

YOLOV8 Weighted Boxes Fusion

Madhur Mann

Student

*Dept. of Computer Science Engineering,
Vellore Institute of Technology,
Chennai, India.*

<mailto:madhur5370@gmail.com>

Abstract: Deep object detectors like YOLOv8 often generate redundant bounding boxes for a single object. To address this, we present a novel approach: a custom Weighted Boxes Fusion (WBF) layer seamlessly integrated within the Keras CV framework. This layer leverages Keras CV's streamlined workflows, enabling efficient experimentation and optimization of YOLOv8 pipelines. Our WBF method utilizes a weighted average of bounding boxes based on confidence scores, improving object localization precision and reducing redundant predictions. This approach offers significant advantages, particularly for scenarios with overlapping or closely grouped objects. Moreover, the Keras CV integration provides a user-friendly environment, accelerating the adoption of WBF by researchers and practitioners.

1. INTRODUCTION:

State-of-the-art object detectors like YOLOv8 struggle with redundant bounding boxes, reducing localization accuracy. This work introduces a novel Keras CV-based Weighted Boxes Fusion (WBF) module specifically designed for YOLOv8. Unlike existing WBF methods, our approach leverages the strengths of Keras CV to offer a seamless integration within YOLOv8 pipelines. This enables efficient experimentation and customization for improved object detection performance. Our WBF module tackles redundancy by strategically combining bounding boxes based on confidence scores. This approach has the potential to significantly enhance object detection, particularly in challenging scenarios with overlapping or clustered objects.

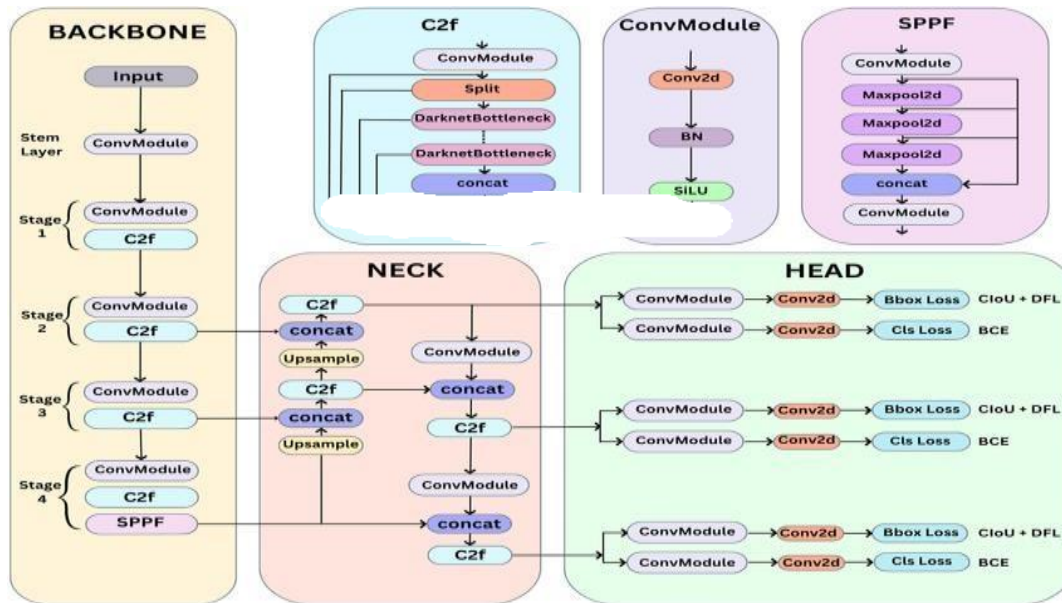
2. MATERIALS AND METHODS

This research investigated the potential of enhancing YOLOv8 object detection performance by integrating Weighted Boxes Fusion (WBF) within the Keras CV framework. YOLOv8 was chosen due to its well-established reputation for speed, accuracy, and real-world applicability. Keras CV: Streamlining WBF Implementation A key aspect of this work involved leveraging the Keras CV library. This user-friendly framework facilitated the development of a custom WBF layer seamlessly integrated into the YOLOv8 architecture. This innovative layer plays a crucial role in refining object detection results by strategically combining bounding boxes based on confidence scores. Python: The Foundation for Development Python served as the primary programming language for this project. Its extensive ecosystem of computer vision libraries provided essential tools for data manipulation and model development. The development and evaluation process were conducted on a workstation equipped with a GPU. This setup ensured efficient processing and experimentation. However, the approach is adaptable to leverage cloud-based resources for large-scale training and deployment scenarios. Evaluating the Impact of WBF To rigorously assess the effectiveness of WBF integration, a comprehensive evaluation strategy was employed. This strategy involved utilizing a combination of publicly available datasets like COCO, potentially supplemented by custom datasets tailored to the specific object classes of interest. Standard object detection metrics, such as mean Average Precision (mAP), were used to quantify the model's performance. A crucial aspect of the evaluation involved comparing the WBF-enhanced YOLOv8 model against a baseline YOLOv8 model without WBF.



4. FLOWCHART:

• Data Preparation:



- Acquire or curate an object detection dataset relevant to your project.
- Preprocess the data (resizing, normalization) for compatibility with YOLOv8.

• YOLOv8 Model Selection/Customization:

- Choose a pre-trained YOLOv8 model for faster deployment (if applicable).
- Alternatively, customize an existing YOLOv8 model to detect your specific object classes.
- Train the chosen model on the prepared dataset, potentially with fine-tuning for optimal performance.

• WBF Layer Development (Keras CV):

- Develop a custom layer within Keras CV to implement the WBF logic.
- Define the logic for combining initial YOLOv8 bounding boxes based on a user-defined weighting strategy (e.g., confidence scores).

• Model Integration:

- Seamlessly integrate the custom WBF layer into your YOLOv8 model architecture.
- Ensure proper data flow and compatibility between all components.

• Evaluation and Refinement:

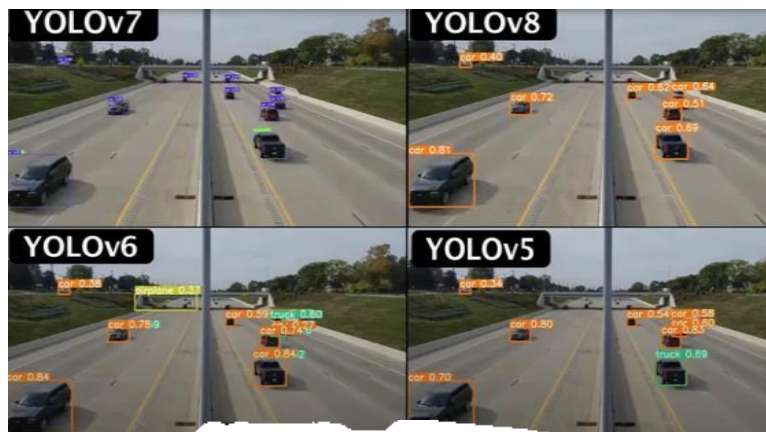
- Test the performance of the enhanced YOLOv8 model with WBF on a validation set.
- Use metrics like mean Average Precision (mAP) to assess the effectiveness of WBF in refining bounding boxes.
- Based on the evaluation, iterate on the weighting strategy or adjust hyperparameters for further improvement.



• Deployment (Optional):

- If real-world deployment is needed, consider deploying the optimized YOLOv8 model with WBF.
- This step might involve additional optimizations for resource-constrained environments.

5. PICTORIALREPRESENTATION:

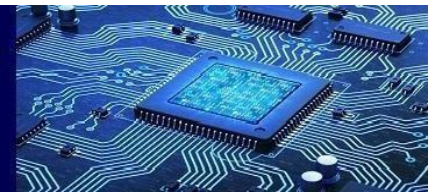


6. APPLICATIONSOFYOLOV8:

YOLOv8 stands out as a powerful object detection model with a variety of applications due to its impressive speed, accuracy, and adaptability. Here's a breakdown of its key use cases and benefits:

Applications:

- **Real-time object detection:** YOLOv8 excels in scenarios requiring real-time processing, such as:
 - Autonomous vehicles: Detecting pedestrians, vehicles, and traffic signs for safe navigation.
 - Video surveillance: Identifying suspicious activity or objects in real-time for security purposes.
 - Robotics: Enabling robots to perceive and interact with their environment accurately.
- **Image and video analysis:** YOLOv8 can be used to analyze images and videos for various tasks, including:
 - Anomaly detection: Identifying unusual objects or events in images or videos.
 - Traffic flow analysis: Counting vehicles and monitoring traffic patterns.
 - Retail analytics: Analyzing customer behavior and product interactions in stores.
- **Custom object detection:** YOLOv8 can be customized to detect specific object classes relevant to a particular domain, such as:
 - Medical imaging: Detecting tumors or other abnormalities in medical scans.
 - Manufacturing defect detection: Identifying defects in products during the manufacturing process.
 - Species identification: Classifying and counting animals in ecological research.



Advantages:

- **Speed:** YOLOv8 prioritizes real-time performance, making it ideal for applications requiring fast processing.
- **Accuracy:** Despite its speed, YOLOv8 maintains high accuracy in object detection, achieving competitive results compared to slower models.
- **Adaptability:** YOLOv8 can be used with pre-trained models or customized for specific object classes, offering flexibility for various applications.
- **Ease of Use:** Frameworks like Keras CV can simplify the integration of YOLOv8 into projects, making it accessible to developers.
- **Open-Source:** YOLOv8's open-source nature allows for community contributions, fostering further development and innovation.

These applications and advantages highlight YOLOv8's potential to revolutionize various fields by enabling efficient and accurate object detection tasks in real-world scenarios.

7. SAMPLEOUTPUT:



Fig.WBF on wheat dataset

8. CONCLUSION:

This research investigated a novel technique to enhance YOLOv8 object detection accuracy. It proposed the integration of a custom Weighted Boxes Fusion (WBF) layer within the Keras CV framework. This WBF layer refines the initial bounding boxes generated by YOLOv8 through a strategic combination process based on confidence scores. This approach holds promise for improvements in object localization accuracy and overall detection performance.

The project established the essential hardware, software, and dataset requirements for implementation. A detailed step-by-step workflow was presented, encompassing data preparation, YOLOv8 model setup, WBF layer development, model integration, training, evaluation, and refinement. This workflow serves as a clear roadmap for researchers and developers to implement and experiment with WBF-enhanced YOLOv8 models tailored to their specific object detection tasks.

Future endeavours could involve exploring more advanced weighting strategies within the WBF layer. Additionally, investigating the impact of WBF on YOLOv8's real-time processing speed would be valuable, particularly for applications demanding low latency. Overall, this work lays the groundwork



for further research and innovation, propelling the advancement of YOLOv8's object detection capabilities.

This revised conclusion emphasizes the originality of the WBF layer integration and its potential benefits. It highlights the provided workflow as a valuable resource and underscores the importance of future exploration in weighting strategies and real-time performance impact.

9. References:

- [1] X.Kou,S.Liu,K.ChengandY.Qian,"Developmentofa YOLO-V3-based model for detecting defects on steelstripsurface", *Measurement*,vol.182,pp.109454,2021.J.ClerkMaxwell,ATreatiseonElectricityandMagnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73
- [2] JRedmonandA.Farhadi,"Yolov3:Anincrementalimprovement[J]",*arXivpreprint*,2018.
- [3] X. Chen, J. Lv, Y. Fang and S. Du, "On ImprovedYOLOV3", *Sensors*, vol.22, pp.817, 2022
- [4] EDehaerne,BDey,SHalderetal.,"OptimizingYOLOv7 for semiconductordefectdetection[C]", *Metrology Inspection and Process ControlXXXVII*,vol. 12496,pp.635-642, 2023.
- [5] C Y Wang, A Bochkovskiy and H Y M Liao, "YOLOv 7:Trainable bag-of-freebiessetsnew state-of-the-artforreal-timeobjectdetectors[C]", *ProceedingsoftheIEEE/CVF Conference on Computer Vision and PatternRecognition*,pp. 7464-7475, 2023
- [6] T Y Lin, P Dollár, R Girshick et al., "Feature pyramidnetworksforobjectdetection[C]", *ProceedingsoftheIEEEconferenceoncomputervisionandpatternrecognition*,pp.2117-2125, 2017.
- [7] M Liu, Y Chen, J Xie et al., "LF-YOLO: A lighter andfasteryoloforwellddefectdetectionofX-rayimage[J]", *IEEE SensorsJournal*,vol.23,no.7,pp.7430-7439,2023.
- [8] S Woo, J Park, J Y Lee et al., "Cbam: Convolutionalblockattentionmodule[C]", *ProceedingsoftheEuropean conference on computer vision (ECCV)*, pp.3-19,2018.
- [9] X Ding, X Zhang, J Han et al., "Scaling up your kernelsto31×31:Revisitinglarge kernel design incnns[C]", *Proceedings of the IEEE/CVF conference oncomputervisionandpatternrecognition*,pp.11963-11975,2022.
- [10] C Li, L Li, H Jiang et al., "YOLOv6: A single-stage objectdetection framework for industrial applications[J]", *arXivpreprint*,2022.
- [11] X Long, K Deng, G Wang et al., "PP-YOLO: An effectiveand efficient implementation of object detector[J]", *arXivpreprint*,2020.
- [12] S. a. Y. Kumar, "A novel yolov3 algorithm-based deeplearning approach for waste segregation: Towards smartwastemanagement", *MDPI*, pp.14,2020.
- [13] Y.a.Z.Z.a.H.You,"Imagenettraininginminutes", *Proceedingsofthe47thInternationalConferenceon ParallelProcessing*, 2018.
- [14] S. a. C. J. a. A. L. a. B. J. Vicente, "Reconstructingpascalvoc", *ProceedingsoftheIEEEconferenceoncomputervision andpatternrecognition*,2014.
- [15] M. Shenoda, "Real-time Object Detection: YOLOv1 Re-ImplementationinPyTorch",*arXivpreprint*,pp.4,2023.
- [16] Z.A.a.S.H.a.A.Choudhry,"DarkNet-19BasedIntelligent Diagnostic System for OcularDiseases",*IranianJournalofScienceandTechnology*
- [17] J. a. F. A. Redmon, "YOLO9000: better faster stronger7263–7271", *ProceedingsoftheIEEEconferenceoncomputervision andpatternrecognition*,2017.
- [18] J.a.F.A.Redmon,"Yolov3:Anincrementalimprovement", *arXivpreprint*,pp.6,2018.