustness, building on previous YOLO versions to better detect small and occluded objects. Its architecture benefits from modern design choices such as decoupled heads, improved non-maximum suppression (NMS) methods like Soft-NMS or DIoU-NMS, and optimized loss functions that balance classification, localization, and confi- dence scores. YOLOv8 also adopts anchor-free detection, re- ducing design complexity and improving generalization across diverse datasets. These features make YOLOv8 particularly effective for real-time applications such as surveillance, smart cities, and UAV monitoring.

IA-YOLO (Image-Adaptive YOLO) emphasizes adaptabil- ity to harsh environments. Its differentiable preprocessing module adjusts contrast, brightness, and dehazing operations on the fly, guided by the model's feedback loop. IA-YOLO represents a step toward self-tuning perception systems, es- pecially for outdoor applications like autonomous driving and maritime navigation where visual clarity can vary dramatically. YOLO with Adaptive Frame Control (AFC) is especially significant for video analytics in embedded systems. AFC integrates temporal redundancy awareness—intelligently skip- ping or blending frames based on change detection. This ensures minimal loss in temporal consistency while conserving computation, which is ideal for drone flight, wildlife tracking, and battery-powered surveillance.

DAMO-YOLO leverages AutoML-based architecture tuning and advanced modules like ZeroHead (a lightweight detection head) and AlignedOTA (a refined label assignment strategy improving convergence speed and stability). The model is also optimized for mixed-precision training, further improving effi- ciency on GPUs and edge TPUs. DAMO-YOLO demonstrates strong performance across both large-scale datasets like COCO and lightweight benchmarks like PASCAL VOC.PP-YOLO series (including PP-YOLOv2 and PP-YOLOE) integrates diverse improvements such as GIoU loss, EMA (Exponential Moving Average) for parameter smoothing, and Matrix NMS. PP-YOLO maintains impressive accuracy and latency performance, making it widely used in real-time indus- trial inspection, public safety, and autonomous retail systems.

Quantized-YOLO enables YOLO to run with reduced- bitwidth arithmetic (INT8, INT4) using methods like QAT (Quantization Aware Training) and Post-Training Quantiza- tion. This greatly reduces energy consumption and enables deployment on ultra-low-power microcontrollers (e.g., Cortex-M series), making it viable for wearables, smart agriculture sensors, and onboard UAV systems.

YOLO-World bridges the gap between visual grounding and object detection by integrating CLIP-like vision-language models. This allows for zero-shot detection, where objects can be identified using textual prompts. It is useful in dynamic environments where new objects appear frequently, such as search-and-rescue missions, military reconnaissance, and in- teractive AI assistants.

LITERATURE REVIEW

Object detection has evolved significantly with models such as Fast R-CNN, Faster R-CNN, and the YOLO family, each offering a balance between accuracy and speed for real-world applications.

Fast R-CNN is widely used for object detection in images and videos, with applications in autonomous driving, surveillance, and medical imaging. It achieves mean Average Precision (mAP) scores of 66.9% on VOC 2007, 66.1% on VOC 2010, and 65.7% on VOC 2012, improving to 68.4% when trained with additional data. Although faster than earlier methods, it is not suitable for real-time use due to slow region proposal generation. Moreover, training is computationally expensive, requiring high-end GPUs and long processing times. The Region of Interest (RoI) pooling layer also causes information loss, reducing accuracy for small objects. Excessive object proposals can further confuse the model, and multi-scale training demands large memory and processing power [8,11]. Despite these challenges, Fast R-CNN remains a strong baseline due to its balance between speed and accuracy [10].

Faster R-CNN improves upon Fast R-CNN by integrating a Region Proposal Network (RPN), achieving an mAP of 76.4%. However, it remains slower than one-stage detectors such as YOLO. In contrast, YOLO (You Only Look Once) achieves an mAP of 78.6% with an impressive 155 FPS, making it far more suitable for real-time applications. Despite this, YOLO struggles with small object detection, unusual aspect ratios, and overlapping objects, and it requires significant computational resources [8,11].

YOLO-based detectors are extensively used across domains such as autonomous driving, surveillance, medical imaging, agriculture, and industrial automation due to their high inference speed. While YOLO achieves an mAP of 63.4% compared to 70% for Fast R-CNN, it is nearly 300 times faster, reinforcing its dominance in real-time systems. Nevertheless, it faces difficulties in detecting small or occluded objects and is sensitive to environmental variations [9].

DAMO-YOLO, a recent real-time detection model, demonstrates strong performance with mAP scores between 43.6% and 51.9% on the COCO dataset across various scales while maintaining low latency on NVIDIA T4 GPUs. Its lightweight versions optimized for edge devices achieve 32.3%–40.5% mAP, offering efficient performance on x86 CPUs.

A newer model leveraging EfficientNet-B4 as its backbone and NAS-FPN for advanced feature fusion achieves an Average Precision Across Scales (APAS) of 52.7 on COCO, surpassing YOLOv7's 50.3. This model, often associated with YOLOv8, offers better feature extraction but still struggles with small object detection, occlusions, and high computational demands, limiting deployment on low-power devices [1].

ViT-YOLO, combining Vision Transformers with YOLO, is designed for drone-based imagery and applications like aerial surveillance, precision agriculture, delivery systems, and autonomous navigation. It integrates Multi-Head Self-Attention (MHSA) and BiFPN modules to improve feature fusion and small object detection.

The AFC-enhanced YOLO variant maintains high accuracy while optimizing frame control for real-time video feeds, reducing frame delay in network camera inputs. However, it remains hardware-dependent and computationally intensive, particularly in complex visual environments [3].

Finally, Image-Adaptive YOLO (IA-YOLO) focuses on robust object detection in adverse weather and low-light conditions. With applications in autonomous driving and surveillance, it employs a Differentiable Image Processing (DIP) module to adapt to environmental changes. IA-YOLO achieves an mAP of 72.03% on VOC_Foggy and 37.08% on RTTS, outperforming conventional YOLOv3 in foggy and low-light scenarios. While it enhances detection performance under challenging conditions, it still requires significant computational power and struggles with extreme weather variations [2].

Image-Adaptive YOLO (IA-YOLO) Loss for Adverse Weather Conditions:

$$L_{\text{IA-YOLO}} = L_{\text{YOLO}} + \lambda_{\text{DIP}} L_{\text{DIP}}$$

Where:

- LYOLO is the original YOLO loss function,
- LDIP represents the differentiable image processing mod- ule loss,
- λDIP is a weighting factor for the importance of the DIP module.

TABLE I
PERFORMANCE FACTORS FOR NON-OBJECT AREA DETECTION USING YOLOv8

Parameter	Issue	Impact	Suggested Fix
Labeling Errors	High	FP↑, Accuracy↓	Better annotation
Dataset Diversity	Low	Recall↓	Add diverse samples
Small Objects	Missed	Detection↓	Use multi-scale
Anchor Fit	Poor	IoU↓	Custom anchors
Confidence Threshold	Default	FN↑ or FP↑	Threshold tuning
Lighting Variation	Present	Accuracy↓	Domain adaptation
Resolution	Low	Missed Areas	Use HD images
Evaluation Metrics	mAP only	Misleading	Add FNR, specificity
Post-Processing	Inaccurate	False regions	Refined NMS, bbox

References

[1] J. Redmon and A. Farhadi, ”YOLO9000: Better, Faster, Stronger,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2017, pp. 7263–7271.https://doi.org/10.1109/ACCESS.2022.3146884

[2] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, ”YOLOv4: Op- timal Speed and Accuracy of Object Detection,” arXiv preprint arXiv:2004.10934, 2020.

[3] D. Barry et al., ”xYOLO: A Model for Real-Time Object Detection in Humanoid Soccer on Low-End Hardware,” in Proc. 2019 Int. Conf. Image Vis. Comput. New Zealand (IVCNZ), IEEE, 2019.

[4] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, ”YOLOv4: Op- timal Speed and Accuracy of Object Detection,” arXiv preprint arXiv:2004.10934, 2020. (duplicate entry, merged with [2]).

[5] F. Sultana, A. Sufian, and P. Dutta, ”A Review of Object Detection Mod- els Based on Convolutional Neural Network,” in Intelligent Computing: Image Processing Based Applications, Springer, 2020.

[6] A. Krizhevsky, I. Sutskever, and G. E. Hinton, ”ImageNet Classification with Deep Convolutional Neural Networks,” in Advances in Neural Information Processing Systems (NeurIPS), 2012, pp. 1097–1105.

[7] S. Albelwi and A. Mahmood, ”A Framework for Designing the Archi- tectures of Deep Convolutional Neural Networks,” Entropy, vol. 19, no. 6, p. 242, 2017.

[8] T. Diwan, G. Anirudh, and J. V. Tembhurne, ”Object Detection Using YOLO: Challenges, Architectural Successors, Datasets and Applica- tions,” Multimed. Tools Appl., vol. 82, pp. 9243–9275, 2023.

[9] J. Carreira, R. Caseiro, J. Batista, and C. Sminchisescu, ”Semantic Segmentation with Second-Order Pooling,” in Proc. Eur. Conf. Comput. Vis. (ECCV), 2012.

[10] T. Diwan, G. Anirudh, and J. V. Tembhurne, ”Object Detection Using YOLO: Challenges, Architectural Successors, Datasets and Ap- plications,” Multimedia Tools and Applications, vol. 82, no. 7, pp. 9243–9275, 2023.

[11] . Carreira, R. Caseiro, J. Batista, and C. Sminchisescu, ”Semantic Segmentation with Second-Order Pooling,” in Proc. Eur. Conf. Comput. Vis. (ECCV), 2012.