iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 12 **Issue:** II **Month of publication:** February 2024

DOI: https://doi.org/10.22214/ijraset.2024.58290

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue II Feb 2024- Available at www.ijraset.com

A Comparative Analysis of YOLOv8 and YOLOv5 for Nut Thread Classification - Deep Learning Approach

Sarthak Patil[1], Gaurav Dave[2], Shubham Urmode[3]
[1]*Department of Mechanical Engineering, Smt. Kashibai Navale College of Engineering (SKNCOE), Pune, India*
[2]*PHD Scholar, Sinhgad College of Engineering, Department of Mechanical Engineering, Assistant Professor, Smt. Kashibai Navale College Engineering (SKNCOE), Pune, India*
[3]*School of Mechanical Engineering MIT Academy of Engineering (D) Pune, India*

Abstract: ***This research provides valuable and compelling insights into the performance comparison of YOLOv8 and YOLOv5 for Thread and No thread detection. The findings underscore the significant impact of architectural improvements and model advancements, resulting in higher precision and accuracy in object detection tasks. Furthermore, the successful implementation of LabelImg and Beautiful Soup demonstrates their remarkable effectiveness in dataset annotation and collection, crucial contributors to the overall success of the study. As a result, these results lay a solid foundation for future advancements in object detection techniques, with wide-ranging applications across diverse domains, encompassing industrial automation, surveillance systems, and beyond.***

Keywords: Yolov8, Yolov5, Nut thread detection, Comparative Analysis, LabelImg, Beautiful Soup

I. INTRODUCTION

Traditional camera sensors heavily rely on human eyes for observation, which can lead to fatigue when continuously assessing objects of varying sizes in complex scenes. Additionally, human [1]cognition has its limitations, making judgment errors common and significantly reducing efficiency. To overcome these challenges, object recognition technology plays a vital role in camera sensors. This technology automates the process of judging an object's category, eliminating the need for [1] human intervention and enhancing efficiency and accuracy in complex visual environments. Threaded components play a vital role in many industries and are the main points of assembly, equipment and products. Ensuring their quality and accuracy during product inspection is of utmost importance to maintain high standards of quality control. To address this critical need, this paper presents a comprehensive comparative analysis of two state-of-the-art object detection models, YOLOv8 and YOLOv5, specifically focusing on their performance in Thread and No thread detection. This study aims to evaluate the effectiveness and precision of YOLOv8 and YOLOv5 in detecting threaded components, providing valuable insights into their respective capabilities. The dataset used in this study is carefully curated, combining data from camera and web sources to ensure diversity and representativeness. The annotation process, facilitated by the LabelImg tool, guarantees accurate labelling of the dataset with bounding boxes around the threads and no threads, laying the groundwork for robust model training. Additionally, to enhance the dataset collection process, this study utilized data scraping techniques such as Beautiful Soup and Selenium. These tools facilitated the extraction of relevant information from web sources, further enriching the dataset with diverse samples for comprehensive model training. The combination of LabelImg, Beautiful Soup, and selenium contributed to the success of the annotation and dataset collection processes, ensuring the dataset's accuracy and representation of real-world scenarios.

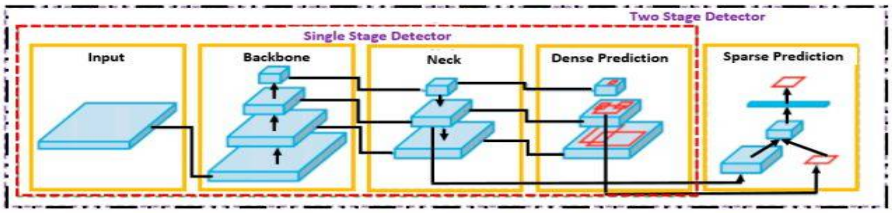


Figure 1: Object detector anatomy [2]

© IJRASET: All Rights are Reserved |

By incorporating these data collection and annotation methodologies, this research provides a robust foundation for the comparative analysis of YOLOv8 and YOLOv5's performance in thread and no thread detection, ultimately advancing the state-of-the-art in object detection and its applications in critical industries.

With a comprehensive understanding of the dataset collection, annotation, and model architectures, this research endeavours to present a thorough comparative analysis of YOLOv8 and YOLOv5's performance in thread and no thread detection. The findings of this study have significant implications for industries that heavily rely on threaded components and depend on precise object detection for maintaining quality control standards. By exploring the capabilities and advancements of these state-of-the-art object detection models, this research aims to contribute to the continuous improvement and application of object detection techniques across diverse domains, from industrial automation to quality inspection in various manufacturing processes.

II. DATA COLLECTION AND ANNOTATION

To circumvent barriers that hinder machine automation, human copy-paste is often employed as the most effective method [3]. However, this approach is not practical, particularly for large company projects, as it can be costly and time-consuming. As an alternative, text grepping using regular expressions is utilized to identify information that adheres to specific patterns. Web scraping techniques such as HTTP programming, DOM parsing, and HTML parsers are also commonly employed to extract and analyse data from [3] web pages. These methods offer more efficient and automated ways of gathering information, making them favourable for various data extraction tasks. To construct a comprehensive and robust dataset for thread and No thread detection, a two-fold approach was adopted. Firstly, images were gathered using cameras to capture real-world examples of threaded and nonthreaded nuts. This ensured that the dataset represented a wide range of scenarios and variations encountered in practical industrial applications. To further enhance the dataset's diversity, web scraping techniques were employed using tools like Beautiful Soup and Selenium. Beautiful Soup [4] is a Python library that utilizes an HTML/XML analytics engine to extract, analyse, and modify information within the DOM tree of webpages. Its concise DOM visitor interfaces enable developers to rapidly create system prototypes and collect experimental data. Furthermore, Beautiful Soup offers high cross-platform flexibility, making it a versatile and powerful tool for web scraping and parsing tasks. This involved automated extraction of images from various websites and online sources. By leveraging web scraping, the dataset was enriched with a broader array of samples, encompassing different nut types, sizes, textures, and lighting conditions, thus reflecting the variability encountered in manufacturing and assembly processes.

The combined dataset contained images of both threaded and non-threaded nuts, encompassing various backgrounds, orientations, and angles. This diversity provided a comprehensive set of data for training the object detection models, enabling them to handle different nut configurations with precision and accuracy. To prepare the dataset for model training, the images were annotated using the LabelImg tool. LabelImg facilitates swift and on-the-fly bounding box annotation and supports the handling of multiple object classes [5]. This annotation process involved manually creating bounding boxes around each thread and no thread region in the images. Each bounding box represented a specific threaded or non-threaded region, providing ground truth annotations to train the object detection models effectively. By incorporating both camera-captured images and web-scraped samples and annotating them using LabelImg, the resulting dataset became a valuable resource for training and evaluating the object detection models. The diverse nature of the dataset facilitated better generalization and robustness of the models, ensuring they could accurately detect threads and no threads in a wide range of real-world scenarios.



Figure 2: Annotation and Labelling in Labelimg

© IJRASET: All Rights are Reserved |

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue II Feb 2024- Available at www.ijraset.com

III. YOLO MODEL ARCHITECTURE

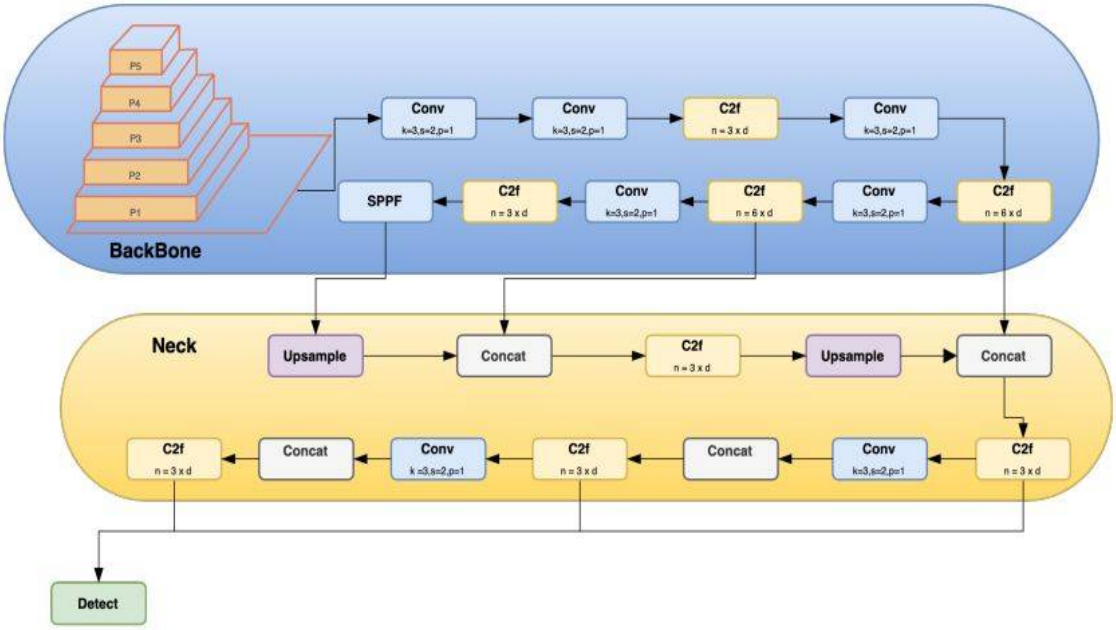


Figure 3: Yolov8 Model Architecture [6]

The YOLO series has seen several successors, including YOLOv4, YOLOv5, YOLOv6, and YOLOv7 [6]. In 2023, Ultralytics released YOLOv8, depicted in Figure 3. YOLOv8 introduces the C2f module, based on Cross Stage Partial (CSP), in its backbone architecture, different from YOLOv5's C3 Module. The CSP architecture enhances the model's learning capacity while reducing computational complexity. The C2f module [6]comprises two Conv Modules and n BottleNeck, connected through Split and Concat operations. The rest of the backbone architecture remains consistent with YOLOv5. Additionally, the SPPF Module is used at the final layer [6] of the backbone in YOLOv8.

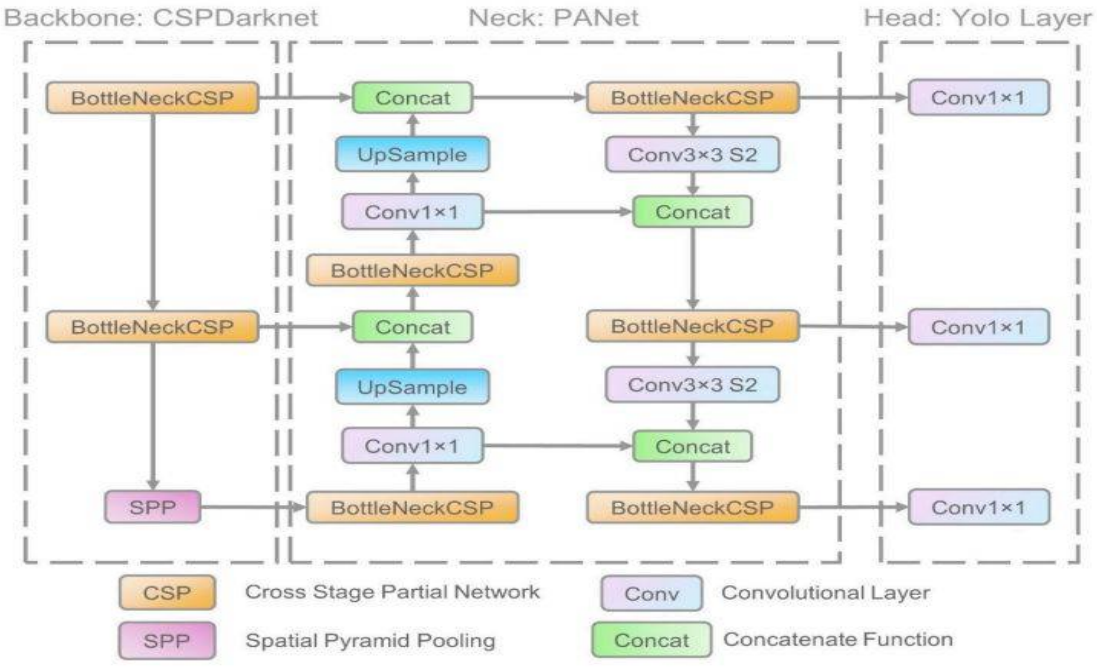


Figure 4: Yolov5 Model Architecture [7]

© IJRASET: All Rights are Reserved |

The YOLOv5 network [7] architecture comprises three essential components: (1) the Backbone, which utilizes CSPDarknet for feature extraction, (2) the Neck, which employs PANet for feature fusion, and (3) the Head, represented by the YOLO Layer. The input data undergoes a sequence of processing, starting with feature extraction using CSPDarknet, followed by feature fusion through PANet. The final step involves the YOLO Layer, which outputs the [7]detection results, including class labels, confidence scores, object locations, and sizes. This well-structured architecture enables YOLOv5 to effectively perform object detection tasks with high accuracy and efficiency.

In YOLOv5, the Model Backbone, which serves to extract crucial features from input images, utilizes Cross-stage Partial Networks (CSP) [8] to capture information-rich characteristics effectively. The utilization of CSPNet helps address the challenge of increased processing time in deeper networks. By evenly distributing computation across convolutional layers, CSPNet ensures that the arithmetic units remain efficiently utilized, avoiding computational bottlenecks. Moreover, this approach reduces memory costs by employing cross-channel pooling to compress feature maps during [8] the feature pyramid generation process, leading to a remarkable 75% reduction in memory usage for the object detector.

The Model Neck's primary function is to create feature pyramids, enabling the model to generalize well for objects of varying scales. Feature pyramids are particularly useful in identifying objects with different sizes, ensuring the model performs effectively on unseen data. By incorporating feature pyramids, YOLOv5 enhances its ability to handle diverse object scales and improves overall object [8] detection performance.

IV. EXPERIMENTAL SETUP

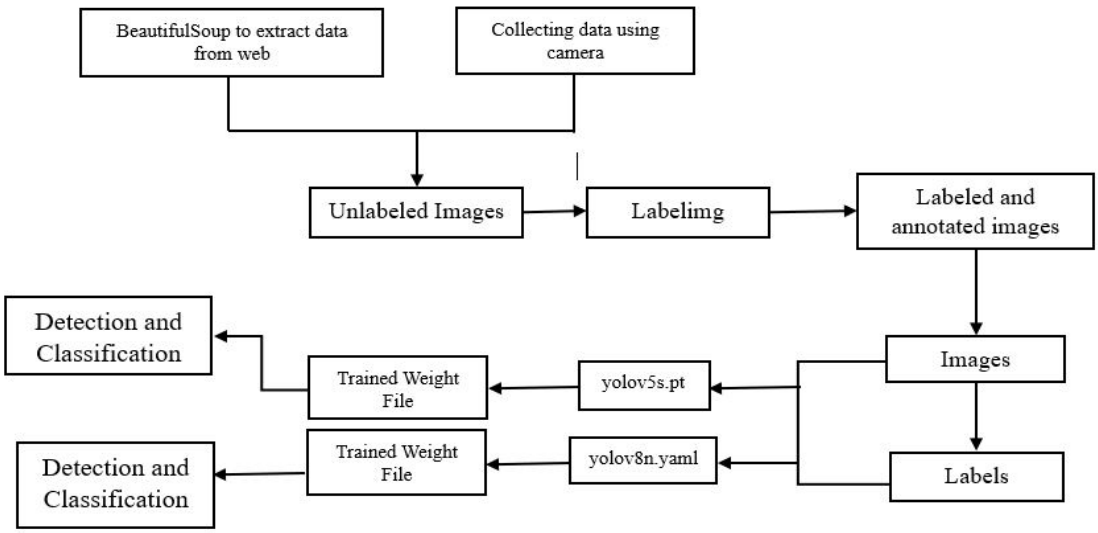


Figure 5: System Flowchart

The flowchart represents the step-by-step process of collecting data, annotating it, training two object detection models (YOLOv5s.pt and YOLOv8n.yaml), and finally evaluating their respective performance in detection and classification tasks. Here's a summary of the flowchart steps:

1) *Data Collection:* Data is collected using two methods - web scraping with Beautiful Soup and manual capture with a camera. This process gathers a diverse set of images, both labelled and unlabelled.
2) *Annotation:* The collected images are annotated using LabelImg, a graphical image annotation tool. Annotations involve drawing bounding boxes around the regions of interest, such as threads and no threads, to create labelled and annotated images.
3) *Data Split:* The labelled and annotated images are divided into two separate files: images and labels. This division facilitates data preparation for model training.
4) *Model Training:* The two object detection models, YOLOv5s.pt and YOLOv8n.yaml, are trained on the labelled and annotated images. The training process adjusts the models' parameters to learn and recognize patterns that distinguish between threads and no threads.

© IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

5) *Model Weights:* After completing the training, the output is two trained weights files, one for each model (YOLOv5s.pt and YOLOv8n.yaml). These files contain the learned information, enabling the models to perform object detection tasks.
6) *Detection and Classification:* The trained models, YOLOv5s.pt and YOLOv8n.yaml, are then employed to perform detection and classification on new, unseen images. The models utilize their respective weights files to predict and distinguish between threads and no threads in these images.
7) *Comparison:* The results obtained from the two models are compared, evaluating their respective detection and classification performance. This comparison provides insights into the strengths and weaknesses of each model and helps in selecting the most suitable model for the specific thread detection application.

Overall, this flowchart outlines a comprehensive approach to collecting data, training object detection models, and evaluating their effectiveness in thread detection and classification tasks. The process involves multiple steps, from data gathering and annotation to model training and inference, culminating in a thorough assessment of model performance.

V. RESULTS

Precision measures the model's capability to correctly identify only the relevant objects, represented as the [9] percentage of accurate positive predictions out of all positive predictions made by the model. On the other hand, recall gauges the model's ability to locate all the relevant cases, indicating the percentage of correct positive predictions among all the actual ground-truth bounding boxes [9]. In real-world scenarios, object detection tasks often involve multiple classes, denoted as K, instead of just one. To account for this, Mean Average Precision (mAP) is introduced, which represents the mean value of AP across all K classes. mAP provides a comprehensive performance assessment by considering the precision-recall trade-off for each class, resulting in a single metric that summarizes the overall detection accuracy across all object classes.

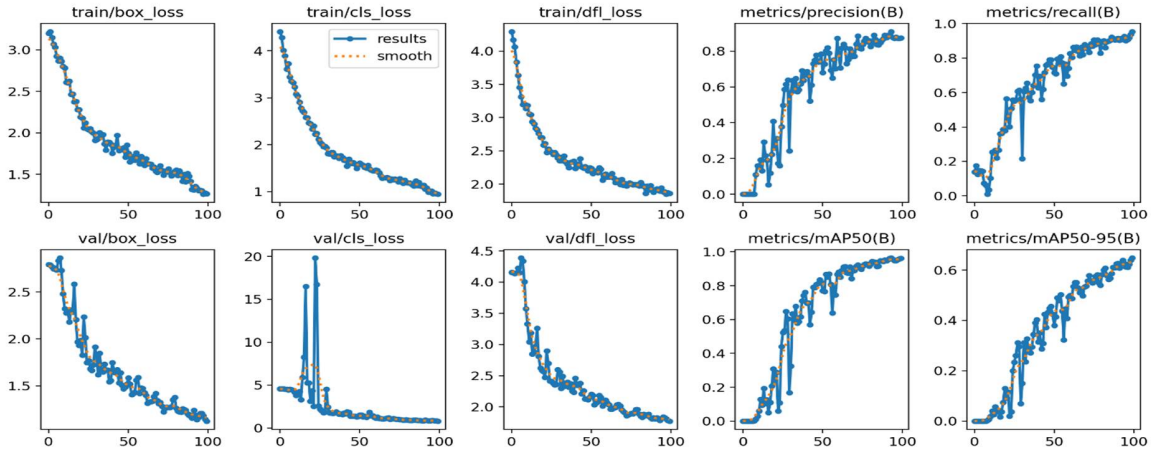


Figure 6: Yolov5 training result

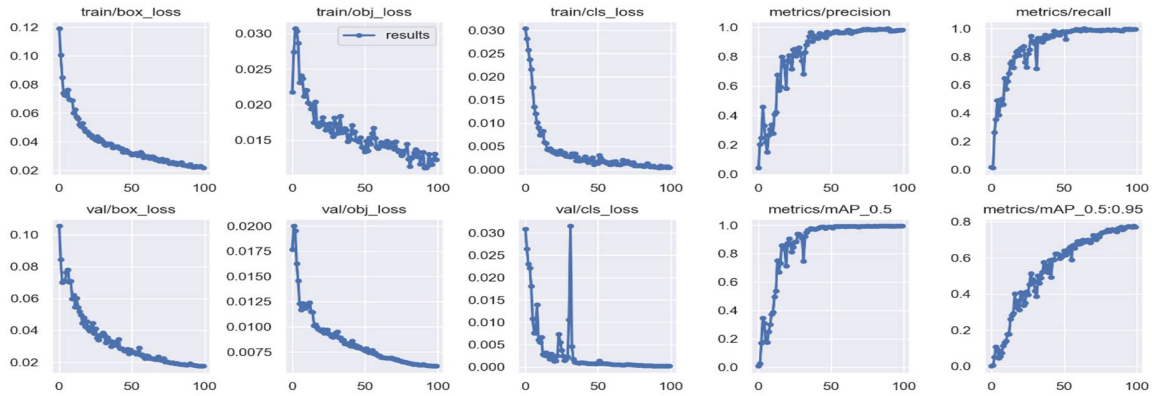


Figure 7: Yolov8 training result

© IJRASET: All Rights are Reserved |

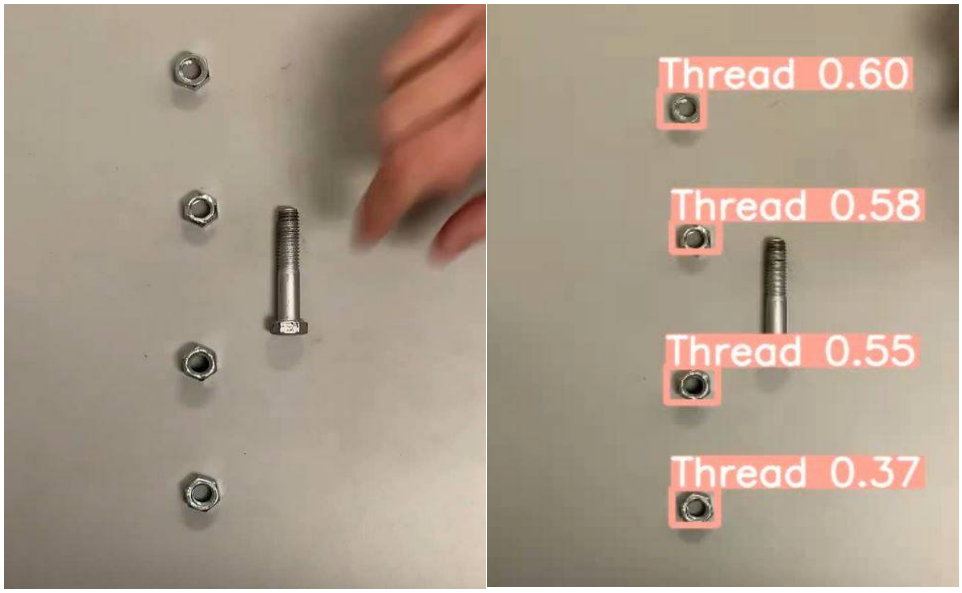
The evaluation of YOLO v8's performance encompasses various essential metrics, including box loss, [10]classification loss, objectness (differentiable focal) loss, precision, and recall. These metrics are measured on both the training and validation data to assess the model's capabilities in different scenarios. By comparing the model's performance, valuable insights into its generalization ability [10] are obtained.

The model demonstrates robustness as it does not overfit; it performs consistently well on both the training and validation data. Moreover, the low box loss on both sets indicates accurate bounding box predictions, further confirming the model's effectiveness in object detection tasks.

The evaluation metrics for both models are presented, including mAP at 50% IoU and mAP from 50% to 95% IoU. YOLOv8 achieved a respectable mAP of 0.96 at 50% IoU and 0.64 from 50% to 95% IoU. However, YOLOv5 significantly outperformed YOLOv8 with a higher mAP of 0.994 at 50% IoU and 0.77 from 50% to 95% IoU. Visualizations of model predictions on sample images and videos are provided for both models.

VI. COMPARATIVE ANALYSIS

A comprehensive comparative analysis is presented, highlighting the strengths and weaknesses of YOLOv8 and YOLOv5 in the context of thread no thread detection. The higher precision achieved by YOLOv5 is discussed, along with factors contributing to the performance differences.



Yolov8 Yolov5

As observed from the aforementioned figure, YOLOv8 demonstrated limitations in accurately predicting intricate details of threads in the image. The model struggled to capture fine-grained features, resulting in less precise thread detection compared to YOLOv5. Despite achieving a reasonable mean Average Precision (mAP) at 50% Intersection over Union (IoU), YOLOv8's performance suffered significantly when evaluated at higher IoU thresholds (from 50% to 95%). On the other hand, YOLOv5 exhibited superior capabilities in capturing fine detailed threads with higher accuracy and achieved a notably higher mAP across various IoU thresholds.

Furthermore, considering the critical importance of thread accuracy in industries relying on threaded components, the findings highlight the significance of continued research and improvement in object detection models like YOLOv8 to meet the stringent requirements of quality control in manufacturing and assembly processes. By addressing the challenges observed in thread detection, advancements in YOLOv8 and similar models can contribute to more efficient and reliable quality inspection procedures across various industrial applications.

The performance of YOLOv8 suffered due to a low mean average precision (mAP), resulting in less accurate and weaker object detection and classification compared to previous versions.

© IJRASET: All Rights are Reserved |

VII. APPLICATIONS AND IMPLICATIONS

Thread detection plays a crucial role in maintaining high standards of quality control in industries. By ensuring the precision and accuracy of threaded components, manufacturers can identify and eliminate defective or faulty products, reducing the risk of product recalls and customer complaints.

Thread detection helps verify the integrity of threaded connections in mechanical assemblies and equipment. It ensures that threaded components are properly aligned and securely fastened, preventing potential issues like leaks, loose connections, or structural failures.

In industries where, threaded components are widely used, such as automotive, aerospace, and construction, accurate thread detection streamlines production processes. This leads to improved efficiency, as faulty components can be identified early in the manufacturing cycle. Moreover, ensuring the integrity of threaded connections enhances overall safety in the operation of machinery and equipment.

The implications of the results are discussed, particularly in the context of industrial product inspection. The higher precision of YOLOv5 makes it more suitable for detecting threaded components, leading to improved quality control and customer satisfaction.

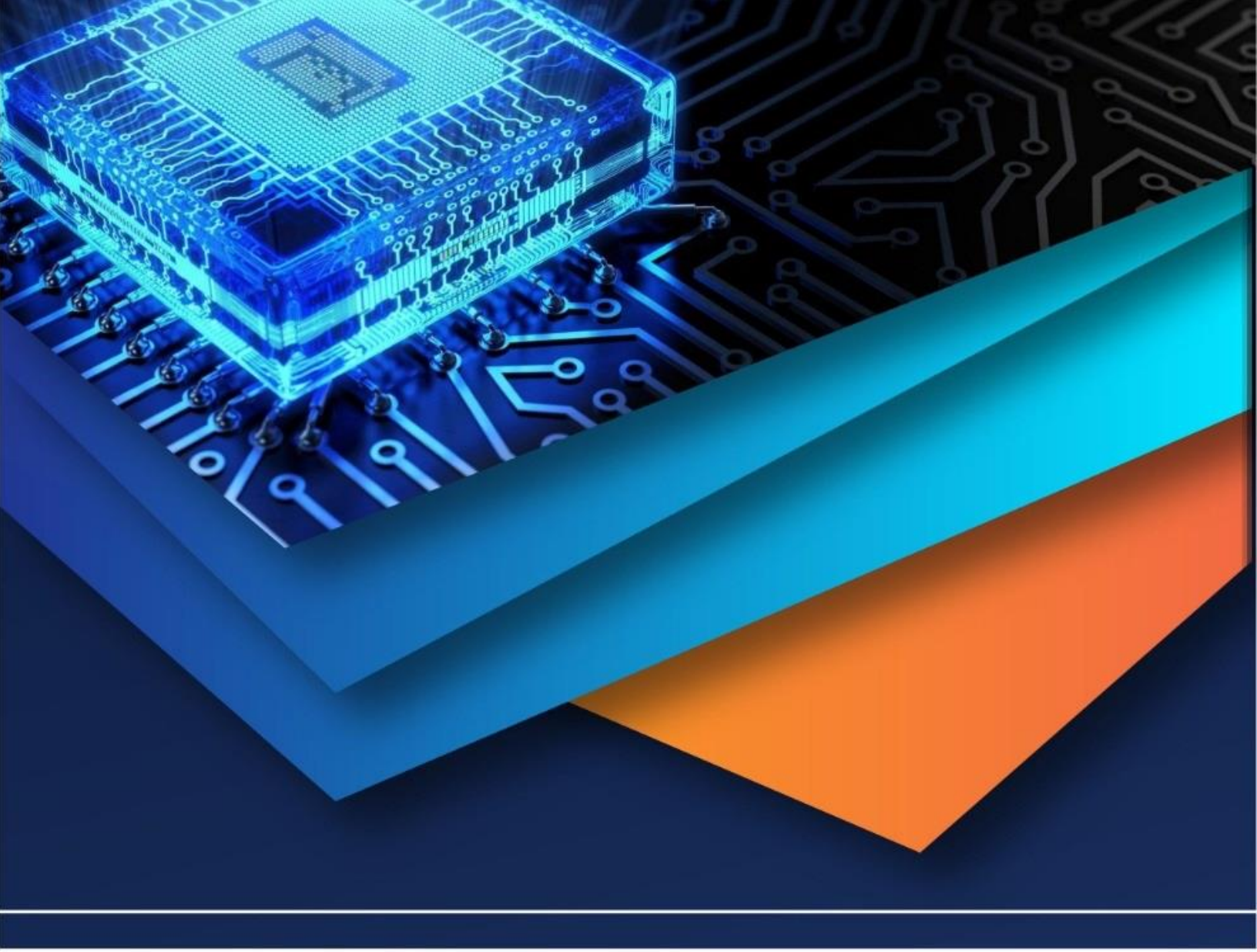
VIII. CONCLUSION

This research highlights the effectiveness of YOLOv5 over YOLOv8 in Thread and No thread detection in images and videos. The comparative analysis demonstrates that YOLOv5 achieved significantly higher mean Average Precision (mAP) scores at various Intersection over Union (IoU) thresholds, indicating its superior accuracy and precision in identifying Threads and No threads. The utilization of LabelImg and Beautiful Soup for dataset collection and annotation further strengthens the study's reliability and real-world applicability. These findings have practical implications for industries reliant on threaded components, emphasizing the importance of advanced object detection models like YOLOv5 to ensure quality control and efficient product inspection processes

REFERENCES

[1] D. J. G. H. L. J. G. L. B. a. H. C. Haitong Lou, “DC-YOLOv8: Small-Size Object Detection Algorithm Based on Camera Sensor,” electronics, 2023.
[2] M. Hussain, “YOLO-v1 to YOLO-v8, the Rise of YOLO and Its Complementary Nature toward Digital Manufacturing and Industrial Defect Detection,” Machines 2023, p. 25, 2023.
[3] M. U. Eloisa Vargiu, “Exploiting web scraping in a collaborative filtering-based approach to web advertising,” Artif. Intell. Res., vol. 2, no. 1, pp. 44-54, 2013.
[4] G. H. Z. P. Chunmei Zheng, “A Study of Web Information Extraction Technology Based on Beautiful Soup,” Journal of Computers, 2015.
[5] C. J. &. P. Z. Christoph Sager, “A survey of image labelling for computer vision applications,” Journal of Business Analytics, 2021.
[6] S. B. M. P. P. a. R. C. Nabin Sharma, “Parking Time Violation Tracking Using YOLOv8 and Tracking Algorithms,” Sensors, pp. 6-7, 2023.
[7] H. L. K. L. L. C. a. Y. L. Renjie Xu, “A Forest Fire Detection System Based on Ensemble Learning,” Forests, p. 17, 2021.
[8] M. I. A. M. Q. A. A. O. A. M. M. A. a. M. H. Abdul Hanan Ashraf, “Weapons Detection for Security and Video Surveillance Using CNN and YOLO-V5s,” Computers, Materials & Continua, pp. 5-6, 2022.
[9] R. &. N. S. &. d. S. E. Padilla, “A Survey on Performance Metrics for Object-Detection Algorithms,” in Conference: 2020 International Conference on Systems, Signals and Image Processing (IWSSIP), Brazil, 2020.
[10] M. Y. I. M. S. U. M. M. H. P. Mohammad Sojon Beg1, “Enhancing Driving Assistance System with YOLO V8-Based Normal Visual Camera Sensor,” Journal of Advanced Research in Applied Sciences and Engineering Technology, 2023.

© IJRASET: All Rights are Reserved |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)