Comparative Analysis of 2D-CAD Comparison with Custom-Trained Computer Vision Models

Roshni Sundrani[1], Shashikant V. Athawale[2]

[1]Student, M.E. Artificial Intelligence and Data Science, AISSMS COE, Pune, Maharashtra, India, roshni.sundrani83@gmail.com
[2]Associate Professor, Department of Computer Engineering, AISSMS COE, Pune, Maharashtra, India, svathawale@aissmscoe.com

Abstract: In manufacturing environments, the ability to efficiently compare and analyze 2D Computer-Aided Design (CAD) drawings is critical for ensuring product quality, minimizing errors, and streamlining the design iteration process. Traditional manual comparison methods are time-intensive and prone to human errors, particularly when analyzing annotations, dimensions, and complex structural details. To address these challenges, this study presents a comparative analysis of state-of-the-art deep learning models—YOLOv8m, YOLOv8x, YOLO-NAS, and Faster R-CNN—for automated CAD design evaluation. The models were trained on a dataset of 5,000 CAD images, encompassing diverse mechanical components with varying complexities. The proposed system leverages object detection and Optical Character Recognition (OCR) techniques to extract and compare dimensions and notes with high precision. Experimental results demonstrate that YOLOv8m outperforms other models in terms of detection accuracy. The findings highlight the effectiveness of deep learning-based CAD comparison systems in reducing verification time and improving design evaluation accuracy. This study provides insights into the strengths and limitations of different computer vision models for CAD analysis, contributing to the advancement of automated design validation in manufacturing industries. This approach enhances the recognition accuracy of deep learning models, making dimension extraction and recognition more practical. The system achieves 90-95% accuracy in detecting dimensions and notes in the CAD designs, and 85-90% accuracy in text recognition. These findings highlight the effectiveness of AI-driven CAD comparison systems in reducing verification time and improving design evaluation accuracy, contributing to the advancement of automated design validation in manufacturing industries.

Keywords: Artificial Intelligence algorithms, machine learning, computer vision, custom training object detection models, YOLOv8m, YOLOv8x, YOLO-NAS, Faster R-CNN, CAD compare, PyTesseract, Paddle OCR.

1. Introduction

In modern manufacturing industries, the accuracy of 2D Computer-Aided Design (CAD) drawings is critical for ensuring product quality and design precision. Companies like ABC Filters Pvt. Ltd., a leading manufacturer of vehicle filtration systems, heavily rely on CAD designs to communicate product specifications with clients. Their design workflow involves creating initial 2D CAD drawings, sending them for client feedback, incorporating revisions, and manually comparing the updated design with the original before finalizing it for production. This manual comparison process is time-consuming, prone to human errors, and lacks a structured digital reference for future validation. Each CAD design may contain 60-80 dimensions, including tolerance values, special symbols (e.g., ϕ, ±), and annotations that require precise verification. Currently, these differences are analyzed manually by printing the drawings, highlighting discrepancies, and reviewing them before manufacturing. This process is not only inefficient but also increases the risk of undetected errors, leading to potential defects, rework, and financial losses. Given the high complexity and variability in mechanical designs, an automated AI-driven solution is essential to enhance accuracy, streamline comparisons, and digitize the verification process.

This research introduces a deep learning-based CAD comparison system that automates the extraction and analysis of dimensions and annotations using state-of-the-art object detection models. The proposed system leverages YOLOv8m, YOLOv8x, YOLO-NAS, and Faster R-CNN to identify critical design elements and detect differences between revised and original CAD drawings. Optical Character Recognition (OCR) further extracts textual information, including special symbols, and stores the results in an Excel sheet, providing a structured, digital reference for future comparisons. To ensure the system's effectiveness, 5,000 CAD images from ABC's real-world manufacturing data were used to train the models. Experimental results show that YOLOv8m outperforms other models in terms of precision and speed, making it the most suitable choice for dimension extraction in CAD drawings. By integrating AI-based comparison techniques, this research aims to eliminate manual errors, improve verification efficiency, and provide a scalable solution for industries relying on CAD-based workflows.

2. LITERATURE REVIEW

The evolution of object detection algorithms has significantly impacted various applications, including text detection and recognition in images. Early methods predominantly relied on manual feature extraction techniques. For instance, Wan et al.[1] utilized the Maximally Stable Extremal Regions (MSER) algorithm to localize text features, effectively extracting text positions even under image rotation and affine transformations. Yu et al. [2] designed a hierarchical localization method combining rule-based filtering and Support Vector Machine (SVM) classifiers. This approach extracts characters using MSER, filters out non-text regions based on text features, and locates text positions using Histogram of Oriented Gradient (HOG) features and bounding rectangles through SVM.

The advent of deep learning has revolutionized image processing, particularly in object detection. Convolutional Neural Networks (CNNs) have been pivotal in this transformation. LeCun et al. [3] proposed the LeNet-5 network model for character recognition, training a multi-layer neural network using the back-propagation algorithm, which became a prominent example of gradient learning. Krizhevsky et al. [4] introduced AlexNet, which increased network depth and enhanced representation capabilities by successfully implementing the Rectified Linear Unit (ReLU) activation function and the dropout technique, addressing issues like the vanishing gradient problem and overfitting. Simonyan et al. [5] developed VGGNet, replacing larger convolutions with consecutive 3×3 convolutions, significantly reducing the number of parameters while maintaining performance, enabling deeper networks.

Region-based Convolutional Neural Networks (R-CNN) along with its enhanced versions—Fast R-CNN and Faster R-CNN [6]—have been instrumental in improving detection accuracy. These methods rely on a two-stage approach, wherein region proposals are first generated and then classified using convolutional networks. Similarly, the Single-Shot MultiBox Detector (SSD) streamlines object detection by eliminating the need for a separate region proposal stage, akin to YOLO's efficiency-focused approach. Other significant advancements include Mask R-CNN [7], which extends object detection to instance segmentation, and models like RetinaNet [8] and EfficientDet, which introduce novel techniques for improving detection performance while balancing computational efficiency. Each of these models presents unique trade-offs between speed, accuracy, and complexity, making them suitable for different application requirements.

In the realm of object detection, the YOLO (You Only Look Once) framework has been particularly influential. Redmon et al. [8] introduced YOLO, a fast and accurate object detection framework that uses regression methods to globally predict objects in images, ensuring high precision in detection. Subsequent iterations, such as YOLOv8, have optimized model architecture and training processes to reduce parameters and computations, surpassing known real-time object detectors in terms of speed and accuracy. Faster R-CNN, proposed by Ren et al., is another significant development in object detection, combining region proposal networks with CNNs to achieve high accuracy in object detection tasks. Comparative analyses have shown that YOLOv8 outperforms Faster R-CNN in terms of both accuracy and speed, making it a preferred choice for real-time applications. Additionally, YOLO-NAS has

emerged as a cutting-edge deep learning architecture known for its adaptability and computational efficiency, further enhancing object detection capabilities.
Optical Character Recognition (OCR) has also seen varied applications across multiple domains, including invoice imaging, the legal industry, banking, and healthcare. OCR is widely used in fields like CAPTCHA, institutional repositories and digital libraries, optical music recognition, automatic number plate recognition, and handwritten recognition.
In summary, the development of neural networks has significantly enhanced methods for graphic feature recognition, improving the recognition capabilities of mechanical engineering drawings. By training on large datasets, machines can automatically analyze and find optimal solutions. However, machine conclusions are not necessarily 100% correct. To improve recognition accuracy, besides increasing training data, image processing and adjustments to deep network model architectures may help solve the problem of poor efficiency in automatic recognition. By training the system to read corresponding feature information from graphics, it can accurately extract dimensions and notes, thereby achieving automated drawing recognition. By custom training pyTesseract with jTessBoxEditor , higher OCR accuracy can be achieved. Paddle OCR has good accuracy for dimensions over other OCR as it can read tolerance and phi symbols as compared to AWS Textract.

3. REAL-TIME OBJECT DETECTION: A FOCUS ON THE YOLO FRAMEWORK

Real-time object detection plays a crucial role in various applications, including autonomous vehicles, robotics, video surveillance, augmented reality. Among the numerous object detection algorithms developed, the YOLO (You Only Look Once) framework has gained prominence due to its exceptional balance between speed and accuracy. Its ability to rapidly and efficiently identify objects in images has made it a preferred choice for real-time applications. Since its introduction, YOLO has undergone multiple advancements, with each new iteration refining its capabilities, addressing previous limitations, and improving performance.
The YOLO framework was first introduced by Redmon et al. in 2016, revolutionizing object detection by treating it as a single regression problem rather than relying on region proposal networks like Faster R-CNN. As discussed by Juan T, Diana-Margarita C [9] YOLOv1 and YOLOv2 are trained on PASCAL VOC 2007 and VOC 2012 dataset. From yolov3 onwards, Microsoft COCO (Common Objects in Context) used for training and benchmarking. The original YOLOv1 model demonstrated real-time processing but struggled with small object detection and localization errors. Subsequent versions introduced significant improvements:
-YOLOv2 introduced batch normalization, anchor boxes, and multi-scale training, enhancing accuracy.
-YOLOv3 [10] adopted a deeper architecture with Darknet-53 and multi-scale feature fusion.
-YOLOv4 [11] optimized computational efficiency while improving detection accuracy.
- YOLOv5 [12] state-of-the-art (SOTA) real-time instance segmentation model, optimized for applications requiring precise object localization such as medical image analysis, industrial defect detection and autonomous driving.
- YOLOv6 [13] is a single-stage object detection framework with balance between efficiency and accuracy. Its architectural improvements include RepVGG-based backbone and SimOTA (Simplified Optimal Transport Assignment) label assignment, enhancing detection performance while maintaining low latency for real-time applications.
-In YOLOv7 [14], several architectural enhancements were introduced, including the Extended Efficient Layer Aggregation Network (E-ELAN), improved model scaling, and trainable bag-of-freebies techniques, which enhance accuracy without compromising detection speed
-YOLOv8 [15] introduced advancements in model architecture, including anchor-free detection, improved backbone and neck structures, and adaptive computation for enhanced speed and accuracy. These refinements allow YOLOv8 to outperform previous versions in real-time object detection tasks while maintaining efficiency across various applications.
The efficiency of YOLOv8 in handling real-time object detection makes it a compelling choice for CAD design comparison, where identifying dimensions, annotations, and design variations is critical.

4. APPLICATION OF YOLO IN CAD DESIGN ANALYSIS

The YOLO (You Only Look Once) framework has demonstrated remarkable versatility across numerous real-world applications, including video sequences in surveillance [16], quickly identifying vehicles and license plate recognition [17], traffic sign recognition [18], YOLO based UAV technology for object detection from Drones [19], YOLO based pests and disease detection for precision farming [20], face mask and social distancing violation detection system [21]. In the medical domain, YOLO has been utilized for applications such as breast cancer diagnosis using YOLO-based multiscale parallel CNN [22,23] and pill identification [24], contributing to enhanced diagnostic precision and more efficient treatment workflows. Given its ability to accurately detect and localize objects of interest in complex environments, YOLO has proven to be an effective solution for various computer vision tasks. Motivated by its success in these diverse domains, this study explores the feasibility of utilizing YOLO for identifying dimensions and textual annotations in CAD designs. Unlike traditional CAD analysis methods, which may struggle with the variability and complexity of engineering drawings, YOLO's object detection capabilities enable it to focus specifically on elements of interest while effectively ignoring irrelevant components such as drawing lines, tables, bill of materials, boundary numbering, and other structural details.

Through training, YOLOv8 effectively distinguished the dimensions and notes in CAD images, demonstrating its adaptability in recognizing key design elements within highly detailed technical drawings. The results indicate that once trained on relevant features, YOLO can successfully detect and extract critical information from CAD designs, highlighting its potential for automating engineering drawing analysis.

5. CUSTOM TRAINED AI - VISION MODELS USED IN THIS STUDY

To evaluate the suitability of the YOLO framework for CAD design comparison, this study tested multiple object detection models, each offering unique advantages in terms of accuracy, speed, and efficiency. The selected models include:

1. YOLOv8m (Medium Model)

YOLOv8m represents a balanced approach within the YOLOv8 family, offering a good tradeoff between speed and accuracy. It features improvements in backbone architecture, decoupled detection heads, and advanced anchor-free mechanisms, making it highly efficient for complex object detection tasks. Given the intricate details in CAD images, YOLOv8m was chosen to assess its ability to detect design elements effectively.

2. YOLOv8x (Extra-Large Model)

YOLOv8x is a larger variant with increased model capacity, deeper layers, and higher computational power. While it improves accuracy compared to smaller YOLOv8 versions, it comes with a tradeoff in speed. This study included YOLOv8x to determine whether higher accuracy justifies the additional computational cost for CAD analysis.

3. YOLO-NAS (Neural Architecture Search)

YOLO-NAS is a recent advancement in the YOLO family that leverages automated architecture optimization. It optimizes performance while maintaining high detection speeds, making it suitable for real-time applications.

4. Faster R-CNN (Region based -Convolution Neural Network)

Unlike YOLO, Faster R-CNN follows a two-stage detection pipeline, first generating region proposals and then classifying objects. While it is known for high accuracy in general object detection, its slower inference time makes it less ideal for real-time applications. By including Faster R-CNN, this study benchmarks how a two-stage detector compares to YOLO models when applied to CAD designs.

In this study, we evaluated multiple variants of the YOLO framework to determine the most effective model for CAD design comparison. The selected variants—YOLOv8m, YOLOv8x, and YOLO-NAS—were chosen based on their performance trade-offs between accuracy, speed, and computational efficiency. Additionally, we compared these models with Faster R-CNN, a widely used two-stage object detection framework, to assess its suitability for CAD analysis.

YOLOv8m vs. YOLOv8x: Both models share the same underlying architecture, but YOLOv8x has a deeper network with more parameters. While YOLOv8x is expected to provide marginally higher accuracy, it comes at the cost of increased computational requirements and inference time.

YOLOv8 vs. YOLO-NAS : YOLO-NAS employs Neural Architecture Search (NAS) to optimize detection performance, particularly for edge devices.

In the study by Yogesh and Pankaj (2024) [25], YOLOv8 demonstrated superior performance in agricultural applications, achieving a higher mean Average Precision (mAP) of 84.0% compared to YOLO-NAS's 81.8%. Our evaluation showed that YOLO-NAS had a lower precision (~60-70%) based on the extracted results, making it less effective for CAD object detection.

YOLOv8 vs. Faster R-CNN : Unlike YOLO’s single-shot detection mechanism, Faster R-CNN follows a two-stage approach, which generally improves accuracy but significantly slows down inference. In the study by Ezzeddini L, Ktari J [26], focusing on real-time detection of fishing vessels and fish demonstrated that YOLOv8 outperformed Faster R-CNN in both accuracy and processing speed, making it more suitable for real-time maritime surveillance tasks. Another research [27] evaluating traffic object detection systems found that YOLOv8 provided more accurate and efficient detection and classification of traffic objects in real-time environments compared to Faster R-CNN. Given the need for complex processing in CAD comparisons, Faster R-CNN was less suitable for our use case.

Based on our experiments, YOLOv8m outperformed all other models, striking the best balance between accuracy and efficiency. The evaluation metrics, including precision-recall curves, F1-score, and confusion matrices, confirm that YOLOv8m delivers superior results for CAD design comparison, making it the most viable choice for this application.

6. PERFORMANCE EVALUATION

1. YOLOv8x Performance Evaluation:

The dataset used for training and validating the AI-powered Dimensions detection system was collected from ABC Filters – around 5000 images were utilized for training our ML model.

The dataset collected is having 4500 images for Training Dataset; 500 images for Validation Dataset; With 35000 instances of annotated Dimensions and 5000 Notes for training YOLOv8x model with custom dataset.

Following is the results metrices after training ultralytics yolov8x on 5000 CAD images for 100 epochs with GPU parallel processing (Ultralytics YOLOv8.2.5 Python-3.11.5 torch-2.3.1+cu118 CUDA NVIDIA GeForce GTX 1650, 4096MiB). By implementing parallel processing and GPU acceleration inference time can reduce significantly, making the system even more suitable for high-volume CAD comparison tasks in large-scale engineering projects.

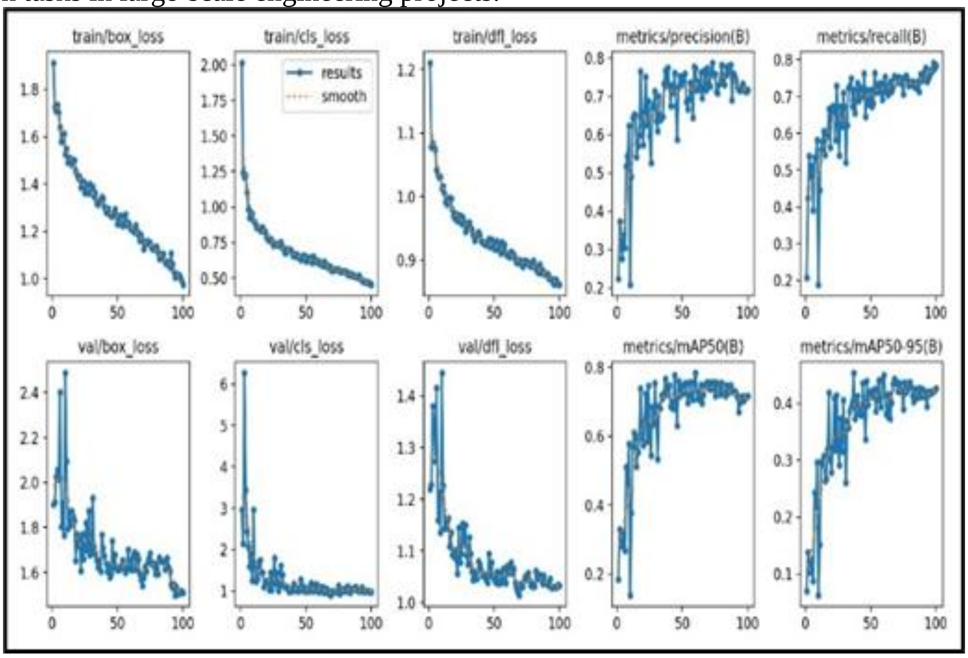

Figure 1. YOLOv8x results metrics after 100 epochs.

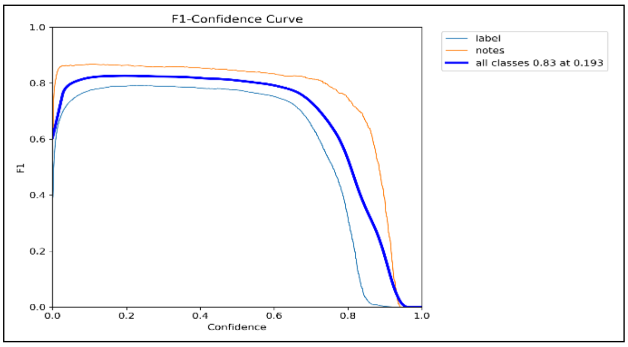

Figure 2. F1- Confidence curve

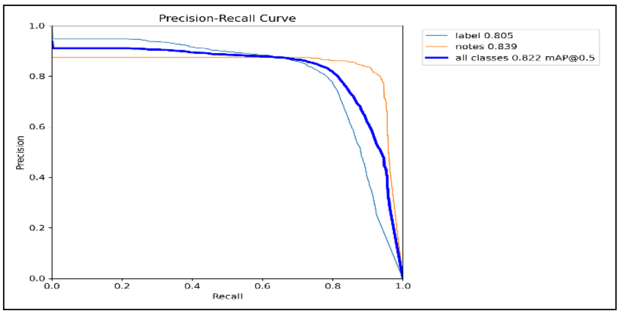

Figure 3. Precision- Recall Curve

From the above results metrics, we observe that the model is performing well on custom object detection. The loss curve is inversely proportional with epochs and Precision is increasing reaching to 80% for all classes.

Confusion matrix shows that the 77.3 % accuracy for detecting Labels (Dimensions) and 78.4% accuracy for correctly predicting notes . F1 Confidence curve is the harmonic mean of precision and recall. Mean Average precision (mAP) measure of the accuracy of object localization and classification simultaneously.

2. YOLOv8m Performance Evaluation:

Following are the results metrics of yolov8m trained on 2D-CAD designs for 100 epochs with same system configuration as yolov8m. As compared to yolov8x , the accuracy of yolov8m is higher, mean Average precision is 0.875 and precision and recall are reaching 0.90. The model has shown accurate results on unseen data.

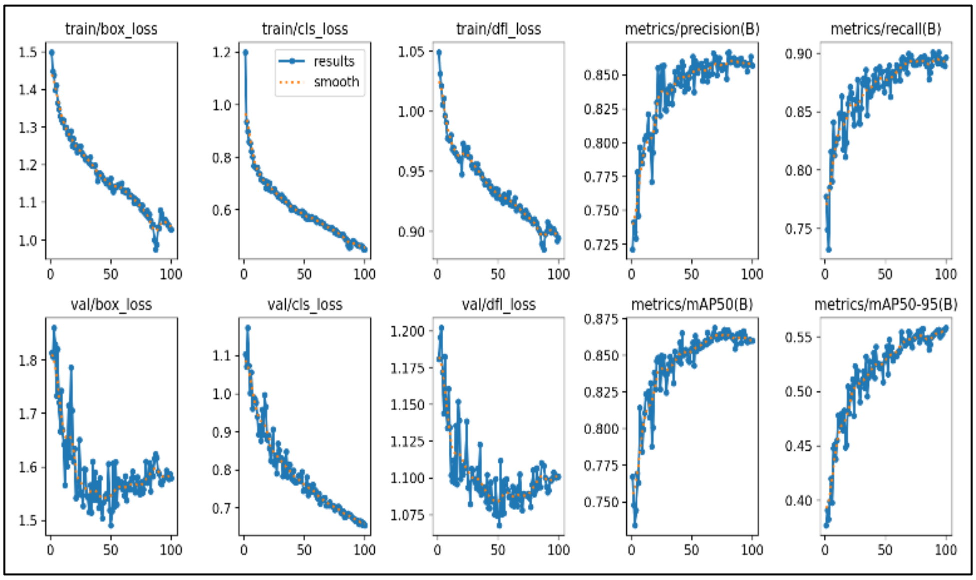

Figure. 4. Results metrics after 100 epochs of YOLOv8m

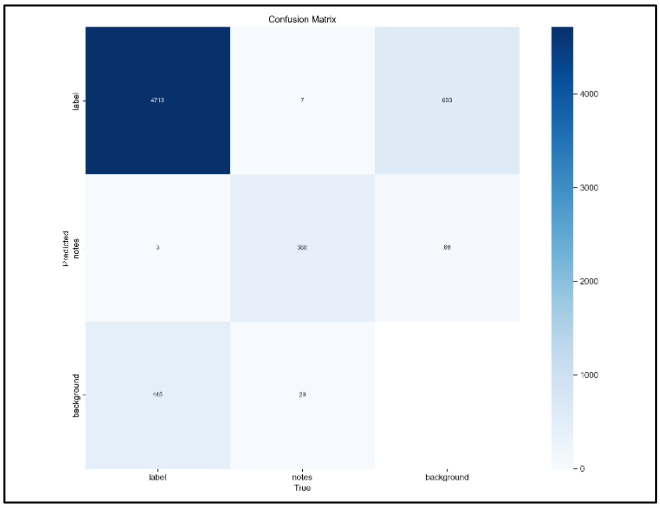

Figure 5. Confusion matrix for Labels (Dimensions) and Notes predicted after 100 epochs of yolov8m

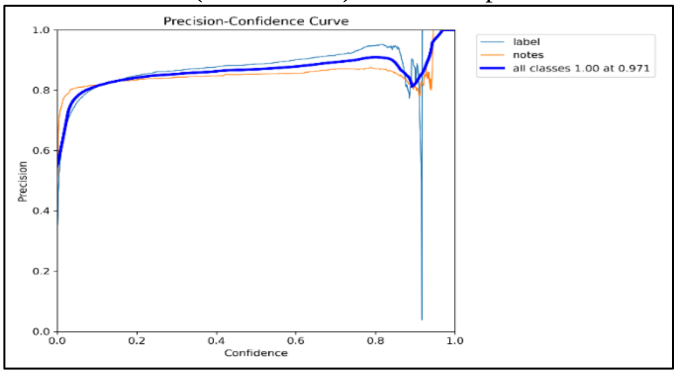

Figure 6. Precision Curve showing all classes 1.00 at 0.971

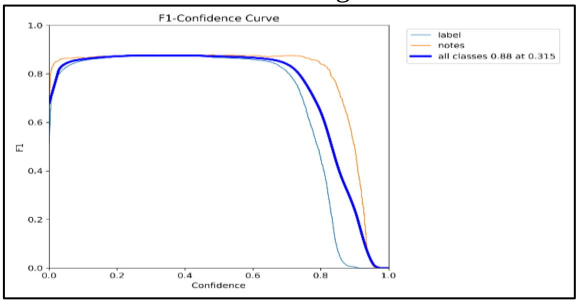

Figure 7. F1- Confidence curve

Table 1. Performance comparison of various models

Model	mAP50	Precision	Recall
YOLOv8m	0.875	0.90	0.90
YOLOv8x	0.785	0.80	0.80
YOLO-NAS	0.654	0.65	0.65
Faster-RCNN	0.41	0.55	0.55

7. SYSTEM ARCHITECTURE

The CAD Comparison System is designed to automate the process of analyzing CAD designs by leveraging object detection and OCR techniques. The architecture consists of multiple interconnected stages that facilitate CAD file processing, object detection, text recognition, and comparative analysis.

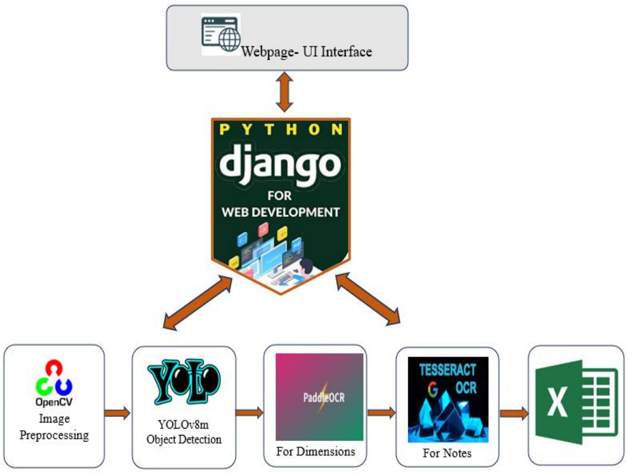

Figure 8. System Overview

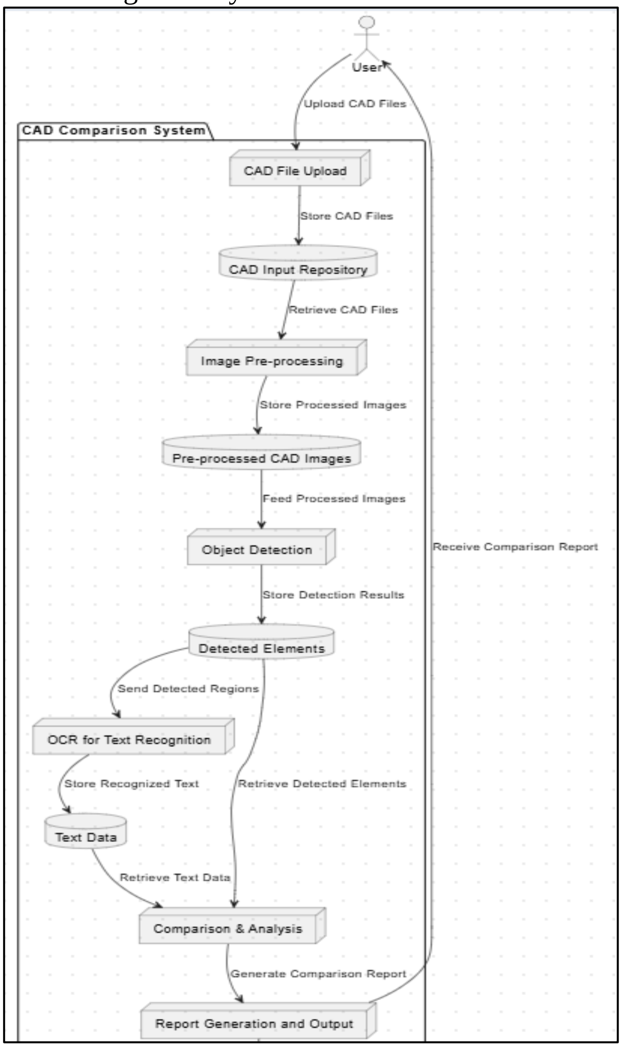

Figure 9. System Architecture of 2D-CAD Comparison

1. User Interaction Layer
The system begins with the user uploading CAD files, which serve as input for further processing. Users receive a final comparison report after the analysis is completed.
2. Data Processing Layer
CAD File Upload: Users submit CAD files, which are stored in the system's repository. The PDFs are uploaded and first converted to .jpeg files using poppler library. The jpeg files are then further processed.
CAD Input Repository: The system retrieves stored CAD files for further processing.

Image Pre-processing: CAD images undergo enhancement techniques such as binarization, noise removal, edge detection and resize to improve object detection accuracy.
3. Object Detection Layer
A deep learning-based YOLOv8m object detection model is utilized to automatically detect and classify various geometric elements, standard symbols, and textual annotations within the CAD drawings. This model is chosen for its balance of speed and accuracy, making it suitable for real-time applications.
Detected Elements: Once identified, elements such as circles, lines, arrows, symbols, and annotations are stored for subsequent processing. These components form the basis for text recognition and dimension comparison in later stages.
4. Text Recognition Layer
OCR for Text Recognition: PyTesseract extracts text from identified regions containing notes and Paddle OCR recognizes the dimensions. The complementary use of both OCR engines ensures more accurate recognition across diverse text types and fonts used in engineering drawings.
Text Data Storage: Recognized text elements are stored and retrieved for comparison in Excel sheet.
5. Comparison and Analysis Layer
In this stage, the system performs a meticulous comparison between the extracted dimensions and textual elements from the input CAD file and the client's revised design. Advanced algorithms identify inconsistencies such as missing annotations, incorrect dimensions, or altered symbols. This layer also supports threshold-based deviation analysis, allowing for configurable tolerance levels depending on project-specific requirements.
6. Report Generation and Output
A structured comparison report is generated, summarizing detected differences and possible mismatches. The report is delivered to the user, completing the automated CAD comparison workflow.
This architecture ensures a structured, efficient, and automated pipeline for CAD design analysis, leveraging real-time object detection and OCR for precise and scalable comparisons.

8. CONCLUSION

This study evaluated various object detection models, including YOLOv8m, YOLOv8x, YOLO-NAS, and Faster R-CNN, to determine the most suitable approach for CAD design comparison. Initially, traditional methods such as Structural Similarity Index (SSIM), CRAFT (Character Region Awareness for Text detection), Karas-OCR, Easy-OCR which are pre-trained OCRs for scene text detection and recognition were explored but found ineffective due to the complexity and variability of CAD designs. By leveraging deep learning-based object detection, we systematically analyzed different YOLO variants and Faster R-CNN using a dataset of 5,000 CAD images.
The results indicate that YOLOv8m outperforms all other models in terms of precision, recall, and overall efficiency. While YOLOv8x exhibited marginally higher accuracy, it came at the cost of increased computational requirements. YOLO-NAS, despite its architectural optimizations through Neural Architecture Search (NAS), showed lower precision (60-70%) in our evaluations. Faster R-CNN, though known for its accuracy, was unsuitable for real-time CAD comparison due to its slower inference time. Overall, YOLOv8m emerged as the optimal model for CAD design comparison, balancing detection accuracy and speed. Its ability to efficiently recognize variations in CAD drawings makes it a powerful tool for automated analysis in engineering and manufacturing applications.
Future Enhancements:
Custom training OCRs- Paddle OCR and PyTesseract OCR can be trained to recognize special symbols and notations more accurately. Recent advancements in Vision Transformers (ViTs) and hybrid models integrating transformers with CNNs (e.g., YOLO with Transformers) could provide improved detection capabilities.
The model can be trained on larger and more diverse Datasets with more variations in scaling, rotation, and different CAD standards to improve the model's generalization and robustness.
By addressing these enhancements, the proposed approach can be further refined to achieve higher accuracy, better efficiency, and broader applicability in real-world CAD comparison tasks.
CONFILCT OF INTEREST
The author declares that there are no conflicts of interest regarding this article.

9. OUTPUT RESULTS

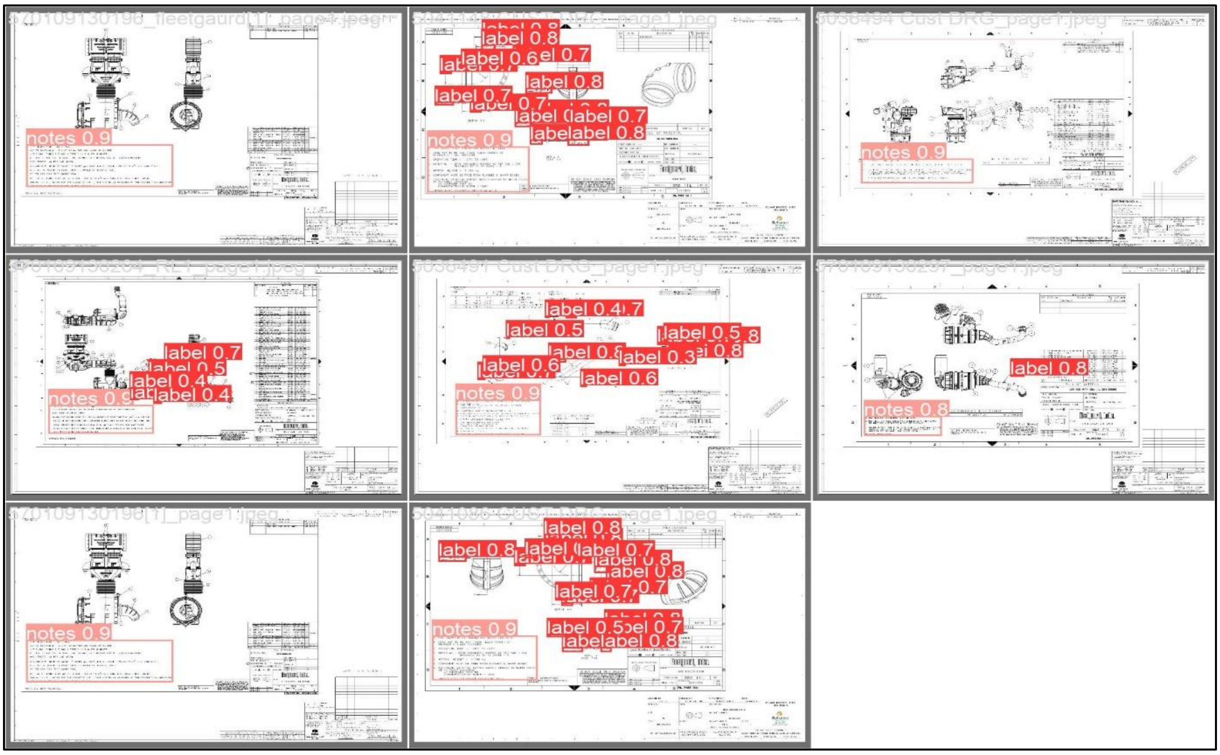

Figure 10. The validation output of yolov8m after 100 epochs ,trained on 2 classes- label (Dimensions) and Notes

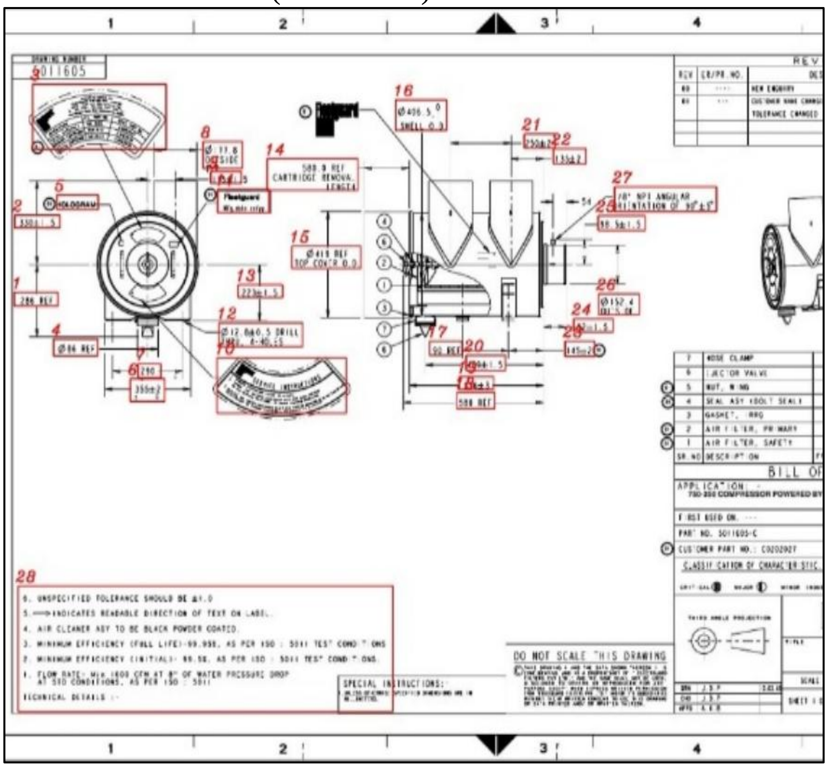

Figure 11. The output of yolov8m on unseen CAD design – identifying all dimensions and notes with 98% accuracy (ABC Company's design)

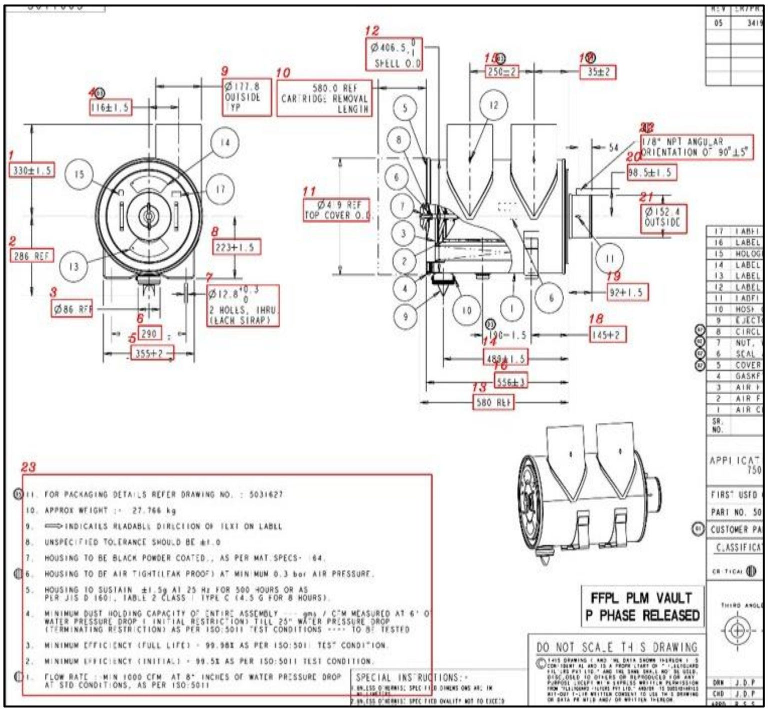

Figure 12. The output of yolov8m on CAD design – identifying all dimensions and notes (ABC Company's Client's design with revisions)

Table 2: Excelsheet storing dimensions and Notes, highlighting differences (Paddle OCR for dimensions and PyTesseract OCR for Notes)

A	B	C	D	E	F
FPL_N	FFPL_TEXT	UST_N	CUST_TEXT	FFPL_NOTES	CUST_NOTES
1	286 REF	2	286 REF	6. UNSPECIFIED TOLERANCE SHOULD BE +1.0	Jil. FOR PACKAGING DETAILS REFER DRAWING NO. : 5031627
2	330±1.5	1	330±1.5	5.<=> INDICATES READABLE DIRECTION OF TEXT ON LABEL.	
3	5011605 5011612			4. AIR CLEANER ASY TO BE BLACK POWDER COATED.	10. APPROX WEIGHT :- 27.766 kg
4	86 REF	3	Φ86 REF	3. MINIMUM EFFICIENCY (FULL LIFE)-99.95%, AS PER 1SO : 5011	
5	HOLOGRAM-			TEST CONDITIONS	9. = > INDICATES READABLE DIRECTION OF TEXT ON LABEL
6	355±2 11 —	5	355±2	2. MINIMUM EFFICIENCY CINITIAL)- 99.5%, AS PER ISO : 5011 TEST	
7	290	6	290	CONDITIONS.	8. UNSPECIFIED TOLERANCE SHOULD BE +1.0
8	177.8 OUTSIDE TVD	9	Φ177.8 OUTSIDE TYP	|. FLOW RATE- Min 1000 CFM AT 8" OF WATER PRESSURE DROP	
9	115±1			AT STD CONDITIONS, AS PER 1SO : 50/1 SPECIAL IN	7. HOUSING TO BE BLACK POWDER COATED., AS PER MAT.SPECS-
10	SERVICE IOIACI			TECHNICAL DETAILS :- Boe	164.
11	Fleetguard Mig.dale r/yy				
12	12.8±0.5 DRILL THRU, 4-HOLES	7	12.8 0 2 HOLES. THRU (EACH STRAP)		6. HOUSING TO BE AIR TIGHT(LEAK PROOF) AT MINIMUM 0.3 bar
13	223±1.5	8	223±1.5		AIR PRESSURE.
14	580.0 REF CARTRIDGE REMOVAL LENGIH	10	580.0 REF CARTRIDGE REMOVAL LENGTH		
15	419 REF TOP COVER 0.D	11	419 REF TOP COVER o.		5. HOUSING TO SUSTAIN +1.5g AT 25 Hz FOR 500 HOURS OR AS
16	0 406.5 SHELL O.D	12	0 406.5 SHELL O.D		PER JIS D 1601, TABLE 2 CLASS | TYPE C (4.5 G FOR 8 HOURS).
17	190 OREF				
18	580 REF	13	580)REF		4. MINIMUM DUST HOLDING CAPACITY OF ENTIRE ASSEMBLY ---
19	556±3	16	556±3		gms / CFM MEASURED AT 6" OF
20	489±1.5	14	489±1.5		WATER PRESSURE DROP (INITIAL RESTRICTION) TILL 25" WATER
21	250±2	15	250±2		PRESSURE DROP
22	135±2	17	135±2		(TERMINATING RESTRICTION) AS PER 1SO0:5011 TEST CONDITIONS -
23	145±2	18	145±2		--- TO BE TESTED
24	92±1.5	19	92±1.5		
25	98.5±1.5	20	98.5±1.5		3. MINIMUM EFFICIENCY (FULL LIFE) - 99.98% AS PER 1S0:5011 TEST
26	D 152.4 OUTSIDE	21	152.4 OUTSIDE		CONDITION.
27	1/8"NPT ANGULAR ORIENTATION OF 90±5°				
28		4	116±l.5		2. MINIMUM EFFICIENCY (INITIAL) - 99.5% AS PER 1S$0:5011 TEST
29		22	I/8" NPT ANGULAR F。 06 ORIENTATION OF		CONDITION.

References

[1] Wan, Y.; Wang, X.; Lu, D. Research on Key Techniques of Chinese Text Positioning in Natural Scenes. Comput. Appl. Softw.2018,35, 243–249.

[2] Yu, B.; Wan, H. Chinese text localization in natural scene based on heuristic rules and SVM. Electron. Des. Eng. 2016,24, 161–164.

[3] Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. Proc. IEEE 1998 ,86, 2278–2324

[4] Krizhevsky, A.; Sutskever, I.; Hinto, G. Imagenet classification with deep convolutional neural networks. Commun. ACM 2012,60,84–90.

[5] Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In Proceedings of the International Conference on Learning Representations (ICLR), San Diego, CA, USA.

[6] Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems, 28.

[7] He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.

[8] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You only look once: Unified real-time object detection", Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 779-788, 2016
[9] Juan T, Diana-Margarita C, Julio-Alejandro R. A comprehensive review of YOLO architectures in computer vision: from YOLOV1 to YOLOV8 and YOLO-NAS. Mach Learn Knowl Extrt 2023; 5: 1680–1716.
[10] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement", arXiv preprint arXiv:1804.02767, 2018.
[11]A. Bochkovskiy, C.-Y. Wang, and H.-Y. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," Apr. 2020.
[12] G. Jocher, A. Chaurasia, A. Stoken, J. Borovec, Y. Kwon, K. Michael, et al., "ultralytics/yolov5: v7.0 - YOLOv5 SOTA Realtime Instance Segmentation", Nov. 2022.
[13] C. Li, L. Li, H. Jiang, K. Weng, Y. Geng, L. Li, Z. Ke, Q. Li, M. Cheng, W. Nie et al., "Yolov6: A single-stage object detection framework for industrial applications", arXiv preprint arXiv:2209.02976, 2022
[14] Wang, C.-Y., Bochkovskiy, A., & Liao, H.-Y. M. (2022). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors.
[15] Jocher, G., Chaurasia, A., & Qiu, J. (2023). Ultralytics YOLO (Version 8.0.0). GitHub. https://github.com/ultralytics/ultralytics
[16] C. K. Siva Ranjani and S. Mahaboob Basha, "Video Stabilization with YOLOv8 Object Detection for Surveillance Application," 2024 5th International Conference on Smart Electronics and Communication (ICOSEC), Trichy, India, 2024, pp. 1588-1595, doi: 10.1109/ICOSEC61587.2024.10722079.
[17] H. Fakhrurroja, D. Pramesti, A. R. Hidayatullah, A. A. Fashihullisan, H. Bangkit and N. Ismail, "Automated License Plate Detection and Recognition using YOLOv8 and OCR With Tello Drone Camera," 2023 International Conference on Computer, Control, Informatics and its Applications (IC3INA), Bandung, Indonesia, 2023, pp. 206-211, doi: 10.1109/IC3INA60834.2023.10285750. [18] Dewi, C.; Chen, R.C.; Jiang, X.; Yu, H. Deep convolutional neural network for enhancing traffic sign recognition developed on Yolo V4. Multimed. Tools Appl. 2022, 81, 37821–37845.
[19] Chen, C.; Zheng, Z.; Xu, T.; Guo, S.; Feng, S.; Yao, W.; Lan, Y. YOLO-Based UAV Technology: A Review of the Research and Its Applications. Drones 2023, 7, 190
[20] Lippi, M.; Bonucci, N.; Carpio, R.F.; Contarini, M.; Speranza, S.; Gasparri, A. A Yolo-based pest detection system for precision agriculture. In Proceedings of the 2021 29th Mediterranean Conference on Control and Automation (MED), Puglia, Italy, 22–25 June 2021; pp. 342–347.
[21] Bhambani, K.; Jain, T.; Sultanpure, K.A. Real-time face mask and social distancing violation detection system using Yolo. In Proceedings of the 2020 IEEE Bangalore Humanitarian Technology Conference (B-HTC), Vijiyapur, India, 8–10 October 2020; pp. 1–6.
[22] Al-Masni, M.A.; Al-Antari, M.A.; Park, J.M.; Gi, G.; Kim, T.Y.; Rivera, P.; Valarezo, E.; Choi, M.T.; Han, S.M.; Kim, T.S. Simultaneous detection and classification of breast masses in digital mammograms via a deep learning YOLO-based CAD system. Comput. Methods Programs Biomed. 2018, 157, 85–94
[23] Mohammed, A. D., & Ekmekci, D. (2024). Breast cancer diagnosis using YOLO-based multiscale parallel CNN and flattened threshold Swish. Applied Sciences, 14(7), 2680.
[24] Tan, L.; Huangfu, T.; Wu, L.; Chen, W. Comparison of RetinaNet, SSD, and YOLO v3 for real-time pill identification. BMC Med. Inform. Decis. Mak. 2021, 21, 1–11.
[25] Yogesh K, Pankaj K. Comparative study of YOLOv8 and YOLO-NAS for agriculture application. Int Conf Signal Process Integr Net 2024; 2024: 72–77.
[26] Ezzeddini L, Ktari J, Frikha T, Alsharabi N, Alayba A, J Alzahrani A, Jadi A, Alkholidi A, Hamam H. Analysis of the performance of Faster R-CNN and YOLOv8 in detecting fishing vessels and fishes in real time. PeerJ Comput Sci. 2024 May 31;10:e2033.
[27] Tiwari, P., & Sharma, R. (2023). Evaluating Faster R-CNN and YOLOv8 for traffic object detection and class-based counting. Journal of Electrical Systems, 20(1), 152–164.
[28] Ultralytics. YOLOv8—Ultralytics YOLOv8 Documentation. 2023. Available online: https://docs.ultralytics.com/models/yolov8/