

# Applying AI-assisted image processing to detect and measure CO<sub>2</sub> microbubbles for carbon storage

LE Dang Nguyen Hai<sup>1,3</sup>, LE Nam Nguyen Hai<sup>2,3\*</sup>

<sup>1</sup> Faculty of Computer Science and Engineering, Ho Chi Minh City University of Technology, 268 Ly Thuong Kiet, Dien Hong Ward, Ho Chi Minh City, Vietnam

<sup>2</sup> Faculty of Geology and Petroleum Engineering, Ho Chi Minh City University of Technology, 268 Ly Thuong Kiet, Dien Hong Ward, Ho Chi Minh City, Vietnam

<sup>3</sup> Vietnam National University Ho Chi Minh City, Linh Xuan Ward, Ho Chi Minh City, Vietnam

\* Corresponding email: [lnhnam@hcmut.edu.vn](mailto:lnhnam@hcmut.edu.vn)

**Abstract:** *To address the challenges in the CO<sub>2</sub> injection process, CO<sub>2</sub> microbubble dispersion has been proposed as an alternative to traditional methods, such as miscible injection and water-alternating-gas (WAG) injection. This study presents an AI-assisted model for detecting CO<sub>2</sub> microbubbles, powered by the YOLOv8 algorithm, renowned for its high-accuracy predictions. Conventional image processing techniques often struggle with detecting microbubbles, particularly in cases involving overlapping bubbles, variations in size, and low-contrast images, which can lead to inaccuracies in bubble identification and measurement. In contrast, YOLOv8's advanced detection capabilities offer a more robust solution by precisely localizing and classifying microbubbles, even in challenging scenarios. The model's performance was rigorously evaluated, demonstrating its effectiveness as a valuable tool for microbubble analysis. The detection images processed using YOLOv8 illustrate its ability to accurately detect and classify bubbles of varying sizes, generating precise bounding boxes around each identified bubble. This combination of data visualization and advanced detection techniques underscores the efficacy of YOLOv8 in microbubble analysis, enabling accurate measurement and detailed characterization of bubble size distributions—an essential factor in optimizing chemical engineering processes.*

**Keywords:** CO<sub>2</sub> microbubble; Computer vision; Object Detection; keyword

## 1. Introduction

To mitigate global warming, various methods have been employed to reduce CO<sub>2</sub> emissions in the atmosphere. Carbon sequestration and storage (CSS) is a promising technique for capturing significant amounts of CO<sub>2</sub> and securely storing it underground. In recent years, numerous researchers have focused on this field. Captured CO<sub>2</sub> is injected into different types of underground formations, such as saline aquifers, depleted oil and gas reservoirs, and coal seams. However, during the injection process, controlling the flow of CO<sub>2</sub> gas or liquid is challenging due to factors such as gravity segregation, high gas mobility, and reservoir heterogeneity. Previous studies have suggested microbubble flow as an alternative to single-phase flow injection [1]. Researchers have conducted studies on the use of CO<sub>2</sub> microbubbles in enhanced oil recovery (EOR) and carbon storage [2, 3]. A CO<sub>2</sub> microbubble solution is typically generated by mixing a Xanthan gum polymer solution with an SDS surfactant and CO<sub>2</sub> gas. A high-speed aggregation system is used to produce the microbubbles [4].

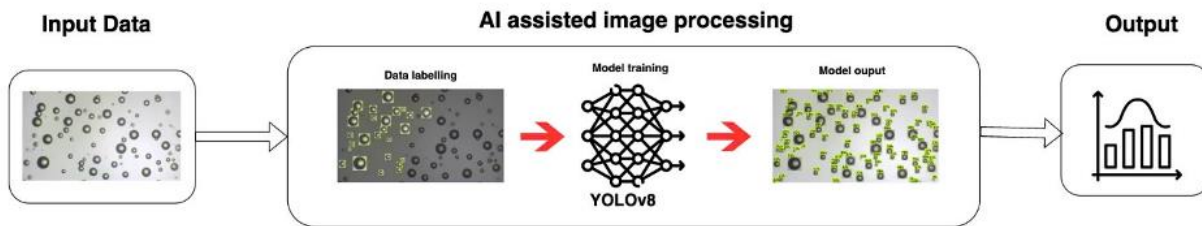
One crucial characteristic of microbubble dispersion is its size distribution. The diameters of CO<sub>2</sub> microbubbles typically range from 10-100 μm [5]. After generation, a sample of the CO<sub>2</sub> microbubble dispersion is placed under a microscope, and several images are captured using an attached digital camera. These images serve as input data for image processing software, which detects and measures the size of the microbubbles. However, this technique requires an in-depth understanding of image processing and is time-consuming, posing a challenge for many researchers in the field.

With the advancements in machine learning, several models have been introduced to overcome the limitations of analyzing microscopic-scale objects [6–8]. YOLOv8 is considered the new state-of-the-art in object detection due to its superior mean Average Precision (mAP) and efficient inference times [9]. These

advancements stem from novel improvements, including spatial attention mechanisms and feature fusion techniques, which enhance accuracy, versatility, and overall efficiency compared to its predecessors [10]. Recent studies have applied this model to improve detection capabilities in image processing tasks [11–13]. In this paper, we propose an artificial intelligence (AI)-assisted image-processing technique that leverages the capabilities of YOLOv8 as the backbone to detect labeled objects more efficiently.

## 2. Methodology

Training and testing data were collected from laboratory experiments in the previous study [3]. In this work, we captured 40 digital images of microbubbles, which were then annotated to generate suitable input for the YOLOv8 model. The annotation process, which begins by providing a class name and a bounding box to each bubble, was done by using the Roboflow annotation tool for bounding box positions around the bubbles. Each image contained an average of 50 bubbles, resulting in a total of 2096 annotations across all images. The data was loaded to Google Colab for training, and the final model was downloaded for post-processing, which calculates the pixel radius and generates the density of bubble size. (Figure 1) shows the diagram of AI-assisted image processing in this work.



**Fig. 1.** Flowchart of AI-assisted image processing model.

### 2.1 YOLOv8 model

YOLOv8 uses a spatial attention mechanism focusing on relevant image regions, leading to more precise object localization [14, 15]. In addition, the feature fusion module integrates high-level semantic features with low-level spatial details, demonstrably improving the accuracy when detecting small objects [16]. Another innovation in YOLOv8 is that the image will be preprocessed with a mosaic augmentation technique. This method involves constructing a new image by combining the crops of multiple existing images, effectively mitigating overfitting and improving the model's accuracy [17].

### 2.2 Loss function definition

All deep learning-based models require an objective function to guide learning and update weights using a backpropagation algorithm. The YOLOv8 model loss function includes box loss, classification loss, and distribution focal loss (DFL) [18]. Box loss measures the difference between predicted and ground truth bounding boxes, ensuring accurate object localization. The class loss is calculated by binary cross entropy to classify objects. DFL addresses class imbalance by assigning higher weights to losses from underrepresented classes, thereby improving performance on less frequent classes. Together, these components enable YOLOv8 to achieve accurate and balanced object detection.

### 2.3 Evaluation metric

Several metrics, including Mean Average Precision (mAP), Precision-Recall Curve, and F1-score, were used to evaluate the performance of the trained model. In this single-class detection task, mean average precision (mAP) simplifies to average precision (AP) since we only consider one object category. AP is calculated based on the Intersection over Union (IoU) [19] between predicted bounding boxes and ground

truth annotations. A high overlapping IoU indicates a good prediction. Precision and recall are calculated based on true positives (detections with IoU exceeding the threshold), false positives (detections with IoU below the threshold), and false negatives (detections with missed ground truth objects). F1 score is defined as a harmonic average of recall and precision. Based on the definitions, AP, Precision, Recall, and F1 score are shown in the Equations (1) - (3):

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \quad (1)$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \quad (2)$$

$$F1\ score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (3)$$

The detected bounding box is not only used to detect the microbubbles but also plays an important role in the post-processing phase, where the size of each microbubble can be measured based on the information provided by its bounding box. The Bounding box is a rectangle surrounding an object, specifying its position, class, and confidence score [20]. However, this work focuses on a case with a single pre-defined object class. This allows the output of a bounding box to be simplified as a vector containing the object's position coordinates, width, height, and confidence score.

### 3. RESULTS AND DISCUSSION

Our experiments were conducted on Google Colab's hardware platform by using Python version 3.10, which leverages Nvidia's T4 GPUs with 16GB of RAM and 8GB of VRAM. This powerful configuration ensured the smooth execution of our computational tasks, even for large datasets and complex models. For our model training, we used an image input size of 640x480x3 (width, height, and color channels). We also employed the Adam optimizer [21] and trained in batches of 32 images. The initial learning rate was set to 0.002. To prevent overfitting, we incorporated weight decay with a strength of 0.0005. Finally, we used an IoU threshold of 0.3 during training. The training process was stopped by early stopping to avoid overfitting at the 107th epoch (shown in Figure 2).

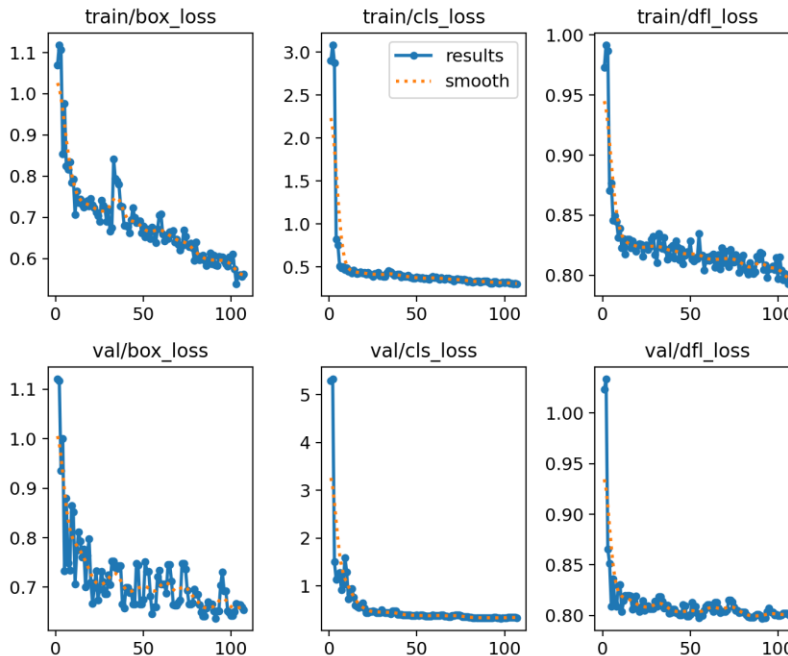


Fig. 2. Model training graphs.

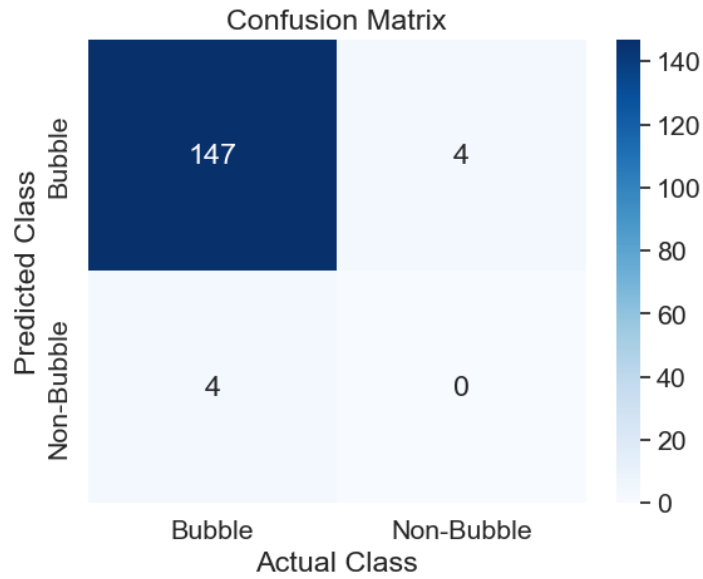


Fig. 3. Confusion matrix on the testing dataset.

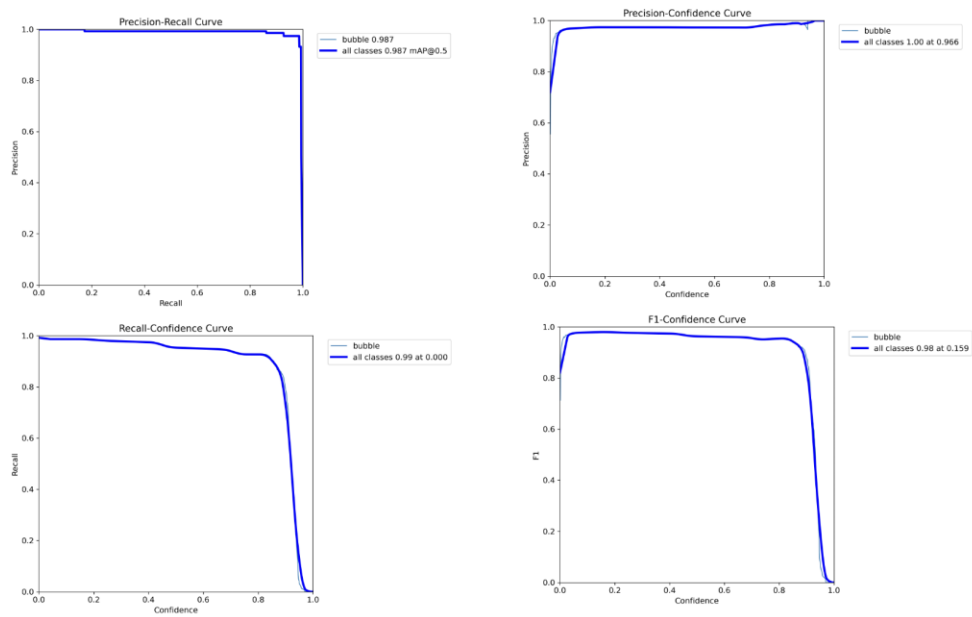
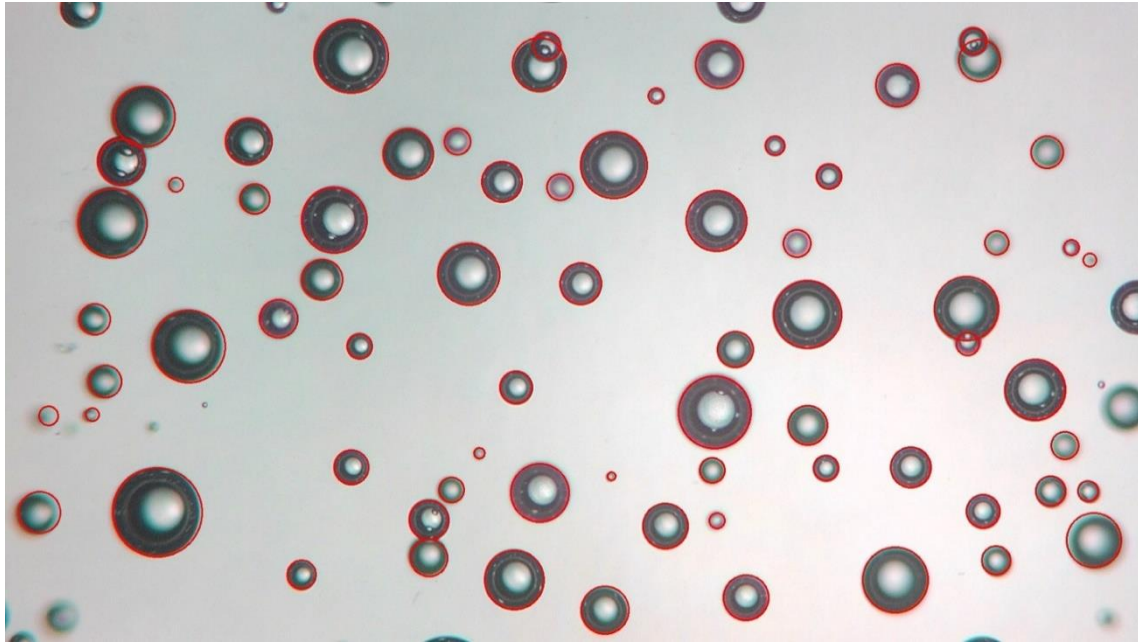
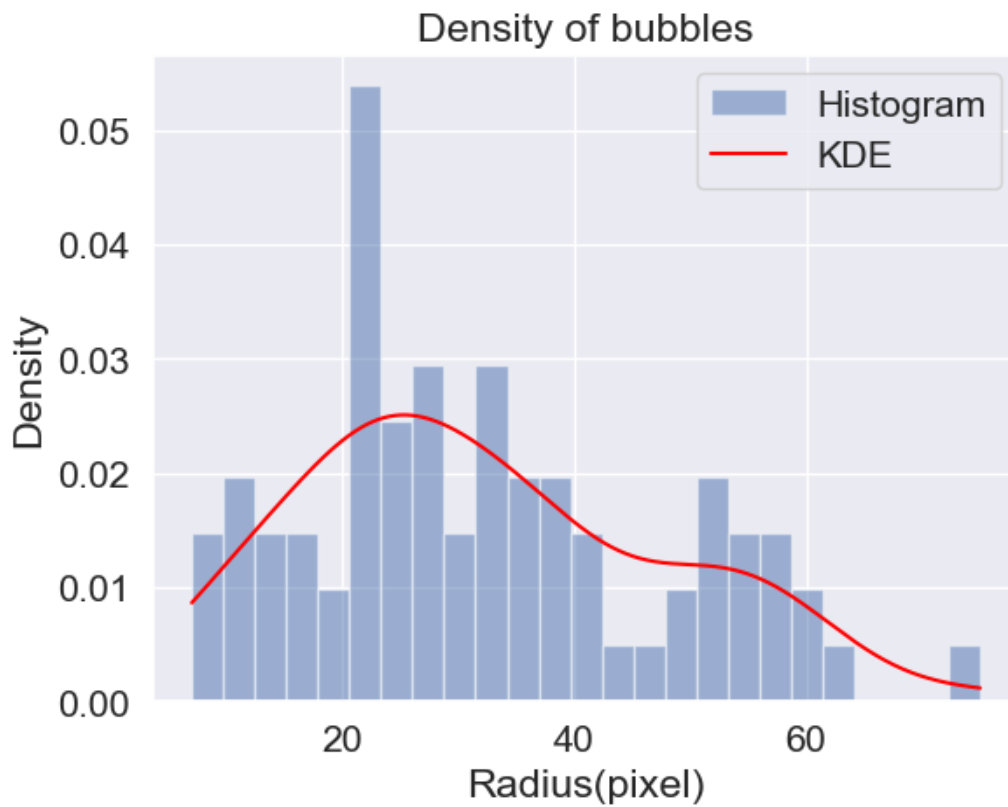


Fig. 4. Model's performance



**Fig. 5.** Microbubbles detected by model.



**Fig. 6.** Histogram of microbubble size with pixel scale.

A confusion matrix that summarizes the performance of the model on the test data is shown in (Figure 3). The model has been created to check whether the object belongs to the bubble class. In the given figure, the model has a considerable True Positive value and a low False Positive value, which indicates that our model is good enough to detect the bubble.

The prevalent Area Under the Precision and Recall curves (PR curves), which consist of precision on the vertical axis and recall on the horizontal axis, are used as a more objective indicator to quantify the model's performance due to the highly skewed domains. The higher the curve value, the better the model's performance. In this work, the AUC of the PR curve is 0.987, as shown in (Figure 4). In addition, our model has a good value of mAP, recall, F1 score, and precision, with respective values of 0.987, 0.99, 0.98, and 1.0.

We tested our trained model on a new image, and the visualization results are presented in Figure 5. The results indicate that our model detected 72 out of 82 bubbles, achieving an accuracy of 88.89%. In this figure, the detected bubbles are highlighted with red contours. The model performs particularly well in identifying overlapping and blurred bubbles. However, in more challenging cases such as tiny, heavily blurred, or incomplete bubbles, its detection performance is less effective. (Figure 6) illustrates the distribution of detected CO<sub>2</sub> microbubbles on this new image. The density plot combines the Kernel Density Estimation (KDE) [15], which provides a smoothed estimate and the histogram. This visualization reveals that the most frequent bubble radius falls around 20 pixels, with a range spanning from 10 to over 60 pixels.

#### 4. CONCLUSIONS

This study introduces an AI-assisted model for detecting CO<sub>2</sub> microbubbles, powered by YOLOv8 with high-accuracy predictions. The model achieved performance metrics of 0.987 for mean Average Precision (mAP), 0.99 for recall, 0.98 for the F1 score, and 1.0 for precision. These results demonstrate that the AI-assisted model is a valuable tool for microbubble analysis. The detection images processed using the YOLOv8 model illustrate its ability to accurately identify and classify microbubbles of varying sizes, generating bounding boxes around each detected bubble. This combination of data visualization and advanced detection techniques highlights the effectiveness of YOLOv8 in microbubble analysis, enabling precise measurement and characterization of bubble size distributions—an essential aspect for applications in chemical engineering processes.

#### 5. ACKNOWLEDGEMENTS

We acknowledge Ho Chi Minh City University of Technology (HCMUT), VNU-HCM for supporting this study.

#### Literature - References

1. Akai, T., Xue, Z., Yamashita, Y., and Yoshizawa, M. "Application of CO<sub>2</sub> micro bubble for the innovative CO<sub>2</sub>-EOR." Society of Petroleum Engineers - Abu Dhabi International Petroleum Exhibition and Conference, ADIPEC 2015, No. November, (2015).
2. Le, N. N. H., Sugai, Y., Vo-Thanh, H., Nguele, R., Ssebadduka, R., and Wei, N. "Experimental investigation on plugging performance of CO<sub>2</sub> microbubbles in porous media." Journal of Petroleum Science and Engineering, Vol. 211, No. November 2021, (2022), 110187. <https://doi.org/10.1016/j.petrol.2022.110187>
3. Nguyen Hai Le, N., Sugai, Y., Nguele, R., and Sreu, T. "Bubble size distribution and stability of CO<sub>2</sub> microbubbles for enhanced oil recovery: effect of polymer, surfactant and salt concentrations." Journal of Dispersion Science and Technology, (2021). <https://doi.org/10.1080/01932691.2021.1974873>
4. Nguyen Hai Le, N. N. H., Sugai, Y., and Sasaki, K. "Investigation of Stability of CO<sub>2</sub> Microbubbles—Colloidal Gas Aphrons for Enhanced Oil Recovery Using Definitive Screening Design." Colloids and Interfaces, Vol. 4, No. 2, (2020), 26. <https://doi.org/10.3390/colloids4020026>

5. Telmadarreie, A., Doda, A., Trivedi, J. J., Kuru, E., and Choi, P. "CO<sub>2</sub> microbubbles - A potential fluid for enhanced oil recovery: Bulk and porous media studies." *Journal of Petroleum Science and Engineering*, Vol. 138, (2016), 160–173. <https://doi.org/10.1016/j.petrol.2015.10.035>
6. Ruan, J., Zhou, H., Ding, Z., Zhang, Y., Zhao, L., Zhang, J., and Tang, Z. "Machine learning-aided characterization of microbubbles for venturi bubble generator." *Chemical Engineering Journal*, Vol. 465, No. March, (2023), 142763. <https://doi.org/10.1016/j.cej.2023.142763>
7. Jin, S., Ye, G., Guo, Y., Zhao, Z., Lu, L., and Liu, Z. "A real-time monitoring and measurement method for microbubble morphology based on image processing technology." *Microchemical Journal*, Vol. 203, No. May, (2024), 110881. <https://doi.org/10.1016/j.microc.2024.110881>
8. Dey, A., and Biswas, S. "Shot-ViT: Cricket Batting Shots Classification with Vision Transformer Network." *International Journal of Engineering Transactions C: Aspects*, Vol. 37, No. 12, (2024), 2463–2472. <https://doi.org/10.5829/ije.2024.37.12c.04>
9. Varghese, R., and M., S. "YOLOv8: A Novel Object Detection Algorithm with Enhanced Performance and Robustness." In *2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS)* (pp. 1–6). <https://doi.org/10.1109/ADICS58448.2024.10533619>
10. Rami, V., and Ch, R. "A Review on YOLOv8 and Its Advancements A Review on YOLOv8 and its Advancements," No. January, (2024). <https://doi.org/10.1007/978-981-99-7962-2>
11. Reis, D., Kupec, J., Hong, J., and Daoudi, A. "Real-Time Flying Object Detection with YOLOv8," (2023). Retrieved from <http://arxiv.org/abs/2305.09972>
12. Chen, Y., Xu, H., Chang, P., Huang, Y., Zhong, F., Jia, Q., Chen, L., Zhong, H., and Liu, S. "CES-YOLOv8 : Strawberry Maturity Detection Based on the Improved YOLOv8," (2024).
13. Amaral, D. De, Diego, E., Freitas, G. De, Abud, H. F., and Gomes, D. G. "Applying YOLOv8 and X-ray Morphology Analysis to Assess the Vigor of *Brachiaria brizantha* cv . Xaraés Seeds," (2024), 869–880.
14. He, K., Zhang, X., Ren, S., and Sun, J. "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition," (n.d.), 1–14.
15. Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. "Feature Pyramid Networks for Object Detection." In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 936–944). <https://doi.org/10.1109/CVPR.2017.106>
16. Terven, J., Córdova-Esparza, D. M., and Romero-González, J. A. "A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS." *Machine Learning and Knowledge Extraction*, Vol. 5, No. 4, (2023), 1680–1716. <https://doi.org/10.3390/make5040083>
17. Sun, X., Liu, T., Yu, X., and Pang, B. "Unmanned Surface Vessel Visual Object Detection Under All-Weather Conditions with Optimized Feature Fusion Network in YOLOv4." *Journal of Intelligent and Robotic Systems: Theory and Applications*, Vol. 103, No. 3, (2021). <https://doi.org/10.1007/s10846-021-01499-8>
18. Li, X., Wang, W., Wu, L., Chen, S., Hu, X., Li, J., and Tang, J. "Generalized Focal Loss : Learning Qualified and Distributed Bounding Boxes for Dense Object Detection," No. *NeurIPS*, (2020), 1–11.
19. Yu, J., Jiang, Y., Wang, Z., Cao, Z., and Huang, T. "UnitBox: An Advanced Object Detection Network." In *Proceedings of the 24th ACM International Conference on Multimedia* (pp. 516–520). New York, NY, USA: Association for Computing Machinery. <https://doi.org/10.1145/2964284.2967274>
20. Girshick, R., Donahue, J., Darrell, T., and Malik, J. "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation." In *2014 IEEE Conference on Computer Vision and Pattern Recognition* (pp. 580–587). <https://doi.org/10.1109/CVPR.2014.81>
21. D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," *CoRR*, vol. abs/1412.6980, 2014, [Online]. Available: <https://api.semanticscholar.org/CorpusID:6628106>