Journal of Techniques, ISSN: 2708-8383, Vol. 8, No. 1, 2026, Pages 11-27
https://doi.org/10.51173/jt.v8i1.2836

JOURNAL OF TECHNIQUES

Journal homepage: http://journal.mtu.edu.iq

RESEARCH ARTICLE - ENGINEERING (MISCELLANEOUS)

Real-Time Bottle Quality Control: Comparing Mask R-CNN and YOLOv8x-Seg on an Industrial Conveyor

Mustafa Emad Ahmed[1], Ahmed A. Thabit[2*], Ali Mahdi Hammadi[3], Haider W. Oleiwi[4]

[1]Department of Control and Automation Techniques Engineering, Electrical Engineering Technical College, Middle Technical University, Baghdad, Iraq

[2]Department of Electrical Engineering Techniques, Electrical Engineering Technical College, Middle Technical University, Baghdad, Iraq

[3]Department of Space Technology Engineering, Electrical Engineering Technical College, Middle Technical University, Baghdad, Iraq

[4]Department of Electronic and Electrical Engineering, Brunel University London, London, United Kingdom

[*] Corresponding author E-mail: ahmed.thabit@mtu.edu.iq

Article Info.

Article history:

Received
02 January 2026

Revised
31 January 2026

Accepted
07 February 2026

Published
31 March 2026

Abstract

Machine Vision (MV) is a computational technology that can independently capture images and analyze them to recognize and interpret their content. to automate visual inspections that were previously performed manually by human inspectors. Automated target recognition is among the key industrial applications of machine vision, and quality assurance on production lines depends heavily on defect detection characterized by speed and accuracy to enhance efficiency and product reliability. This study presents a comparative analysis of two instance-segmentation models, Mask R-CNN and YOLOv8x-Seg, for real-time visual inspection of bottles on industrial conveyor systems. Both models are used to detect multiple attributes, including bottle caps, labels, and liquid levels, with a specific focus on empty bottle detection. When test results are compared, Mask R-CNN is adjudged to have perfect accuracy (100%) at detecting caps, labels, and liquid levels, but has no reliability in the detection of empty bottles. It has processing time of between 0.16 and 0.24 seconds, with the time of inspection on the conveyor belt being 0.265 seconds, which limited the system throughput to 226 bottles per minute at a conveyor speed of 45 cm/s. In contrast, the system has been developed with YOLOv8x-Seg with 100% accuracy in all inspections, including detecting an empty bottle, and has also achieved much lower processing times, from 0.02 to 0.07 seconds. The time of inspection on the conveyor belt is 0.140 seconds. This in turn enhances the performance, which allows the system to run at a higher conveyor speed of 67.5 cm/s with a throughput of 428 bottles per minute and also surpasses previous works. Furthermore, decoupling image acquisition from processing and using an index-based PLC tracking mechanism instead of timer-based synchronization to reject defective bottles from the conveyor belt improves performance and reliability, while IoT technologies enhance production line monitoring, controlling, and productivity under dynamic conditions.

Publisher: Middle Technical University

Keywords: Machine Vision; Bottle Inspection; YOLOv8x-Seg; Mask R-CNN; IoT-Based Monitoring; PLC Integration.

1. Introduction

Deep learning (DL) and machine vision (MV) have made a lot of progress in the last few years. This is very important for a company when assessing the quality of plastic bottles. In this regard, convolutional neural network architectures such as NASNetMobile, MobileNetV2, EfficientNetB0, and InceptionV3 were successfully employed, as they are able to autonomously learn hierarchical feature representations without manual feature engineering. These gadgets have become increasingly popular since they are small, useful, and fast. All of these are highly important for delivering edge content in real time [1]. Infinite Symmetrical Exponential Filter (ISEF) can also be used to locate lines in cameras or figure out how much liquid is in bottles. With a Siemens PLC connected to a GUI, it was easy to locate bottles that were either too full or too empty. These machines operated well in controlled situations, but they could not handle bottles of variable shapes, labels that got in the way, or lighting conditions that were not the same. They also could not distinguish the difference between modest adjustments in the fill level. This highlights how crucial it is to have stronger models that can learn by perceiving things in new ways [2].

Nomenclature & Symbols

MV	Machine Vision	ROI	Region of Interest
CV	Computer Vision	HMI	Human-Machine Interface
CNN	Convolutional Neural Network	RPNs	Region Proposal Networks
PLC	Programmable Logic Controller	ResNet-50	Residual Network with 50 Layers
R-CNN	Region-based Convolutional Neural Network	FPN	Feature Pyramid Networks
YOLOv8xS	You Only Look Once Version 8 Extra-large for Segmentation	IOU	Intersection Over Union

As Industry 4.0 progresses, MV is gaining more significance in business. The hand-held examinations for the eyes were unreliable, tiring, and slowed workflow. These systems allow for quick inspection by robots and lines, giving accurate results that are fair to all parties. These systems snap pictures, slice the pictures, locate parts, and analyze them in order to look for and repair faults. LabVIEW, OpenCV, and MATLAB form the basic rejection software toolbox [3].

MV technologies are absolutely great for beverage bottling companies, as they have to check things quickly, right, and continuously. They are also inexpensive and can be used in harsh industrial environments where authentic sensors would not work [4]. People utilize MV for more than simply bottles. This implies that machine vision may be utilized for many different kinds of quality control tasks. Things may also go wrong while inspecting bottles, including when the light changes or the form of the object changes. This indicates that a number of different fields may utilize the same methodology [5]. Lastly, the research work establishes an operational system that inspects industrial bottle filling operations through machine vision and deep learning methods for real-time inspection. The system inspects various quality parameters, which include cap presence, label placement, and liquid level, while it particularly focuses on accurate detection of empty bottles at high conveyor operation speeds. The research presents an industrial system that integrates instance segmentation models with control through PLC and monitoring through IoT, which provides a complete solution that surpasses earlier research. This is basically depending on manual inspection and traditional machine vision systems and deep learning systems tested only in controlled environments.

Referring to the related previous work, Shah et al. [6] presented a low-cost, real-time method for detecting bottle fill levels on a conveyor belt. The system used a grayscale camera and a proximity sensor. Image processing started with a central seed point and used thresholding and morphological filtering to find the liquid-air barrier and measure the fill level. The test was conducted in MATLAB. It specifically achieved 0.5–0.75% accuracy on clear and foggy bottles and scans 40 bottles per minute. The method is scalable, lightweight, and requires little computing power, but is limited by static calibration and an inability to learn.

Sridev et al. [7] identified absent caps, tamper rings, and inadequate tampering autonomously. An infrared sensor triggers a rapid black-and-white camera when bottles traverse at a velocity of 100 mm/sec and halt every 1.5 seconds. Backlighting emphasized the caps in the photographs. Also, image processing in MATLAB employed thresholding and morphological techniques. The program evaluated ROI against a reference image for approval. Image complementation and structural filters eliminated noise. Principal results included precise inspections, rapid processing (≤2.5 seconds), and distinct variations in light manipulation among regions of interest. However, it failed to filter intricate faults or manage ambient light. Abdul Rahman et al. [8] verified bottle shape and liquid levels. LSD distinguished items, and the Hough Transform calculated water levels in 155 sample photographs. A decision tree classifier sorted faults. The results indicated an accuracy of (97% for shape recognition and 93% for level detection). The researchers then recommended to find an automated solution to human inspection to address the previous systems' under- and over-segmentation, with the necessity of having real-time industrial validations and fresh datasets.

A real-time, automated MV system was developed by Farhangi et al. [9] for factory liquid bottle inspection. A flexible, cost-effective vision-based method finds cap defects, label misplacements, and abnormal liquid levels, making food and drink quality control easier. Pattern matching finds cap flaws, edge detection finds liquid, and thresholding aligns labels. The average accuracy is 95.6%, which is good. Maximum liquid level accuracy is 100%, caps 95%, and label placement 92%. Because it is simple, cheap, and fast (150–250 ms per bottle), the study can process 7,200 bottles per hour in industry. However, it used set thresholds, only looks at one side, and is less accurate in low light. The study also recommended dual-camera setups and hyperspectral imaging.

Deep learning-based computer vision systems was introduced by Narayan et al. [10] who measured liquid levels in medicines and drinks easier and more accurate using torch builds and compares Res-Net-50, VGG-16, and DenseNet-121 models. These models were trained and tested on 486 labeled full, half-full, and overfull bottle images to avoid overfitting. CNNs with categorical cross-entropy loss functions and stochastic gradient descent optimizers were also used to preprocessed, scaled, and rotated images. Due to dense connectivity made gradients flow and features reusable, the best model was DenseNet-121, with 93% accuracy and 18% loss. ResNet-50 and VGG-16 failed the tests (83% and 86%, respectively), suggesting overfitting. Realistic CNN-based inspection pipeline design and model comparison were identified as the study's strengths. However, the use of the study's small dataset and fixed camera setup in different lighting and angles were identified as disadvantages. More diverse datasets should be investigated for better results. The applied method suggested that AI-powered industrial automation quality control systems could be improved.

The real-time AI-driven inspection system was introduced by Deepak Raj et al. [11] who detected six beverage bottle issues, including broken bottles, misaligned labels, and overfilled or underfilled containers. Using CRISP-ML(Q), 7,000 improved production-line photos were evaluated. Ultralights trained YOLOv8, v9, and v11 architectures on AWS EC2 instances. Model accuracy, recall, and mean Average Precision were assessed. YOLOv8m performed well with mAP@50 of 0.88 and mAP from 50 to 95 of 0.75. The study's systematic approach, extensive data collection, and fair model comparison were elaborated as the merit of this study. However, it cannot be assessed in low light. The paper ignored segmentation-based models and false positives. AI-driven inspection systems improved supply chain efficiency, reduced human error, and enhanced product quality, making YOLOv8m useful in industry. Afterwards, the researchers concluded that YOLOv8m can be used in industrial applications that can be integrated with IoT/sensor integration.

Mikhail et al. [12] showed that checking by hand is neither helpful nor right. Automation might improve accuracy and save costs. When a bottle stops on the conveyor, the Cognex Insight 2000 camera, the Fanuc LR Mate 200iD robotic arm, and the MicroLogix PLC check the label and cap. The PLC gets image data from the camera, then tells the robot to get rid of any bad items. The PLC-HMI-camera-robot modules worked together to show real-time data and set standards for inspections, and also make a noise when there are mistakes. There are a lot of problems with the system, such as the HMI not responding quickly enough and requires change of visual settings by hand. However, the system was able to find and sort out difficulties. It lets users change their efficiency goals and issue alarms if they are not fulfilled. Next advancements should emphasize expedited interfaces and lighting-adaptive image processing.

Kaewmorakot et al. [13] attempted to produce a cost-effective and efficient prototype for industrial bottling cap visual inspection. Since most visual inspection methods are expensive and imported, this study investigated the need for precise, fast, and affordable approaches. LabVIEW features an LED-lit Basler acA640-120gc CCD industrial camera with a global shutter sensor were used. After installing the hardware, the program measured bottle cap brightness using LabVIEW IMAQ Light Meter algorithms. It mostly employs light intensity thresholding instead of complicated edge detection. Caps increase intensity (around 50,000) if compared to bottles without tops (15,000). Within a 6-hour test with different conveyor speeds, the system found 96.77% of items. The best pace was 300 bottles per minute with 98.87% accuracy. The average bottle processing time was 25.69 milliseconds, allowing for easy, inexpensive, and accurate research.

The current study demonstrates its key contribution by providing a direct industrial assessment that compares Mask R-CNN and YOLOv8x-Seg and previous work results based on accuracy, processing speed, production output, and bottle inspection duration on the conveyor and conveyor belting speed. The tracking mechanism based on the index system uses the defective bottle array stored in the PLC database to create an advanced method that replaces traditional timer-based synchronization for accurate identification of defective bottles and removes them from the conveyor belt during different operating scenarios. The study also establishes its contribution to scientific knowledge through its dual focus on the performance of algorithms and their capacity to function in industrial environments, which provides a distinct research opportunity because existing studies investigate isolated inspection processes without real-time requirements or practical production system evaluation.

2. Methodology

The following points are devoted to explain the system design steps and parameters used in this paper. These points are sorted by role:

- Conveyor belt: Three motors powered a 6-meter conveyor belt in the simulation. In the Mask R-CNN configuration, the motors ran at 30 cm/s, 30 cm/s, and 45 cm/s, while in the YOLOv8x-Seg configuration, the speeds were 30 cm/s, 54 cm/s, and 67.5 cm/s. These speeds were designed to balance inspection reliability and production efficiency.
- DC motor: It was used by a small DC motor equipped with two input ports: x1 for 24V and x2 for 0V.
- Photoelectric sensor: Using three optical sensors mounted on the conveyor belt, the first sensor triggers a camera to take a photo for visual evaluation of the bottle. The second sensor activates the linear solenoid actuator to eject defective bottles. Finally, the third sensor counts the defective bottles for direct production quality verification.
- Linear solenoid actuator: When voltage is applied, a piston-based solenoid pushes out. There are two ports: x1 for 24V input and x2 for 0V. After a visual inspection system finds a defect in a bottle, the Snap7 library sends the bottle's identification number and class name from a Python program to an array (used as a database) in the TIA Portal environment. The array keeps track of all the IDs of defective bottles. The second photoelectric sensor sees bottles passing by and compares them. If the sensor finds a bottle with an identification number that matches one of the array's defective entries, the PLC tells the linear solenoid actuator to turn on and physically push the defective bottle off the conveyor line. The PLC array's ID numbers stand for defective bottles, and this process automatically rejects defects in real time.
- camera: A 640-pixel industrial camera powered by 24 V was used to capture real-time images. It inspected bottles for the presence or absence of a cap, label, and liquid filling level to ensure product quality.
- programmable logic controller (PLC): The Siemens PLC S7-1200, specifically the CPU 1212C AC/DC/Relay variant, is used by the system.
- human-machine interface (HMI): The system used an HMI of the KTP600 Basic Color PN type (Part No. 6AV6 647-0AD11-3AX0). The human-machine interface makes it easy to see (Defective bottles and total bottles) and control the production process(remote) through a graphical interface
- hub switch: A 5-port hub switch connected to the PLC, HMI, and PC, enabling a reliable communication network. This ensured real-time monitoring and smooth data exchange across all system components

3.1. Data preparation and training framework for mask R-CNN and YOLOv8x-seg models in bottle inspection

First, these images were obtained from an industrial camera at 640 x 640 resolutions, capturing a variety of bottle types under diverse capping, labeling, and liquid levels. The resulting images were organized into labeled folders, and annotations were carried out using the LabelMe tool, creating precise polygon masks with a subsequent quality assurance to ensure accuracy and consistency standards. Preparation and training of Mask R-CNN and YOLOv8x-seg models towards bottle inspection started with this systematic approach. Annotations were then converted into COCO format using "labelme2coco" for compatibility with Detectron2. The dataset was divided into training (70%), validation (15%), and testing (15%) subsets to enable effective learning, hyperparameter tuning, and evaluation. Training was conducted on a high-performance workstation equipped with an Intel Core i9 processor, 32 GB RAM, and an NVIDIA RTX 4060 GPU, using Python 3.10, Ultralytics, and Detectron2 v0.6, which provided the computational power necessary for efficient deep learning in industrial automation contexts, as shown in Fig. 1, which represents the data organization.

3.1.1. Mask RCNN model configuration

Mask R-CNN is initialized with a ResNet-50 backbone and Feature Pyramid Network using the Detectron2 model zoo "mask_rcnn_R_50_FPN" configuration. This architecture is ideal for instance segmentation because it extracts multi-scale image features. The first model is adjusted to detect four bottle inspection object classes: “bottle cap & no label”, “bottle cap & label”, bottle no cap and no label and “bottle no cap and label”. Model 2 customized liquid-level sensor, the classes were defined using manually collected and annotated data. A low learning rate of 0.00023 ensured smooth and effective fine-tuning on this custom dataset. The model uses pre-trained weights from large datasets like COCO, so this conservative rate prevents overfitting and stabilizes learning. Training takes 6000 iterations to help the model learn new data and improve. This setup customizes a production line visual bottle inspection model using transfer learning. at 4.3 GB VRAM, GPU training maxed. This proves the hardware-efficient training process can be used in industry, as clearly seen in Fig. 2, which mentions the Mask R-CNN framework for instance segmentation with all necessary equations [15- 17].

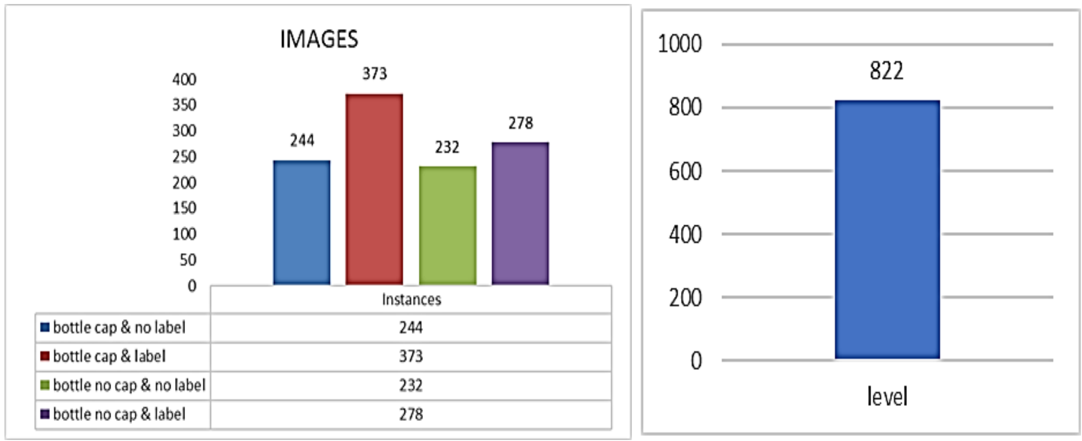


Fig.1. Data organization

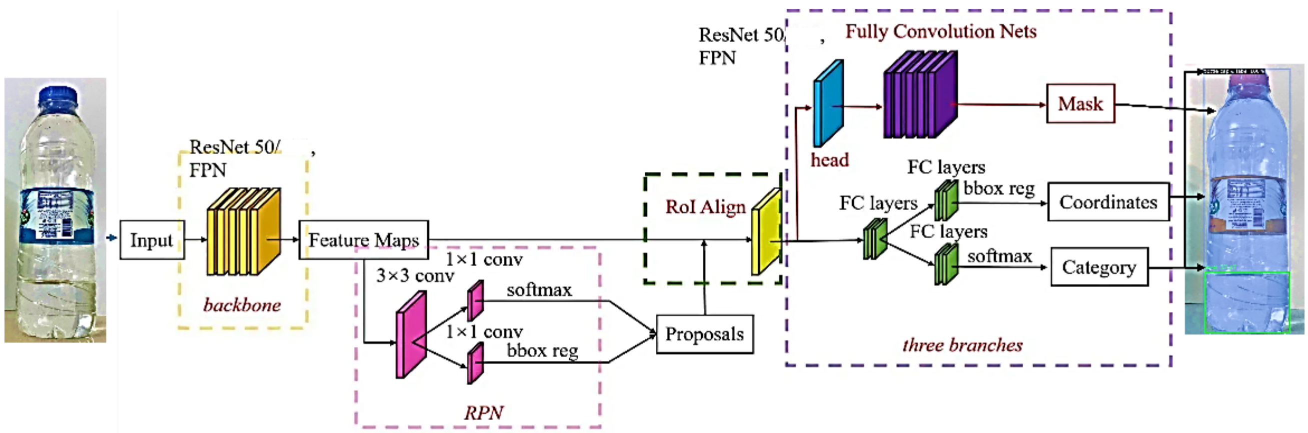


Fig. 2. The mask R-CNN framework for instance segmentation [14]

$$L_{box}(t,t^{*}) = \sum smooth\ (t_j - t_j^{*}) \quad (1)$$

L_{box} : Bounding Box Loss

t_j : Predicted displacement of the box from the neural network

t_j^{*} : Actual displacement from the actual box

$$L_{rpn\ cls}(p_j, p_j^{*}) = -[p_j^{*} log\ p_j + (1 - p_j^{*}) log(1 - p_j)] \quad (2)$$

$L_{rpn\ cls}$: Loss to determine if there is an object or not

p_j: Predicted probability of each anchor containing an object

p_j^{*}: Ground truth

$$P_k = \frac{e^{z_k}}{\sum e^{z_j}} \quad (3)$$

P_k : Predicted probability of each category

$$L_{cls} = -\log P_k \quad (4)$$

L_{cls} : Classification losses to determine the type of object

$$L_{mask} = -\frac{1}{m^2} \sum_{u=1}^{m} \sum_{v=1}^{m} [y_{uv}\ log\ p_{uv} + (1 - y_{uv})\ log(1 - p_{uv})] \quad (5)$$

y_{uv}: Actual value per pixel

UV: Represents row and column pixels

p_{uv}: Expected probability per pixel

$$p_{uv} = \sigma(S_{uv}) = \frac{1}{1+e^{-S_{uv}}} \quad (6)$$

S_{uv}: Values given by the model

$$L_{Total} = L_{Cls} + L_{Box} + L_{Mask} \quad (7)$$

3.1.2. Yolov8x-segmentation model configuration

The state-of-the-art YOLOv8x-Segmentation model was used for real-time bottle inspection, providing semantic segmentation accuracy and speed. Dataset preparation, annotation, model configuration, training, and real-time deployment comprised the development pipeline. To segment instances precisely, LabelMe with polygon masks annotated images of bottles with/without caps, labels, and different liquid levels. The labelme2yolo tool converted these annotations into YOLO-compatible format with a 15% validation split. The dataset was organized into training and validation sets using a YAML configuration file for class names and structure. initializing the model with pre-trained YOLOv8x-Seg weights improved efficiency with a small dataset using transfer learning. Training parameters included 640x640 input size, 8 batches, 0.0001 learning rate, 100 epochs, Adam optimizer, and patience = 23 early stopping criterion. Model generalization was improved by using horizontal flips, HSV shifts, and mosaic augmentation. The model was added to a real-time inspection system after training. A secondary segmentation model estimated the liquid level as the vertical distance within the segmentation mask for bottles with caps and labels. Bottles without caps or labels or with liquid levels below 95% were rejected by the PLC and automatically removed from the conveyor. In industrial manufacturing, this end-to-end system ensures fast, reliable, and automated quality control. GPU training maxed out at 7.2 GB VRAM. This proves the hardware-efficient training process can be used in industrial settings. All details were shown in Fig. 3, which stands for Yolov8 Seg Architecture [19, 20].

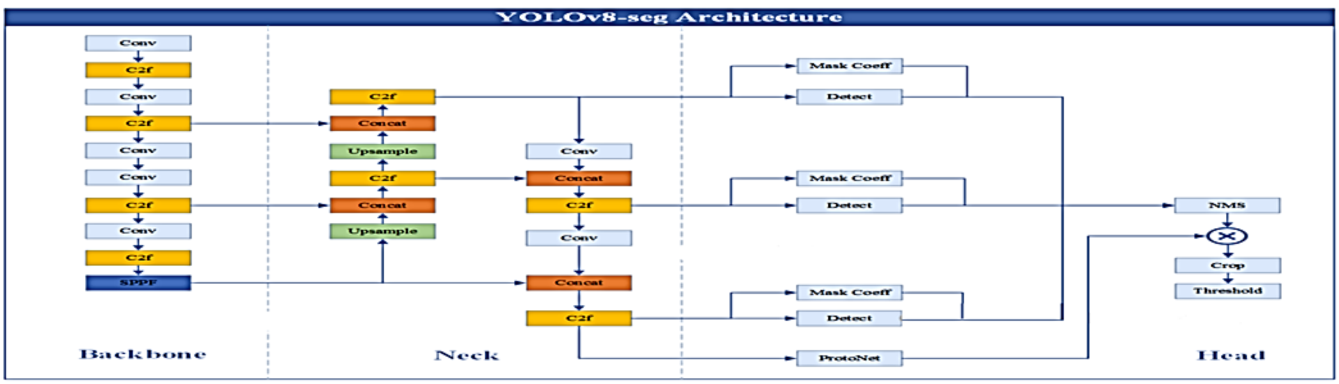


Fig. 3. Yolov8 seg architecture [18]

$$L_{box} = 1 - IOU + \frac{\rho^2(b,b_{gt})}{c^2} + \alpha v \quad (8)$$

$$\text{IOU}(B,B_{gt}) = \frac{|B \cap B_{gt}|}{|B \cup B_{gt}|} \quad (9)$$

$$v = \frac{4}{\pi^2}\left(arctan\frac{w_{gt}}{h_{gt}} - arctan\frac{w}{h}\right)^2 \quad (10)$$

$$\rho\left(b, b_{gt}\right) = \sqrt{(y_2 - y_1)^2 + (x_2 - x_1)^2} \quad (11)$$

$$\alpha = \frac{v}{(1-IOU)+v} \quad (12)$$

c: Diameter of the smallest frame that can contain both frames (predicted and actual)

IOU: Intersection over union

ρ: Distance between the center of the frame predicted by the network (b) and the center of the actual frame (b_{gt}).

v: A measure based on the difference between the dimensions of the frames (width and height angles).

α: A coefficient that depends on iou, v.

$$L_{cls} = -[y\, log\, \sigma(x) + (1-y)\, log(1-\sigma(x))] \quad (13)$$

X: Values that signify the model outputs. These values were taken and substituted them into the sigmoid to convert them to an expected range between 0 and 1.

Y: Real values that are either 0 or 1 in the classification.

$$l_{mask} = \frac{1}{N}\sum_1^N mean(B_i)\left[-G_i\, log\left(M^{\wedge}\right) - (1-G_i)\, log\left(1-M^{\wedge}\right)\right] \quad (14)$$

N: Number of objects in the image

B_i: Bounding box

$M^{\wedge}$: Expected mask of the object

G_i: Real mask of the object

3.2. PLC-Integrated visual inspection in industrial automation

The construction of a bottle inspection system consists of a design incorporated within modern industrial automation scenarios and an application of technologies in Industry 4.0 in bringing together sensing input, data processing, and executive control in closed-loop control. An optical sensor continuously inspects bottles on the conveyor and sends a digital trigger to the PLC, which will command a Python processing unit to capture high-resolution images via an industrial camera. The images are systematically numbered sequentially (e.g., 001.jpg, 002.jpg, etc.) to keep track of synchronization between the bottles and the data. Those images are analyzed using advanced deep learning algorithms- YOLOv8-Segmentation and Mask R-CNN- for various quality parameters such as the checking of cap and label presence, as well as liquid filling levels. If defective bottles are found, then the defective bottle number will be sent in an array to the PLC that acts as a database with the help of the snap7 protocol. Next, a secondary optical sensor works with a linear solenoid actuator. Upon identifying a bottle corresponding to any defective numbers stored in the PLC array, the PLC will send a command to the linear solenoid actuator, which ejects the defective bottle out of the conveyor line. This complete automation leaves little room for human interaction, thus providing accurate, consistent quality checks and boosting production efficiency through deterministic synchronization of defect detection and rejection.

3.3. Real-time bottling line control via IoT and cloud

This study showed that integrating industrial control systems with IoT and cloud technologies can improve monitoring and control. Siemens S7-1200 PLC, Python middleware, Firebase cloud platform, and Flutter-based application enable real-time data flow between the factory and the cloud, allowing operators to make faster decisions and intervene when needed. This project successfully integrates IoT and cloud technologies with industrial control systems to improve bottle production line monitoring, control, and efficiency. Real-time monitoring and remote control with a reliable PLC, a cloud intermediary (Firebase), and a cross-platform user interface (Flutter) empower operators with actionable insights and quick problem resolution. Modern industrial environments can benefit from this flexible, cost-effective, and scalable system for smart manufacturing. Figs. 4 and 5 represent the PLC_IOT Control Panel in mobile and Smart PLC Control via Firebase and Flutter.

Fig. 4. PLC_IOT control panel in mobile

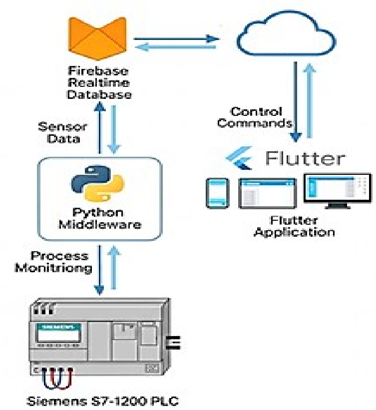


Fig. 5. Smart PLC control via Firebase and Flutter

Fig. 6 shows the main hardware components of the proposed inspection system, including a Siemens S7-1200 PLC, a SIMATIC HMI touch panel, and a network switch for communication. The HMI allows real-time monitoring and manual control, while the PLC executes the control logic and manages data exchange with the inspection software. All devices are connected via Ethernet and powered through a protected electrical circuit, creating a reliable setup that links the virtual simulation with a real industrial control system.

The ladder diagram in Figs. 7 and 8 shows how an automated system checks bottles on a conveyor. As each bottle passes, a photoelectric sensor sends a signal to the camera for counting bottles and taking pictures, and saves them in a Simulink file while keeping the order of the bottles. Python then takes the file path where the images are stored, processes them to check the cap, label, and liquid level on the images, and saves the results in a new file called processed images. This is performed in real time. When a defect is found in a bottle, the number of the defective image is stored in an array in the database PLC. Later, when the bottle gets to the rejection station, its current number is compared to the stored array. If there is a match, a cylinder is turned on to take the defective bottle off the conveyor.

By separating the image acquisition process from the image processing, the performance and reliability of inspection systems are greatly enhanced, especially for fast conveyor applications. In conventional real-time inspection systems, images are sequentially captured and processed, delaying the next bottle to be inspected. With cumulative delays due to image processing time, this can cause bottles' positions and inspection decisions to fall out of sync, leading to wrong rejection behavior.

Most previous studies and industrial implementations mostly utilize timer-based synchronization of the inspection zone and the rejection zone. In such a type of system, any variation in conveyor speed or fluctuations in image processing time directly affect the accuracy of bottle rejection. These small delays may allow defective bottles to pass undetected or cause the wrongful rejection of good products.

The proposed system, by contrast, eliminates the timer and instead uses an index-based tracking mechanism through the defective bottle array saved in the PLC database. It provides for a rejection decision to be made based solely on the logical order of bottles, rather than a time delay. Thus, the system is resilient to any variations in speed by the conveyor or processing time by the inspection, and can accurately detect and reject defective bottles under dynamic conditions. It is, therefore, robust against variations such as conveyor speed and processing time by the inspection, in order to reject defective bottles correctly and reliably with dynamic behavior.

Fig. 6. Hardware system

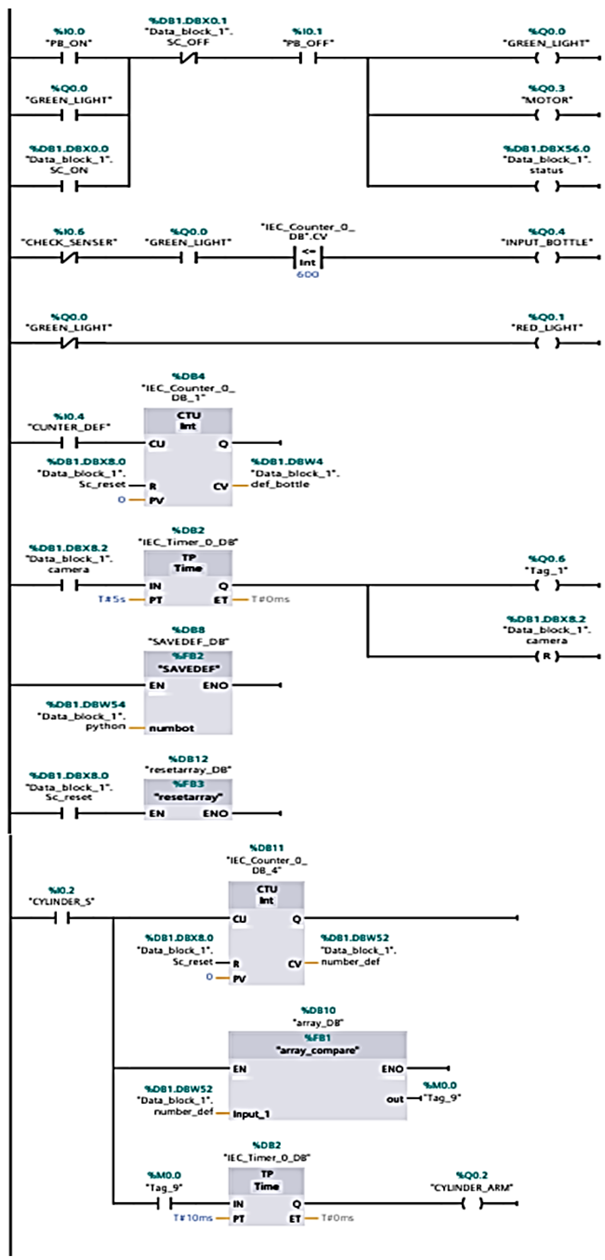


Fig. 7. The ladder diagram

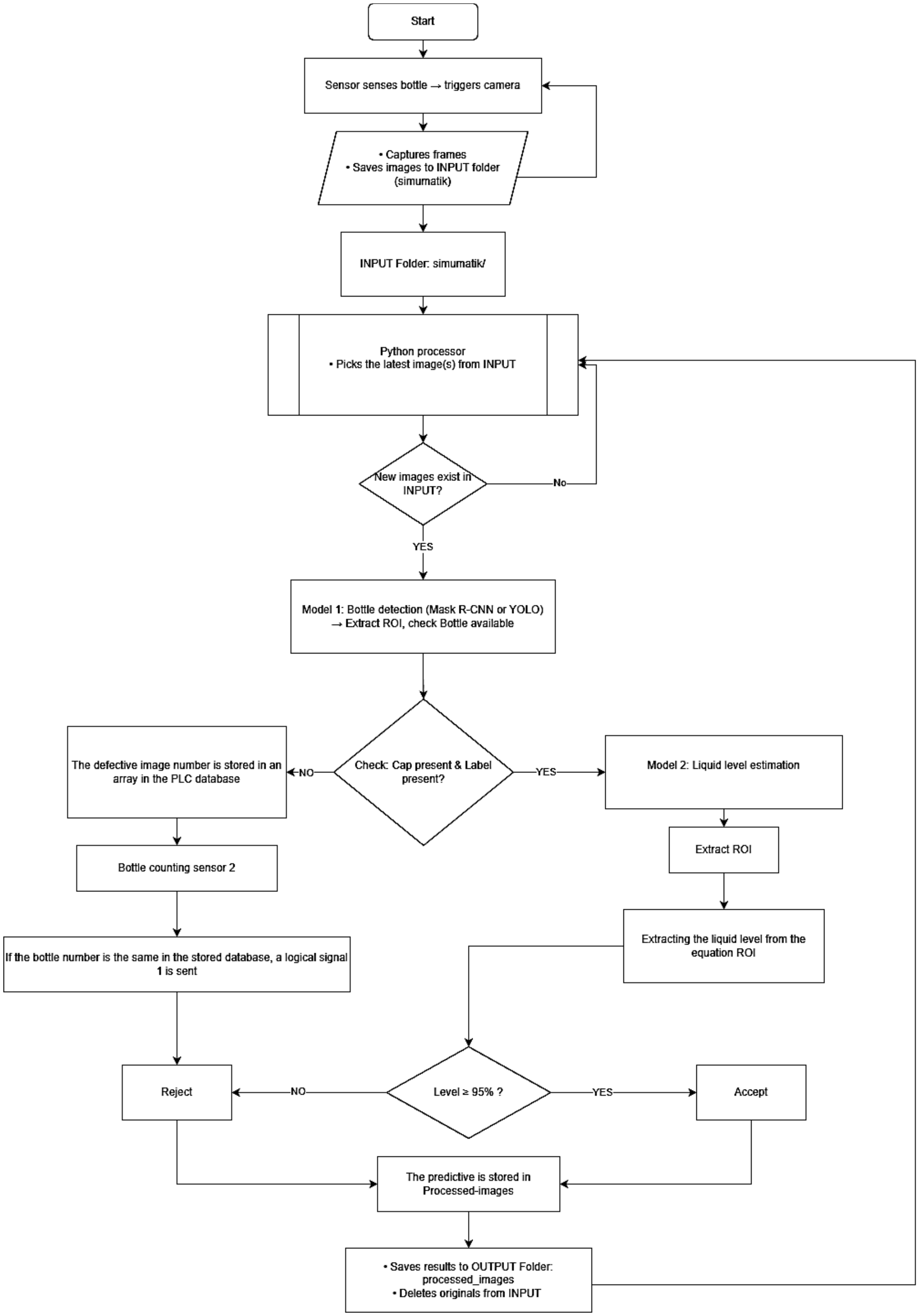


Fig. 8. Flowchart of the work system

4. Results and Discussion

This section presents the results derived from inspecting the bottles on the conveyor simulated in an industrial setting with Simumatik. Real-time checking via a camera was completed using two unique algorithms, Mask R-CNN and YOLOv8x-seg. The first model determines whether the bottle cap and label are present or absent, whereas the second model examines the level of liquid in the bottle.

The results illustrate the efficacy of both algorithms, which are captured by a number of indicators. Starting with Classification Loss (CLS Loss), Box Loss follows through to the height of a boundary box, and Mask Loss relates to the performance of models concerning the segmentation of objects. The final component, Total Loss, then combines all these losses along with the comprehensive assessment of performance. On top of that, the section provides results of Average Precision (AP) for both models to evaluate how accurate each model is in detecting and segmenting the bottles. The study here describes the workings of each model in the examination of caps and labels, as well as with respect to liquid levels in bottles, looking into the application of such models in actual industrial settings with considerable precision and reliability

4.1. Mask RCNN training results for the first and second models

The three losses (CLS, Box, and Mask) gradually got better until they reached very low levels. The first model makes sure that the label and cap are there. The Classification Loss (CLS Loss) started at about 1.5, then dropped quickly to 0.2, and then stayed at 0.01. This shows that the class distinction is very stable. The Box Loss started at 0.12, went up to 0.18 for a short time, and then slowly went down to 0.01 after about 1500 steps. This shows that the bottles were found with great accuracy. The Mask Loss started high at 0.7, but after 500 steps, it quickly dropped to below 0.1 and then stayed at 0.03, showing that it is very good to capture the shapes of objects right.

In the second model, which is all about finding the level of a liquid, both the Classification Loss and the Box Loss went down slowly but surely. This means that the model is favorable at finding and naming places where there is a liquid level. The Mask Loss, on the other hand, did much better. It kept getting better at separating things and making very clear liquid level lines inside the bottle, as seen in Figs. 9 and 10, respectively.

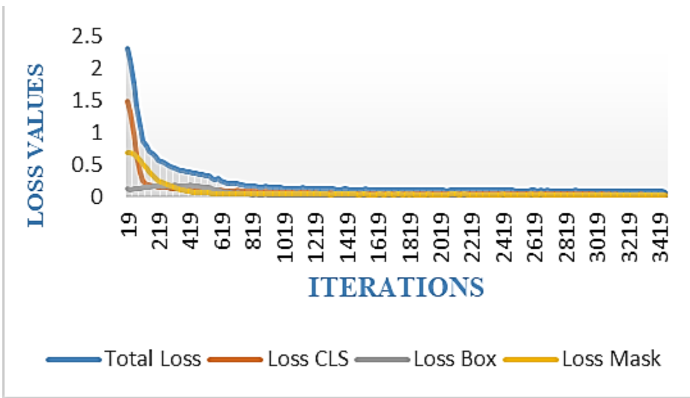


Fig. 9. First model training of total loss of cap and label

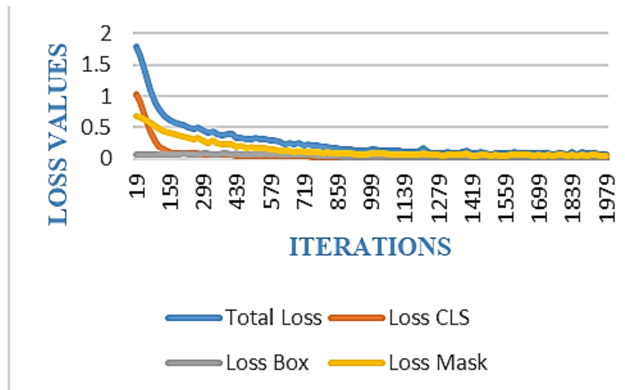


Fig. 10. Second model training of total loss of liquid level

4.2. Precision average training results for Mask RCNN

Figs. 11 and 12 show the model's Average Precision (AP). Specifically, Fig. 11 shows the liquid level detection, the model is accurate at localizing and segmenting the liquid level, with segmentation being slightly better (BBox AP = 95.38% and Segmentation AP = 97.33%). However, Figure 12 shows the comparison of cap/label detection, four cases (bottle with/without cap and label). AP was always higher for segmentation than for bounding box detection. BBox AP ≈ 95.47% and Segm AP ≈ 97.60% with cap and label. Bounding box detection was 94.21%, and segmentation was 97.29%, proving that the model could segment objects better than boxes.

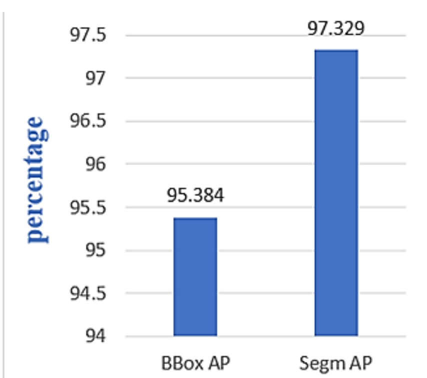


Fig. 11. Average precision of the second model

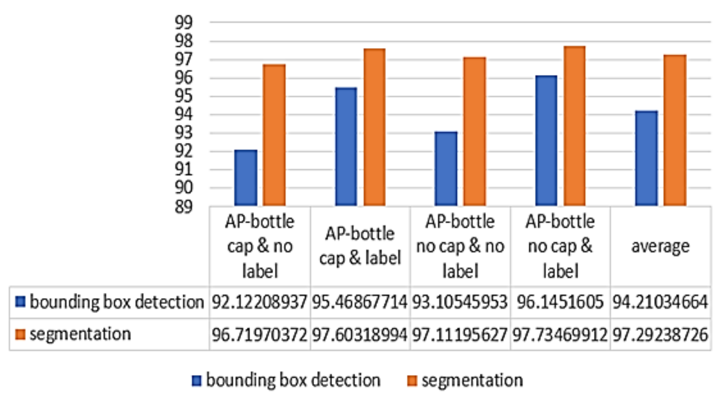


Fig. 12. Average Precision of the first model

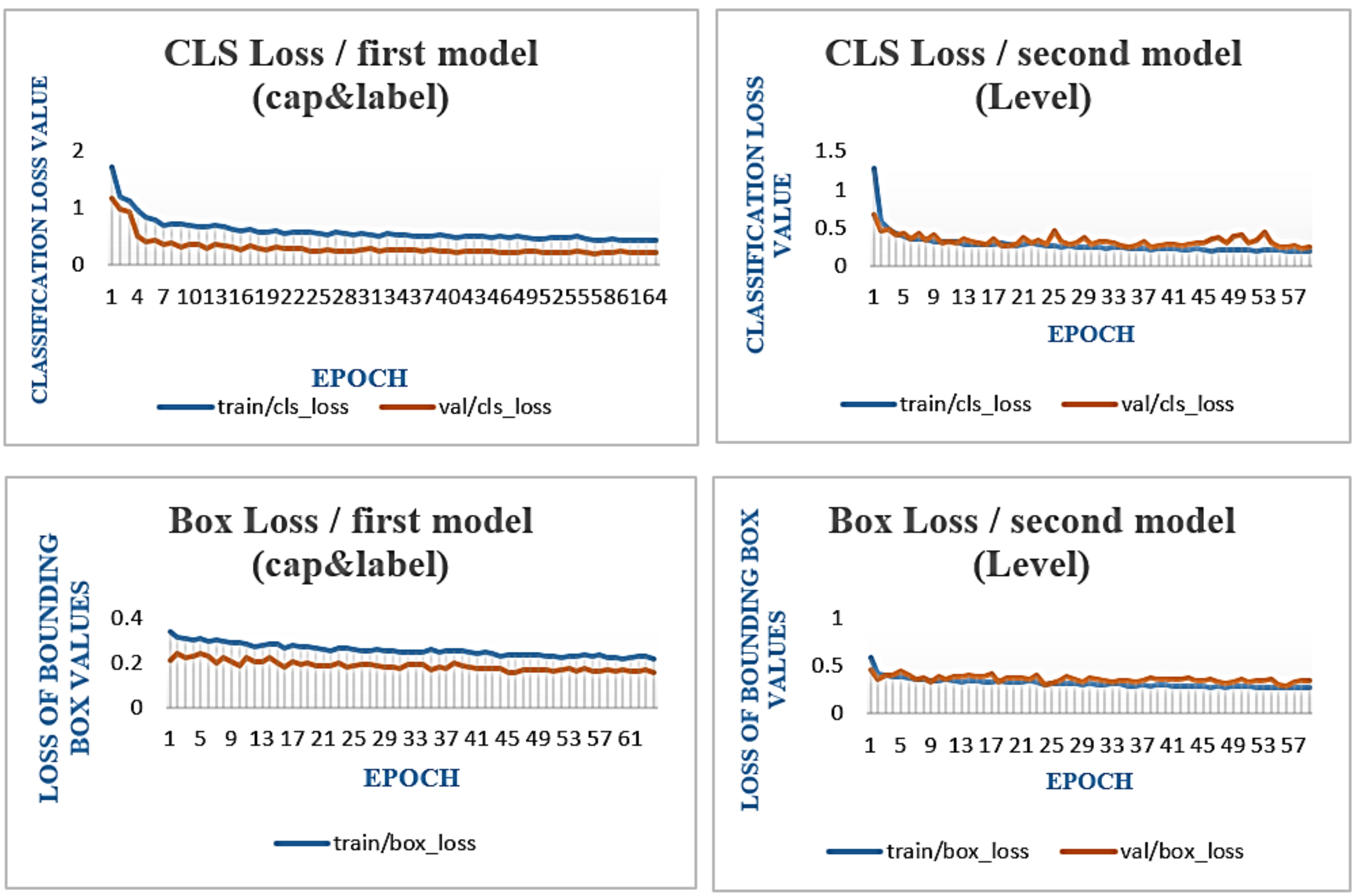
4.3. Yolov8x seg training results for the first and second model

Fig. 13 stands for YOLOv8x-Seg Bottle Inspection Model Training and Validation Losses. The YOLOv8x-Seg framework performed well in two industrial bottle inspection models: cap and label detection and liquid level assessment. The classification loss for the first model (cap and label) dropped significantly from ≈1.7 and 1.1 for training and validation to a stable range of 0.2–0.3 after 30-40 epochs, indicating rapid feature learning and reliable generalization. Bounding box regression showed decreasing losses, with validation loss consistently lower than training loss (≈0.15–0.2), signifying accurate bottle region localization. The model can generate accurate pixel-level masks for bottle components without overfitting, as segmentation losses initially peaked at ≈4.0 but quickly converged to 0.2–0.3.

In the second model (liquid level), classification losses initially ranged from ≈1.3 to 0.7 but stabilized at 0.18–0.24, indicating better liquid level discrimination. Bounding box losses gradually decreased from ≈0.55–0.45 to stable values of 0.27–0.34, indicating liquid level localization generalization. Segmentation losses peaked around 4.2, but quickly decreased and stabilized between 0.37 and 0.5, proving the model can learn accurate liquid-level boundaries with strong training-validation curve alignment. Both models demonstrated that YOLOv8x-Seg attains stable convergence, strong generalization, and accurate defect detection in complementary bottle inspection tasks, guaranteeing high reliability for real-time industrial applications.

4.4. Precision average training results for Yolov8x seg

The findings depicted in Fig. 14 demonstrate that the YOLOv8x-Seg algorithm attained a high mean Average Precision (mAP = 97.7%) for both bounding box identification and segmentation, exhibiting robust consistency across various bottle conditions (with or without cap and label). This illustrates its capacity to deliver consistent and reliable performance, rendering it exceptionally appropriate for industrial applications necessitating rapid inference speed and precise real-time inspection. Conversely, the Mask R-CNN approach attained somewhat superior segmentation accuracy (≈97.3%), but exhibited diminished bounding box performance (≈94.2%), in addition to being more computationally intensive and slower in processing. In liquid level estimation, YOLOv8x-Seg demonstrated a balanced accuracy in bounding box detection (AP_Box ≈ 95.7%) and segmentation (AP_Seg ≈ 96.3%), alongside expedited inference and reduced computational expense. Conversely, Mask R-CNN attained slightly superior segmentation accuracy (≈97.3%) with comparable bounding box precision (≈95.38%) but necessitated substantially greater computational resources, thereby constraining its efficiency in industrial settings. Consequently, it can be inferred that Mask R-CNN is better suited for academic and research environments prioritizing precise segmentation accuracy, whereas YOLOv8x-Seg is the superior option for industrial applications in production lines, owing to its equilibrium of accuracy, speed, and dependability when utilized with automation systems like PLCs.



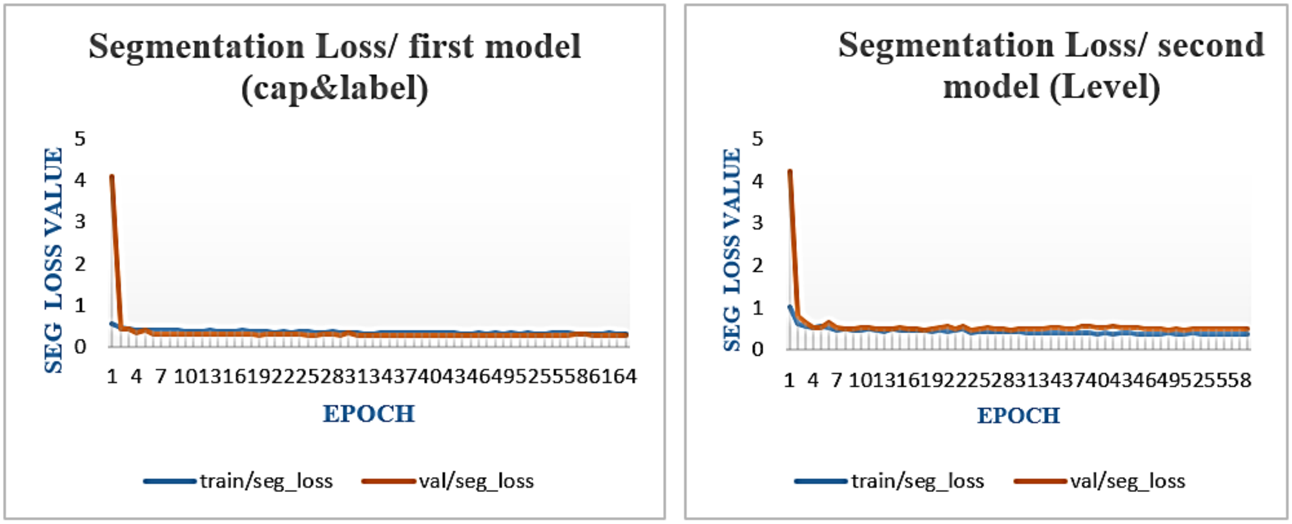


Fig. 13. Yolov8x seg training results for the first and second models

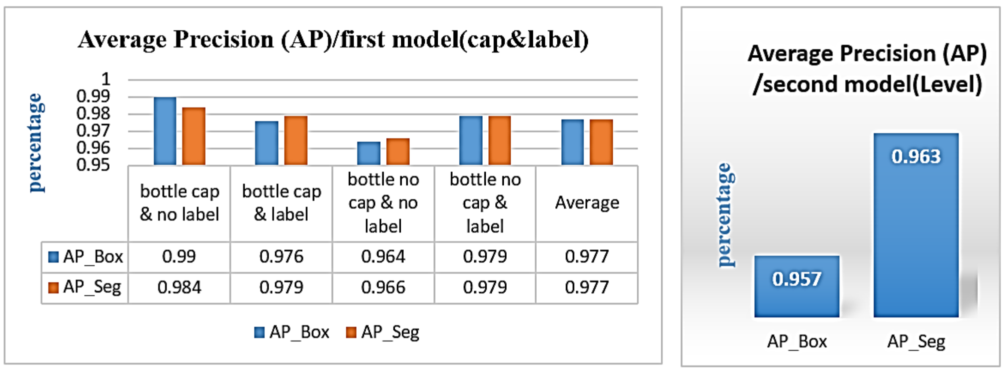


Fig. 14. Average precision training results for Yolov8x seg

4.5. Mask R-CNN algorithm result test

In this section, the results obtained by testing the Mask R-CNN algorithm in detecting and classifying bottles in different conditions were presented and discussed to detect the presence or absence of the cap and the label, as well as to determine the liquid level inside the bottle by creating precise segmentation masks that allow for accurate differentiation between these regions. Figure 15 demonstrates water and soft drink bottle detection results. The system accurately located and classified bottle components despite illumination and material transparency changes. The system detected the cap and label on soft drink bottles, demonstrating its ability to handle glossy surfaces and printed labels. For translucent water bottles, the system confidently separated the body and found missing caps and labels.

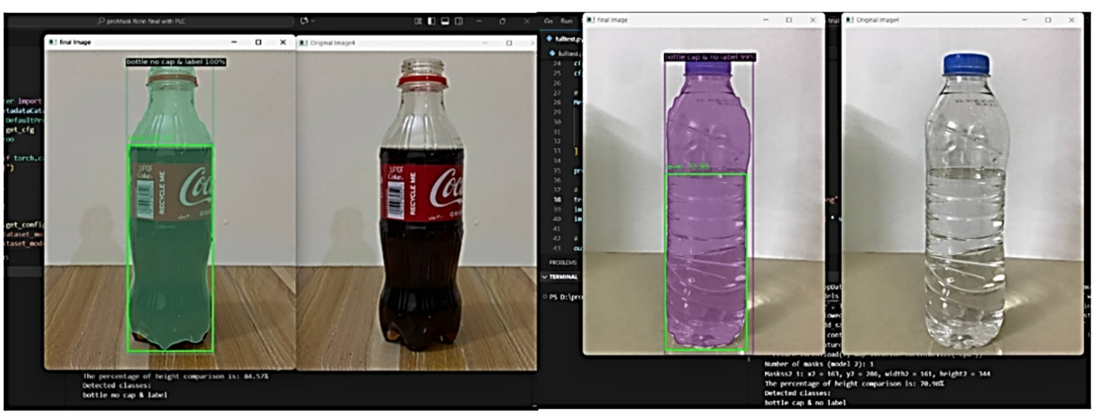


Fig. 15. Mask R-CNN algorithm test

Fig. 16 illustrates the algorithm's performance when applied to bottles placed on a moving conveyor, simulating a real industrial inspection environment. When applying the Mask RCNN algorithm, 100% accuracy was achieved in analyzing bottle images (presence or absence of a cap or label), as well as 100% accuracy in determining liquid levels. However, empty bottles were not identified. 226 bottles were examined on a moving conveyor belt within 60 seconds. The time required to examine a bottle on the conveyor belt was 0.265 seconds. The processing time for each image ranged from 0.16 to 0.24 seconds, with the conveyor belt speed being 45 cm/s, as shown in Fig. 16.

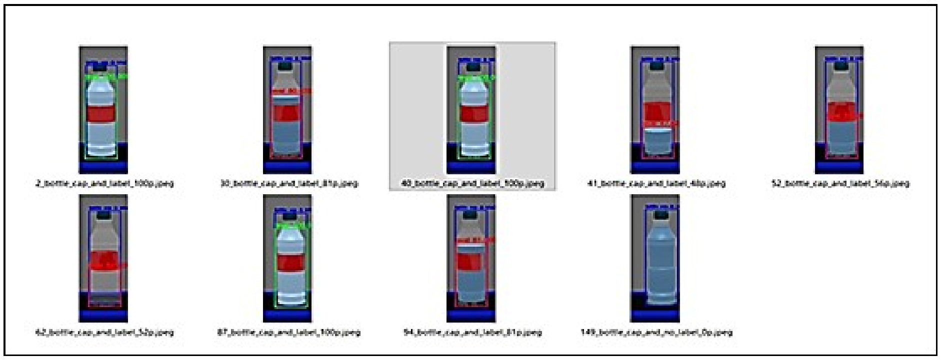


Fig. 16. Mask RCNN algorithm test on a conveyor belt

4.6. Yolov8x seg algorithm result test

In this section, the results obtained by testing the Yolov8x seg algorithm in detecting and classifying bottles in different conditions were presented and discussed to detect the presence or absence of the cap and the label, as well as to determine the liquid level inside the bottle by creating precise segmentation masks that allow for accurate differentiation between these regions. Fig. 17 demonstrates water and soft drink bottle detection results. The system accurately located and classified bottle components despite illumination and material transparency changes. The system detected the cap and label on soft drink bottles, demonstrating its ability to handle glossy surfaces and printed labels. For translucent water bottles, the system confidently separated the body and found missing caps and labels.

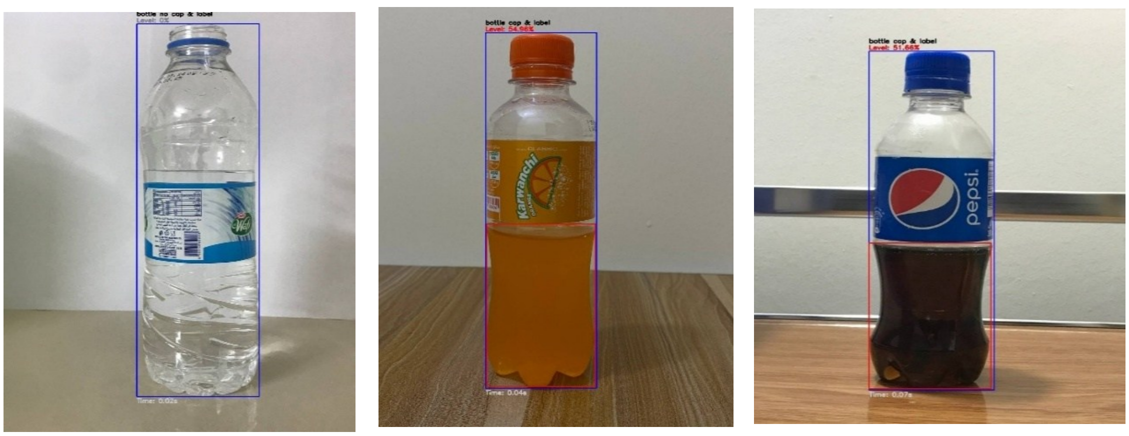


Fig. 17. Yolov8x seg algorithm result test

Fig. 18 illustrates the Yolov8x seg algorithm's performance when applied to bottles placed on a moving conveyor, simulating a real industrial inspection environment.

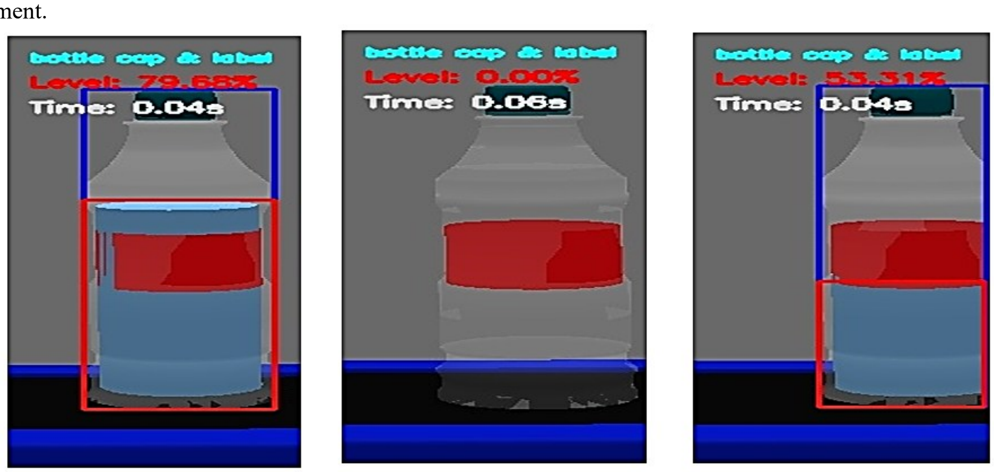


Fig. 18. Yolov8x seg algorithm result test on a conveyor belt

Fig. 19 provides the results obtained from testing the Yolov8x seg algorithm in detecting and classifying different types of bottles under different conditions, with details of the bottle prediction shown within the Python program. When applying the Yolo v8x Seg algorithm, 100% accuracy was achieved in bottle analysis (presence or absence of a cap or label), as well as 100% accuracy in determining liquid levels and empty bottles. 428 bottles were examined on a moving conveyor belt within 60 seconds. The bottle examination time on the conveyor belt was 0.140 seconds. The processing time ranged from 0.02 to 0.07 seconds, with the conveyor speed being 67.5 cm/s.

The search results indicate that the YOLOv8x-Seg algorithm clearly outperformed Mask R-CNN in conveyor belt-based bottle inspection applications. YOLOv8x-Seg achieved a very low processing time ranging from 0.02–0.07 s, and on-belt conveyor inspection time is 0.140s, which enabled operation at a conveyor speed of 67.5 cm/s and a production of 428 bottles/min, with 100% accuracy (cap, label, liquid level). By contrast, Mask R-CNN required a longer processing time (0.16–0.24 s) and an inspection time of 0.265 s, operated at a slower belt speed (45 cm/s) and with Less production (226 bottles/min), with some failures recorded for empty bottles as depicted in Table 1. Fig. 20 shows the working system of both algorithms (Yolov8x seg, mask RCNN) on the conveyor belt.

The superiority of YOLOv8x-Seg can be explained scientifically by its one-stage design that uses prototype-based segmentation and dynamic task assignment, which shortens the processing steps and preserves the full visual context of the image. Mask R-CNN, on the other hand, is a two-stage algorithm that employs Roi Align and patch-based segmentation—features that can provide relatively higher accuracy in complex or overlapping scenes but increase processing time, making it less suitable for real-time applications.

Table 1. Work system results

No.	Algorithm	item	Processing Time	Bottle Inspection Time on Conveyor Belt	conveyor belt speed	Accuracy	Production
1	Yolov8x seg	cap, label, liquid level	0.02 to 0.07s	0.140s	67.5cm/s	100%	428bottles Per minute
2	Mask R-cnn	cap, label, liquid level	0.16 to 0.24s	0.265s	45cm/s	100% excluding empty bottles	226 bottles Per minute

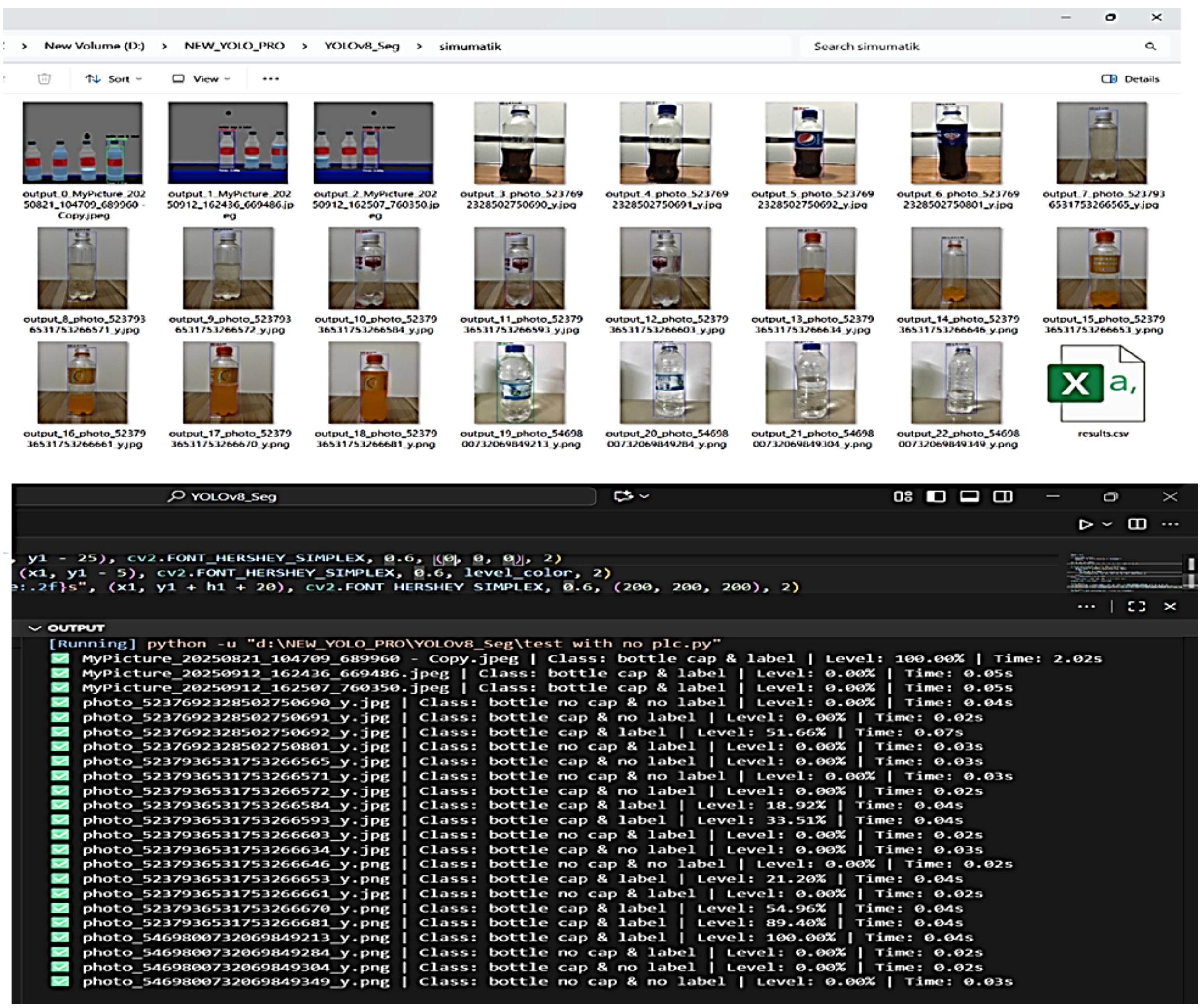


Fig. 19. Yolov8x seg algorithm result test

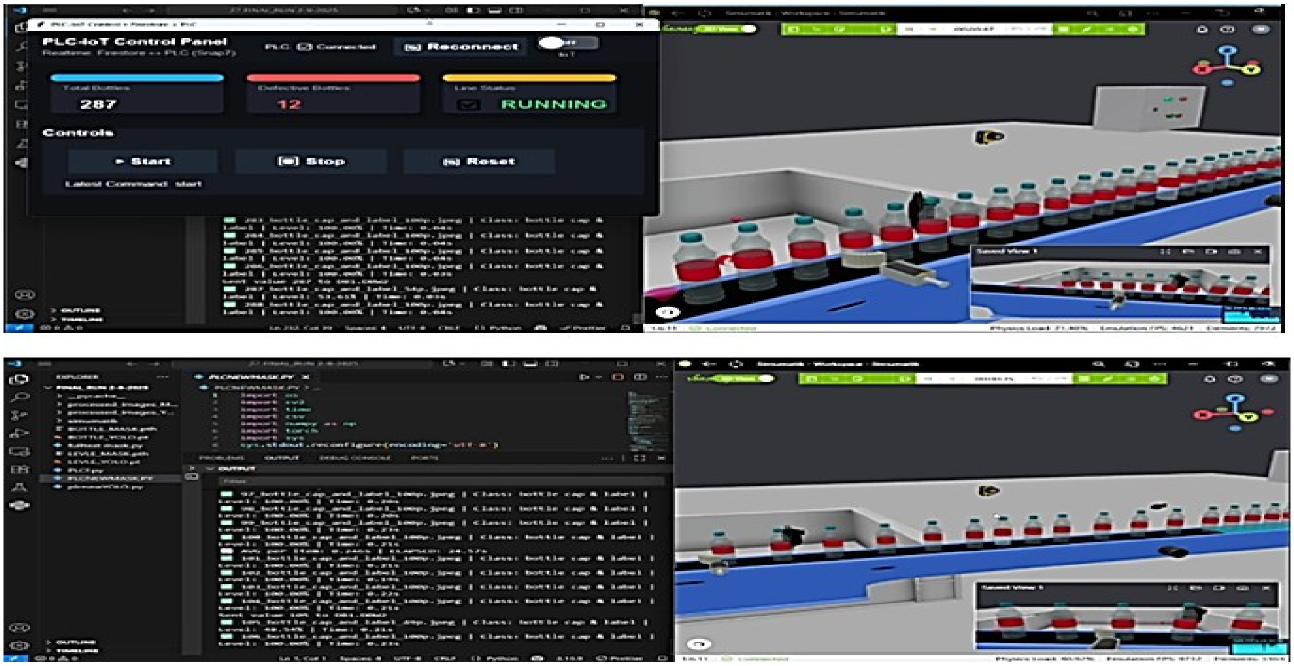


Fig. 20. Work system for Yolo v8x seg & mask RCNN

The following equation represents the above ideas for calculating the liquid level inside the bottle for both algorithms, which is illustrated in Fig. 21.

$$h_{bottle} = y_{bottom\_b} - y_{top\_b} \tag{15}$$

y_{top_b}: The coordinates of the highest point on the bottle

y_{bottom_b}: The coordinates of the lowest point on the bottle

h_{bottle}: Bottle height

$$h_{liquid} = y_{bottom\_b} - y_{top\_l} \tag{16}$$

h_{liquid}: Liquid height

y_{top_l}: Coordinate of the highest liquid level

$$level = \frac{h_{liquid} + h_{cap}}{h_{bottle}} * 100\% \tag{17}$$

h_{cap}: Cap height

Level: Liquid level ratio

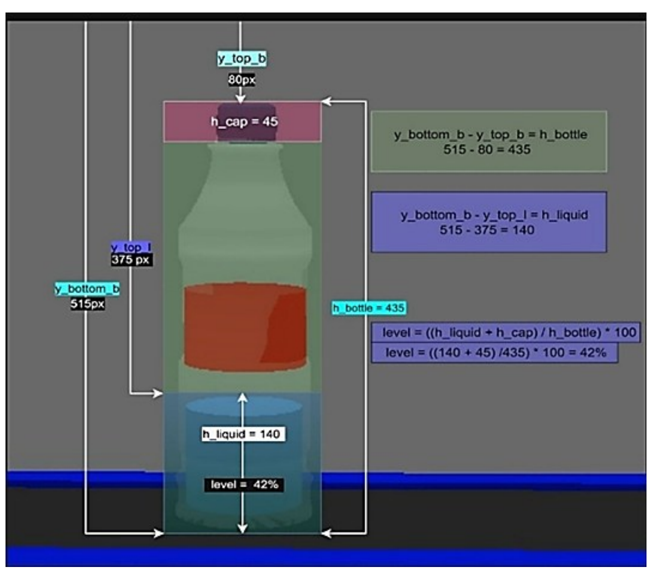


Fig. 21. Liquid level calculation

A comprehensive performance evaluation of the proposed inspection system was carried out by comparing the results of Mask R-CNN and YOLOv8x-seg, the two models employed in this study, with the results reported in previous studies related to this work in Table 2. This comparison seeks to showcase the improvements brought about in accuracy, detection speed, processing time, production, and quality of segmentation, especially regarding real-time industrial environments. Table 2 shows the quantitative results obtained from the two models used in this work, along with a number of criteria extracted from previous research.

The results of this study are distinguished by achieving superior performance in terms of speed, accuracy, and productivity, as the reported accuracy in prior studies (93–98%) was surpassed, reaching 100% while maintaining a short inspection time without the need to stop the conveyor. Moreover, most previous studies were limited to a single feature, such as cap inspection, liquid level estimation, or label detection, whereas the present study introduces an integrated system that simultaneously performs cap, label, and liquid level inspection in a real industrial environment. In addition, it provides the capability to isolate defective bottles and remove them from the conveyor, unlike other studies, which merely focused on defect identification, while also offering remote monitoring and control of system performance from anywhere in the world. From a scientific perspective, this study contributes to clarifying the architectural differences between one-stage and two-stage segmentation algorithms, while from a practical perspective, the proposed system provides a highly efficient solution that enhances production quality, reduces stoppages and waste, and thus represents a direct value for modern industrial bottling lines. Accordingly, it can be concluded that YOLOv8x-Seg is the most suitable algorithm for high-speed industrial production environments, as it combines speed, accuracy, and comprehensiveness, whereas Mask R-CNN remains strong in terms of theoretical mask accuracy but is less compatible with the requirements of continuous production lines.

Table 2. Comparison of previous work results with Mask R-CNN and YOLOv8x seg

Source	Previous Work Results	YOLOv8x-Seg Results	Mask R-CNN Results
[6]	Processing Time 0.1–0.125s, accuracy ~0.75%, traditional image processing, liquid level detection	Processing Time 0.02–0.07s, accuracy 100% (cap, label, liquid level).	Processing Time 0.16–0.24s, accuracy 100% (cap, label, liquid level) except empty bottles
[7]	Processing Time 2.5s, conveyor stop 1.5s, conveyor belt speed 10 cm/s (cap).	Processing Time 0.02–0.07s, inspection time on conveyor belt 0.140s, conveyor belt speed 67.5 cm/s without stop	Better than [7] but slower than YOLO, Processing Time 0.16–0.24s, inspection time on conveyor belt 0.265 s, conveyor belt speed 45 cm/s without stop
[8]	97% for shape, 93% for level, and statistical methods	accuracy 100% (cap, label, liquid level).	Accuracy 100% (cap, label, liquid level) except empty bottles
[9]	Processing time 0.15-0. 25, inspection time on conveyor belt 0.5s, conveyor belt speed 20 cm/s, average accuracy of 95.6%, accuracy for liquid level detection (100%), 95% for cap, and 92% for label placement. Production of 120 bottles/min	Processing time 0.02–0.07s, inspection time on conveyor belt 0.140s, conveyor belt speed 67.5 cm/s, accuracy 100% (cap, label, liquid level). Production of 428 bottles/min	Processing time 0.16–0.24s, inspection time on conveyor belt 0.265s, conveyor belt speed 45cm/s, accuracy 100% (cap, label, liquid level) except empty bottles. Production of 226 bottles/min
[10]	1-Dense-Net 121 (93%) 2-Res-Net 50 (83%) 3-VGG-16 (86%) (liquid Level)	Accuracy 100% (cap, label, liquid level).	Accuracy 100% (cap, label, liquid level) except empty bottles
[11]	1- YOLOv8m (88%) 2- YOLOv8s (83%) 3- YOLOv9c (79%) 4- YOLOv11m (85%) 5- YOLOv11s (80%) (Defective Label, Missing Label, Overfilled, Defective Bottle, Normal Bottle, Underfilled)	100%, outperforms all previous YOLO versions	100% accuracy except for empty bottles, but slower than YOLO
[12]	Conveyor stops 1s for processing (label and cap defects (different color or missing cap)	0.140s without stop, higher throughput	Outperforms [12] but slower than YOLO
[13]	Processing time 0.02-0.03, inspection time on conveyor belt 0.2s, conveyor belt speed 50 cm/s, accuracy of 98.8% for cap detection. Production of 300 bottles/min	Processing time 0.02–0.07s, inspection time on conveyor belt 0.140s, conveyor belt speed 67.5 cm/s, accuracy 100% (cap, label, liquid level). Production of 428 bottles/min	Processing time 0.16–0.24s, inspection time on conveyor belt0.265s, conveyor belt speed: 45 cm/s, accuracy 100% (cap, label, liquid level) except empty bottles. Production of 226 bottles/min

5. Conclusions

The results of this study successfully surpassed the previous works in terms of speed, accuracy, and productivity. Prior studies reported an accuracy in the range of 93% to 98%, while in the present study, it achieved the elusive 100%, and within the short inspection time, with no stopping of the conveyor. Most previous works are limited to one feature, like cap inspection, liquid level estimation, or label detection. However, the current study proposed an integrated system that conducts cap, label, and liquid level inspection simultaneously in a real industrial setup. Also, the defective bottles were isolated and removed from the conveyor, unlike in previous studies where only defect identification was made.

Experimental results showed that the YOLOv8x-Seg technique outperformed Mask R-CNN, used in previous works, in superior speed, throughput, and accuracy scores for an industrial conveyor inspection environment. YOLOv8x-Seg had the least processing times at 0.02-0.07 seconds and an inspection time of 0.140 seconds on a conveyor moving at 67.5 cm/s, and it recorded a 100% success rate in performing all inspections, including empty-bottle detection. Therefore, it showed a great fit for real-time industrial applications. While Mask R-CNN achieved 100% accuracy in identifying caps, labels, and liquid levels, it failed to consistently detect empty bottles. Because the Mask RCNN model requires a large dataset, as a two-stage detector, Mask R-CNN is data-hungry and typically requires a massive volume of annotated examples to effectively capture fine-grained details and complex features. In addition, the increased processing times (0.16-0.24 seconds) and inspection time on a conveyor (0.265 seconds) reduced the performance of Mask R-CNN in high-speed production environments. Improved reliability as well as reduced system defects were provided through the methodological framework employed in this study, which comprises segregation of image capture from processing, use of two specific models (cap/label inspection followed by liquid level analysis), and replacement of timer-based rejection with PLC database. The introduction of IoT connectivity improved the applicability of the system in an industrial context by means of remote monitoring and control aspects not usually explored by other research that focused on one Particular inspection but not total system integration.

6. Future Works

- Assess system scalability for higher conveyor speed and more complex production settings.
- Examination to search for other defect types (is the label/cap defective?) and to see whether there are blisters in the liquid level.
- Thermal imaging cameras to find the liquid level inside the label.
- Using other learning models, such as yolo v9/v10, and comparing the results with the two algorithms used in the current system.
- Feedback for correcting and recycling defective bottles instead of throwing them away from the conveyor.
- Using more than one camera in the bottle inspection process on the conveyor belt.

Acknowledgment

This research was supported by the Electrical Engineering Technical College, Middle Technical University, Baghdad, Iraq.

References

[1] M. Kazmi, B. Hafeez, F. Aftab, J. Shahid, and S. A. Qazi, "A Deep Learning-Based Framework for Visual Inspection of Plastic Bottles," IEEE Access, vol. 11, pp. 125529–125542, 2023, Doi: 10.1109/ACCESS.2023.3329565.

[2] K. J. Pithadiya, C. K. Modi, and J. D. Chauhan, "Machine Vision Based Liquid Level Inspection System using ISEF Edge Detection Technique," in Proc. International Conference and Workshop on Emerging Trends in Technology (ICWET 2010), Mumbai, India, 2010, pp. 601–605.

[3] S. Sathiyamoorthy, "Industrial Application of Machine Vision," International Journal of Research in Engineering and Technology, vol. 3, no. 1, pp. 1–3, Jan. 2014.

[4] V. A. Dave and S. K. Hadia, "Liquid Level and Cap Closure United Inspection using Image Processing," International Journal for Innovative Research in Science & Technology, vol. 1, no. 12, pp. 165–170, May 2015.

[5] M. Hao, H. Yu, and D. Li, "The Measurement of Fish Size by Machine Vision- A Review," in Proc. 9th International Conference on Computer and Computing Technologies in Agriculture (CCTA), Beijing, China, Sep. 2015, pp. 15–32, Doi: 10.1007/978-3-319-48354-2_2.

[6] C. Shah and N. Gund, "Bottling Line Inspection System using Digital Image Processing," Social Science Research Network, 2020.

[7] S. Sridev, P. Karthikeyan, C. A. Prakash, A. Jaganathan, and A. Mani, "Bottle Cap Inspection based on Machine Vision," International Journal of Engineering Research & Technology (IJERT), vol. 5, no. 4, pp. 508–511, Apr. 2016.

[8] N. N. S. Abdul Rahman, N. M. Saad, and A. R. Abdullah, "Shape and Level Bottles Detection Using Local Standard Deviation and Hough Transform," International Journal of Electrical and Computer Engineering, vol. 8, no. 6, pp. 5032–5040, Dec. 2018.

[9] O. Farhangi, E. Sheidaee, and A. Kisalaei, "Machine Vision for Detecting Defects in Liquid Bottles: An Industrial Application for Food and Packaging Sector," Cloud Computing and Data Science, vol. 5, no. 2, pp. 242–254, Jun. 2024.

[10] A. Narayan, C. DK, S. E. Saji, T. Fang, and J. Saniie, "A PyTorch-Based Deep Learning Approach for Enhanced Liquid Level Detection in Industrial Environments," presented at the 2024 IEEE International Conference on Electro Information Technology (EIT), Eau Claire, WI, USA, May 30–Jun. 1, 2024.

[11] R. Deepak Raj, R. Sowmiya, K. Swathi, G. Harikaran, K. Gayathri, A. Ezhil Litta, C. Vishvash, and D. Bharani Kumar, "AI-Driven Automated Quality Inspection for Beverage Bottles: Leveraging Object Detection Models for Enhanced Supply Chain Efficiency," International Journal of Innovative Science and Research Technology, vol. 10, no. 3, pp. 2773–2782, Mar. 2025, Doi: 10.38124/ijisrt/25mar1796.

[12] M. Mikhail and K. Abad, "Robotic inspection and automated analysis system for advanced manufacturing," IAES International Journal of Robotics and Automation (IJRA), vol. 12, no. 4, pp. 352–364, Dec. 2023.

[13] U. Kaewmorakot, L. Poolperm, P. Prongphimai, and S. Tansriwong, "The development of a prototype of an automated bottle caps visual

inspection system," presented at the International Electrical Engineering Congress (iEECON), Khon Kaen, Thailand, Mar. 9–11, 2022.
[14] P. Petru, "What is Mask R-CNN? The ultimate guide," Roboflow Blog, Aug. 9, 2023. [Online]. Available: https://blog.roboflow.com/mask-rcnn/.et al.
[15] R. Girshick, "Fast R-CNN," in Proc. 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 2015.
[16] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in Advances in Neural Information Processing Systems 28 (NeurIPS), Montreal, QC, Canada, 2015.
[17] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in Proc. 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 2017.
[18] Glenn-Jocher, "summary of YOLOv8-Seg model structure," GitHub, Mar. 29, 2023. [Online]. Available: https://github.com/ultralytics/ultralytics/issues/1710.
[19] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression," in Proc. 34th AAAI Conference on Artificial Intelligence (AAAI-20), New York, NY, USA, 2020.
[20] Z. Tian, C. Shen, and H. Chen, "Conditional Convolutions for Instance Segmentation," in Computer Vision – ECCV 2020, Cham: Springer International Publishing, 2020.