



Article

An Assessment of Human Inspection and Deep Learning for Defect Identification in Floral Wreaths

Diego Caballero-Ramirez ¹, Yolanda Baez-Lopez ¹, Jorge Limon-Romero ¹, Guilherme Tortorella ^{2,3,4} and Diego Tlapa ^{1,*}

¹ Facultad de Ingeniería, Arquitectura y Diseño, Universidad Autónoma de Baja California, Ensenada 21100, Mexico; diego.caballero@uabc.edu.mx (D.C.-R.); yolanda@uabc.edu.mx (Y.B.-L.); jorge.limon@uabc.edu.mx (J.L.-R.)

² Systems and Production Engineering Department, Universidade Federal de Santa Catarina, Florianopolis 88040, Brazil; gtortorella@bol.com.br

³ Department of Mechanical Engineering, The University of Melbourne, Melbourne 3052, Australia

⁴ IAE Business School, Universidad Austral, Buenos Aires 1611, Argentina

* Correspondence: diegotlapa@uabc.edu.mx; Tel.: +52-6461744333

Abstract: Quality assurance through visual inspection plays a pivotal role in agriculture. In recent years, deep learning techniques (DL) have demonstrated promising results in object recognition. Despite this progress, few studies have focused on assessing human visual inspection and DL for defect identification. This study aims to evaluate visual human inspection and the suitability of using DL for defect identification in products of the floriculture industry. We used a sample of defective and correct decorative wreaths to conduct an attribute agreement analysis between inspectors and quality standards. Additionally, we computed the precision, accuracy, and Kappa statistics. For the DL approach, a dataset of wreath images was curated for training and testing the performance of YOLOv4-tiny, YOLOv5, YOLOv8, and ResNet50 models for defect identification. When assessing five classes, inspectors showed an overall precision of 92.4% and an accuracy of 97%, just below the precision of 93.8% obtained using YOLOv8 and YOLOv5 with accuracies of 99.9% and 99.8%, respectively. With a Kappa value of 0.941, our findings reveal an adequate agreement between inspectors and the standard. The results evidence that the models presented a similar performance to humans in terms of precision and accuracy, highlighting the suitability of DL in assisting humans with defect identification in artisanal-made products from floriculture. Therefore, by assisting humans with digital technologies, organizations can embrace the full potential of Industry 4.0, making the inspection process more intelligent and reliable.

Keywords: floriculture; deep learning; visual inspection; human inspection; defect detection



Citation: Caballero-Ramirez, D.; Baez-Lopez, Y.; Limon-Romero, J.; Tortorella, G.; Tlapa, D. An Assessment of Human Inspection and Deep Learning for Defect Identification in Floral Wreaths. *Horticulturae* **2023**, *9*, 1213. <https://doi.org/10.3390/horticulturae9111213>

Academic Editor: Jianwei Qin

Received: 10 October 2023

Revised: 3 November 2023

Accepted: 6 November 2023

Published: 8 November 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Quality inspection in floriculture has been commonly conducted by humans; however, with the advancement of Industry 4.0, DL models have the potential to conduct real-time and remote inspection. Floriculture is a subdivision of horticulture that produces ornamental plants, flowers, and greenery [1]. Over the years, floriculture has become one of the most profitable in the agricultural sector; the global flower market is valued at around 44 billion US dollars annually [2,3]. As in other developing countries such as India [4,5], Mexican floriculture is a crucial activity that generates over 250,000 direct jobs and nearly one million indirect jobs, of which nearly 60% are female workers [2]. Handmade wreaths of preserved greenery are a type of product mainly destined for exportation, with quality and safety being prominent characteristics that must be revised during different production stages. Wreath manufacturers rely on visual inspection during the manufacturing process, aligning with traditional horticulture inspection to visually examine plants for signs of disease, pests, and other characteristics. However, this manual approach is a tiring activity

prone to human error [6] that can be time-consuming, subjective, and limited by the knowledge and skills of the individual inspector. Studies in the manufacturing sector reported an accuracy of 80% to 85% for correctly rejected precision-manufactured parts [7]. Inspection error rates vary depending on many factors [7], with values of 20% to 30% commonly identified in the literature [8] and with variations depending on the inspection activity; for example, the error rate in inspections of highway bridges ranges from 48% to 19% [9], while in metal casting inspection, it ranges from 17.8% to 29.8% [10], indicating that the reliability and accuracy of visual inspection frequently prove insufficient [11]. Despite the importance of visual inspection in floriculture, particularly in floral wreath manufacturing, there is scarce evidence regarding the accuracy level or error rates in this sector, which leads us to the first research question.

RQ1. What is the precision and accuracy level of defect identification in floral wreath using visual inspection?

The growing interest in recent years for utilizing digital technologies in horticulture includes robots [12], digital twins [13], internet of things [14], artificial intelligence (AI) [15], including machine learning [16], and DL [17]. Notably, the inherent dependence on visual inspection has fostered the increasing use of DL techniques due to the positive effects and promising contribution to improving visual inspections. Several cases of DL have been deployed for a variety of purposes, including disease detection in fruits [18], healthy flower detection [4], flower recognition [5], pest recognition [17], surface defect detection [19], and fruit grading [20]. Despite these efforts, few studies on defect identification of floral wreaths have been conducted, particularly in developing countries; therefore, we proposed the following research question.

RQ2. What is the precision and accuracy level of defect identification in floral wreaths using deep learning?

The increasing utilization of DL encompasses many possibilities to digitalize the floriculture and horticulture sector, particularly in derived products such as floral wreaths. However, most of the studies mainly focused on the advantages of DL and the computational efficiency metrics of different architectures and less on the actual accuracy of visual inspection, suggesting a research gap to be bridged. To the best of our knowledge, this research is the first to investigate the suitability of DL techniques within the inspection process of artisanal-made products, specifically those designed for ornamental purposes. This research marks a significant departure from traditional manual inspection methods by leveraging the power of AI, including quality control and human-dependence reduction, particularly in hand-made products in developing countries. This novel contribution holds promise in enhancing the efficiency and reliability of inspection procedures, ultimately benefiting the economic growth and global competitiveness of these countries' artisanal industries.

2. Related Work

The main challenging problem in computer vision is object detection [21] due to the complexity of recognizing objects and localizing them within the image. With the continuous updating of neural networks, different models are used to complete different situations. Table A1 (Appendix A) summarizes the most recurrent architectures for object detection in agriculture and horticulture applications. For inspection purposes, object detection algorithms, such as region-based classification, create a bounding box around the region of interest (ROI), such as density and color, thus detecting a defect or not. Broadly, standard methods are classified into single-stage and two-stage object detectors [21]. The latter group includes Region-based Convolutional Neural Networks (R-CNNs), Fast R-CNN [22], Faster R-CNN [23], and Mask R-CNN [24]. The former group includes YOLO (You Only Look Once) [25] and single-shot multi-box detectors [26]. Although two-stage detectors have demonstrated ideal accuracy, their detection is not very fast [27]. One-stage detectors overcome these limitations, with YOLO and SSD being very popular due to being faster than two-stage deep-learning object detectors and requiring less time for model training [28].

Since its release, the YOLO network has been used in several cases, including grape detection [29,30]; potted flower detection [31]; tomato, flower, and node detection [32]; pineapple surface defect detection [28]; sugarcane stem node recognition [33]; and tomato growth period tracking [34]. The YOLO network has stood out in the average recognition speeds and accuracy for occluded grapes compared to Resnet50 and SSD300 [29].

ResNet has been a recurrent model for multi-stage networks due to the resulting accuracy and the obtained balance between accuracy and training cost [35]. ResNet is based on stacked residual units consisting of convolution and pooling layers [36]. There are different versions, including ResNet18, ResNet34, ResNet50, Resnet10, and ResNet152, with the last three being reported as more accurate [36]. Particularly, ResNet50 stands out as a popular version, achieving maximum validation accuracy among several pre-trained models [37] as well as training accuracy [38], thus outperforming similar networks such as ResNet101 [39] and other network architectures including AlexNet, GoogLeNet, Inception v3, ResNet-101, and SqueezeNet [35]. In fact, among these architectures, ResNet-50 struck the best balance between training cost and accuracy [35].

We used both one-stage and two-stage networks to identify the network suitable for visual inspection purposes in horticulture contexts. Therefore, after conducting an evaluation of multiple object detection architectures and due to the good results of previous studies [28–31,33,34], we used YOLO networks, particularly YOLOv4-tiny, YOLOv5, and YOLOv8. YOLOv4-tiny is the compressed version of YOLOv4 designed to train on machines with less computing power [40,41]. YOLOv4, an improved version of YOLOv3, generates bounding-box coordinates and assigns probabilities to each category, converting the object detection task into a regression problem [42]. Continuing with the YOLO family, Ultralytics proposed the YOLOv5 algorithm [43], which is smaller and more convenient, enabling flexible deployment and more accurate detection [44]. YOLOv8 represents the most recent iteration of the YOLO object detection model [45,46]. It incorporates numerous enhancements, including a novel neural network architecture that leverages both the Feature Pyramid Network (FPN) and the Path Aggregation Network (PAN) [47], alongside the implementation of an improved labeling tool designed to streamline the annotation process.

Moreover, to provide a comprehensive comparison with the YOLO networks and due to the proven results in previous studies [35,37,39], we also used ResNET50. This inclusion aims to enhance the validity and reliability of the findings, adding perspective to the evaluation. Moreover, by considering multiple architectures, the evaluation aimed to strike the right balance between accuracy, speed, and resource efficiency.

3. Materials and Methods

3.1. Decorative Wreaths

This study analyzed the visual inspection of wreaths produced by a Mexican company and exported to the U.S.A. These wreaths are used as a base to embed flowers and other ornamental elements. The process starts with branches of Ephedra Californica that operators cut and assemble manually. The quality assurance of wreaths consists of visual inspections to identify whether wreaths comply with the quality standard or have defects based on predefined thresholds. To assess the suitability of the DL approach, we first assessed four different models; then, we assessed inspectors using an attribute agreement analysis.

3.2. Deep Learning Approach

3.2.1. Tools

We assessed the one-stage architectures YOLOv4-Tiny, YOLOv5, and YOLOv8 to identify five classes of wreaths, including correct wreaths and four types of defects on wreaths. In addition, as a complementary analysis, we included ResNet50 to identify either correct or defective wreaths. The evaluation aimed to balance precision, accuracy, speed, and resource efficiency by considering multiple architectures. For this analysis, we used a computing system consisting of an AMD Ryzen 5 4600H Radeon Graphics processor running at a clock speed of 3.00 GHz, accompanied by 32 GB RAM. The operating system was the 64-bit

version of Windows. An NVIDIA GeForce GTX1660 Ti graphics card was incorporated to enhance computational performance. A Logitech C920 Pro HD Webcam captured images at 1920×1080 pixels. Furthermore, the analysis included the utilization of Google Colab to assess its performance and suitability as a viable solution for small- and medium-sized enterprises (SMEs) that may not have access to more sophisticated computing equipment or the necessary resources. The setup of Google Colab Pro was configured to utilize the A100 GPU option, with 20 GB of GPU RAM and 40 GB of CPU RAM.

3.2.2. Dataset

A total of 3408 images were used for the dataset. The number of wreath images was balanced to represent correct and defective wreaths, corresponding to specific quality criteria, including lack of material (empty), incorrect color (brown), low volume (Lvolume), high volume (Hvolume), and correct wreath (Vok). Due to our relatively small dataset, we used data augmentation as a way to reduce overfitting [48] and to increase the dataset [48]. Particularly, we used different data augmentation methods, including geometric and color transformations [48], such as horizontal and vertical flip, rotation $\pm 15^\circ$, grayscale 10%, magenta filter 36%, and brightness 0% to +25%. Similar to previous studies [49], the dataset was partitioned, allocating close to 70% of the images for training the models (2556 images) while reserving the remaining 30% for validation and testing purposes (852 images). This ratio is also suggested to avoid overfitting [50] in relatively small datasets like ours. We used different wreaths for training and testing images at a similar angle to the inspector view angle. Regarding the background of the images, we intentionally imitated the actual inspection conditions by utilizing different solid backgrounds. In addition, to encompass different conditions, the images were captured using different filters on different days, at various times throughout the day, thus introducing variability to ensure the robustness of the models when presented with diverse environmental conditions (see Figure 1). This intentional introduction of noise aimed to enhance the models' ability to adapt to real-world scenarios and improve performance by training on a more comprehensive and representative dataset and mimicking the inspection conditions. The wreath dataset is available at Kaggle [51].

3.2.3. Experimental Strategy

The models based on YOLOv4-Tiny, YOLOv5, and YOLOv8 were trained using the five different classes utilizing bounding boxes to identify defects visually. In addition, we focused on identifying either correct or defective wreaths; thus, all models, YOLOv4-Tiny, YOLOv5, and YOLOv8, as well as ResNet50 were trained to identify these two classes. ResNet50 was used when classifying two classes due to the requirement of ResNet50 for a large dataset of labeled images per class to achieve reasonable performance [52,53]. Moreover, we trained all architectures with varying epochs (10, 50, and 100) to identify the optimal training epoch, which defines the saturation stage of the training process and when the model begins to flatten out [54]. The inference speed of each architecture was also considered to evaluate the trade-off between accuracy and computational efficiency.

3.2.4. Evaluation Indexes for the Deep Learning Models

The performance indexes of the object detection models employed in this study encompassed accuracy, precision, recall, and mean average precision. The results were assessed as true positive (TP), i.e., the number of correctly detected objects; true negative (TN), i.e., samples that are correctly rejected from class; false positive (FP), i.e., the number of falsely detected objects; and false negative (FN), i.e., the number of missed objects.

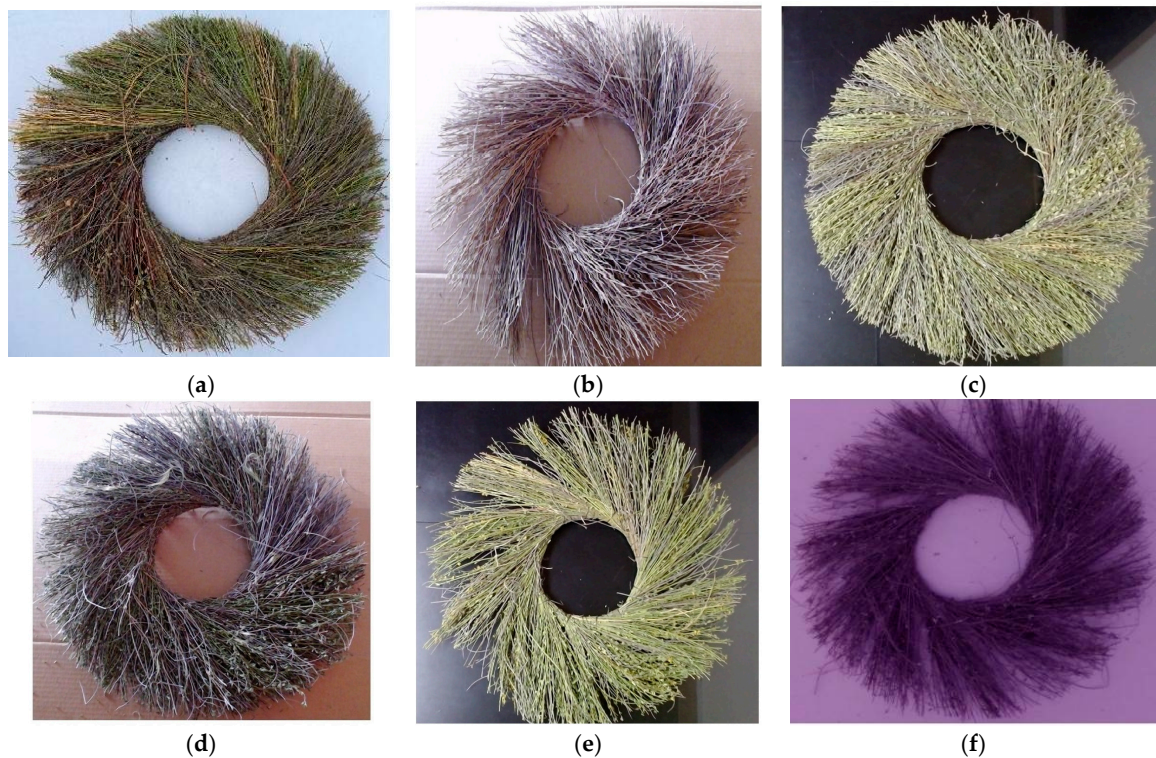


Figure 1. Example of wreath images with different classes, backgrounds, and light conditions: (a) correct wreath with white background and normal inspection light condition; (b) wreath with incorrect color, lack of material with poor light condition, and brown background; (c) wreath with low volume, black background, and normal inspection light condition; (d) wreath with high volume, brown background, and normal inspection light condition; (e) wreath with lack of material, black background, and light from a different angle; (f) wreath with lack of material, white background, normal light condition, and magenta filter.

Accuracy indicates the rate of correctly classified images out of all the images in a test set, showing the overall effectiveness of the classifier [49,55,56]; thus, accuracy shows the level of a model to predict the class of a pre-labeled image [57]. There are different approaches to measuring accuracy, including those used in [20] or [35]. In this study, we estimated accuracy as in Equation (1).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

Precision, also known as per-class precision [58] or positive predictive value (PPV) [35], represents the proportion of true positive images among all images predicted to be positive [29,34,37,55,56]. Precision is the probability, given a positive label, of how many of them are actually positive [49], indicating the performance of a model to predict the positive class [57]. Therefore, the higher the precision is, the fewer false positives are generated by the model. Equation (2) shows the formula used in this study.

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

Recall represents the proportion of images predicted to be positive among the true positive images [29,37,56]. Thus, recall or sensitivity is the accuracy of positively predicted instances describing how many were labeled correctly [49], showing the degree of the

model to predict the positive class when the actual class is positive [57]. In this study, we computed recall as in Equation (3).

$$\text{Recall} = \frac{TP}{TP + FN} \quad (3)$$

Mean average precision (mAP) measures in what percentage the algorithm predicted the object from all individual classes correctly [50]. mAP is calculated using the averaged AP of all classes [59]. Equation (4) presents the mathematical description of mAP , where AP_k is the average precision of the classes and n is the number of classes [60].

$$mAP = \frac{1}{n} \sum_{k=1}^{k=n} AP_k \quad (4)$$

3.3. Human Visual Inspection

Three quality inspectors conducted the quality inspection of wreaths. Due to the nominal response and the number of inspectors, we conducted an attribute agreement analysis focused on assessing whether inspectors are consistent with themselves, with one another, and with known standards of defective and non-defective wreaths.

3.3.1. Visual Inspection Procedure

The quality inspectors (males averaging five years of experience) independently inspected 50 wreaths three times in random order. The sample of wreaths was balanced to represent correct and defective wreaths (including up to four types of defects) according to the predefined quality criteria. Each inspector did not know the standard value for each wreath. The inspection procedure followed standardized protocols and guidelines to ensure consistency and reliability. The participants were provided clear instructions and a checklist to facilitate the assessment process.

3.3.2. Evaluation Indexes for Visual Inspection

To obtain comparative metrics with those utilized for the DL approach, we computed accuracy and precision as in Equations (1) and (2), respectively. In addition, we calculated standard metrics related to attribute agreement analysis (AAA), including Kappa statistics, which assesses the correctness of each inspector's ratings vs. the standard, the consistency of each inspector's rating, and the consistency between the inspectors' ratings. Due to the number of inspectors, we could not compute Cohen's kappa, which is utilized for two raters [61]. Instead, we computed Fleiss' kappa [62], a generalization of Cohen's kappa. Values of kappa range from 1 (perfect agreement) to -1 (perfect disagreement), with 0 representing a completely random agreement. Interpretive guidelines suggest a slight agreement (0–0.20), fair agreement (0.21–0.40), moderate agreement (0.41–0.60), substantial agreement (0.61–0.80), and almost perfect agreement (0.81–1.0) [63]. We used Minitab software, version 18.1 (developed by Minitab LLC, State College, PA, USA), to compute these statistics.

4. Results

4.1. Deep Learning Approach

Regarding the use of deep learning for visual inspection, Table 1 shows the results of the evaluated models when assessing five classes, resulting in a performance that varied across different epochs. Most of the higher results occurred with 100 epochs for all models. Regarding YOLOv5, the model exhibited a precision of 93.8%, accuracy of 98.2%, recall of 95.9%, and mAP of 97% in 1.33 h. Similarly, YOLOv8 achieved a precision of 93.8%, recall of 94.8%, and mAP of 95.7% in an execution time of 0.37 h. The more considerable accuracy for YOLOv8 was 93.9% with 50 epochs. Following these models, YOLOv4 Tiny resulted in a precision of 89%, accuracy of 88.7%, recall of 87%, and mAP of 95.5% in 0.48 h. Figure 2 depicts examples of the detection achieved with YOLOv8 using 100 epochs.

Table 1. Results of architectures YOLOv4-Tiny, YOLOv5, and YOLOv8.

Architecture	Epochs	Accuracy	Precision	Recall	<i>mAP</i>	Time (h)
YOLOv4-Tiny	10	0%	2.0%	0%	0.9%	0.10
YOLOv4-Tiny	50	75.4%	74.0%	76.0%	86.4%	0.23
YOLOv4-Tiny	100	88.7%	89.0%	87.0%	95.5%	0.48
YOLOv5	10	92.9%	85.7%	86.1%	90.6%	0.15
YOLOv5	50	94.9%	93.1%	93.2%	95.4%	0.67
YOLOv5	100	98.2%	93.8%	95.9%	97.0%	1.33
YOLOv8	10	88.1%	82.3%	86.3%	89.4%	0.12
YOLOv8	50	93.9%	93.6%	99.6%	94.8%	0.37
YOLOv8	100	92.8%	93.8%	94.8%	95.7%	0.37

**Figure 2.** Example of the detection with YOLOv8 with 100 epochs.

Regarding classes, Table 2 depicts the results where both YOLOv8 and YOLOv4 Tiny achieved slightly higher precision values than YOLOv5; particularly for the Ok class, the precision was 100% with both models. For the high-volume class, YOLOv4 exhibited a precision of 99.9%, while for the low-volume class, YOLOv8 was the model with a higher precision of 99.5%. For brown and empty, YOLOv4 presented a precision of 99.9% and 85.1%, respectively—the highest results. Concerning accuracy, YOLOv4 exhibited a value of 89.5%, while YOLOv5 showed values of 99.6%, 99.3%, and 99.5% for brown, low volume, and high volume, respectively. The highest accuracy result for the Ok class was 99.3% obtained using YOLOv8. Recall presented higher results in three classes (brown, low volume, and Ok) with YOLOv5, one class (high volume) with YOLOv8, and one class (empty) with YOLO-v4. *mAP* showed similar values in all classes when using YOLOv8 and YOLOv5. The execution time for YOLOv4 (0.483 h) was consistently lower than those of YOLOv5 (1.331 h) and YOLOv8 (1.152 h).

The loss curve for YOLOv4-tiny for five classes is shown in Figure 3. The training loss decreases steadily, while the validation loss remains relatively stable. The *mAP* curve shows that the training and validation sets increase over time.

Table 2. Comparative table of architectures: YOLOv4-tiny, YOLOv5, and YOLOv8 with classes.

Architecture	Class	Epochs	Accuracy	Precision	Recall	mAP	Time (h)
YOLOv4-Tiny	Empty	100	89.5%	85.1%	90%	86.5%	0.483
YOLOv4-Tiny	Brown	100	90.1%	99.9%	89.1%	98.5%	0.483
YOLOv4-Tiny	Lvolume	100	79.8%	92.5%	79.6%	91.4%	0.483
YOLOv4-Tiny	Hvolume	100	90.4%	99.9%	88.5%	99.5%	0.483
YOLOv4-Tiny	Vok	100	95.4%	100.0%	84.5%	99.5%	0.483
YOLOv5	Empty	100	78.4%	80.1%	80.9%	87.9%	1.331
YOLOv5	Brown	100	99.6%	94.2%	100%	98.5%	1.331
YOLOv5	Lvolume	100	99.3%	99.0%	100%	99.5%	1.331
YOLOv5	Hvolume	100	99.5%	95.7%	98.4%	99.4%	1.331
YOLOv5	Vok	100	99.1%	99.9%	100%	99.5%	1.331
YOLOv8	Empty	100	76.7%	75.9%	75.3%	80.8%	1.152
YOLOv8	Brown	100	99.1%	98.1%	99.8%	99.0%	1.152
YOLOv8	Lvolume	100	99.2%	99.5%	99.4%	99.5%	1.152
YOLOv8	Hvolume	100	98.7%	95.4%	99.6%	99.5%	1.152
YOLOv8	Vok	100	99.3%	100.0%	99.8%	99.5%	1.152

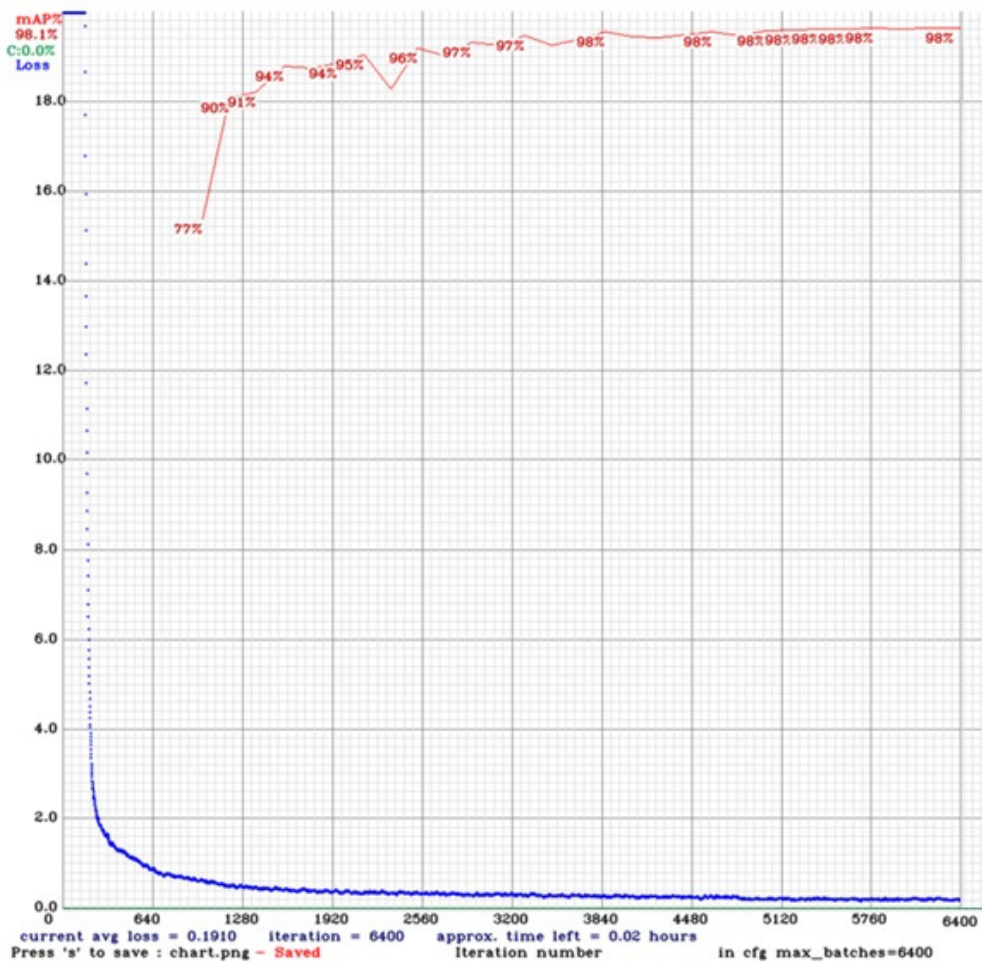


Figure 3. Training loss and mAP for YOLOv4-tiny for five classes.

The training and validation loss curves for YOLOv5 are depicted in Figure 4. Training losses decrease rapidly in the first few epochs and more slowly thereafter, indicating the model’s ability to learn the basic features of the dataset quickly. However, it takes more time to learn the finer details. The precision and recall on the training set are both high, while the validation losses are slightly higher than the training losses; thus, the model is not overfitting the training data.

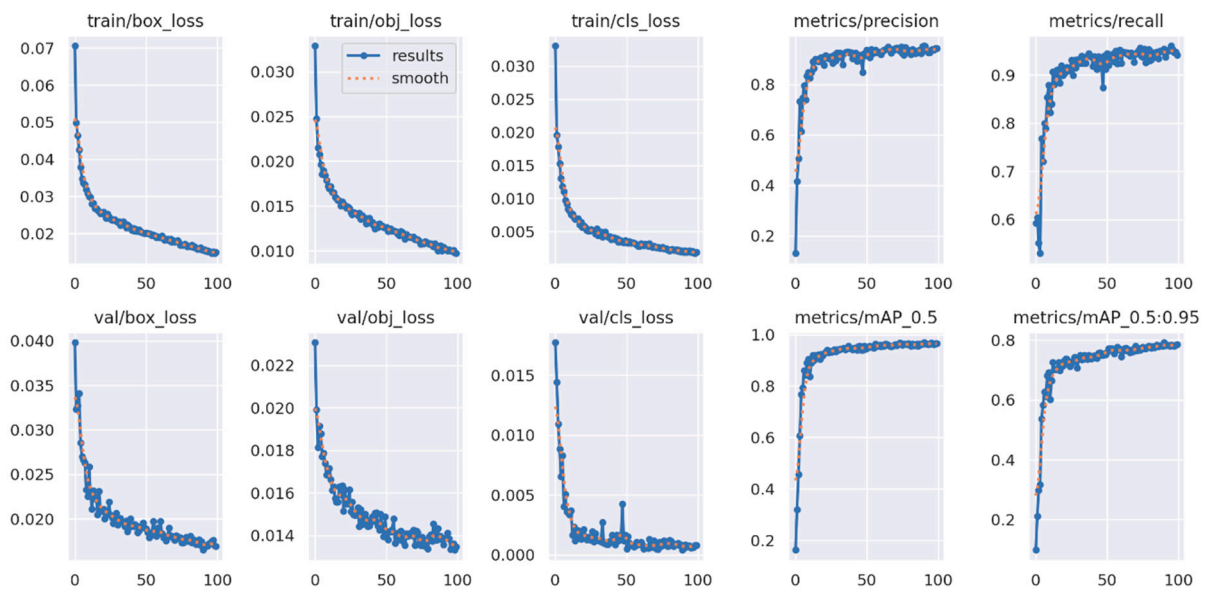


Figure 4. Training and validation loss graphs of YOLOv5 for 5 classes.

The training and validation loss curves for YOLOv8 are depicted in Figure 5. Similarly to YOLOv5, the model is able to learn the basic features of the dataset quickly, but it takes more time to learn the finer details. The precision and recall on the training set are both high. At the same time, the validation losses are slightly higher than the training losses, indicating that the model is not overfitting the training data. The *mAP* on the validation set is high, which suggests that the model is able to generalize well to unseen data. Overall, the graphs suggest that the YOLOv8 model has trained successfully on the Wreath dataset. The model is able to detect objects in the dataset with reasonable accuracy, and it is likely to generalize well to new data.

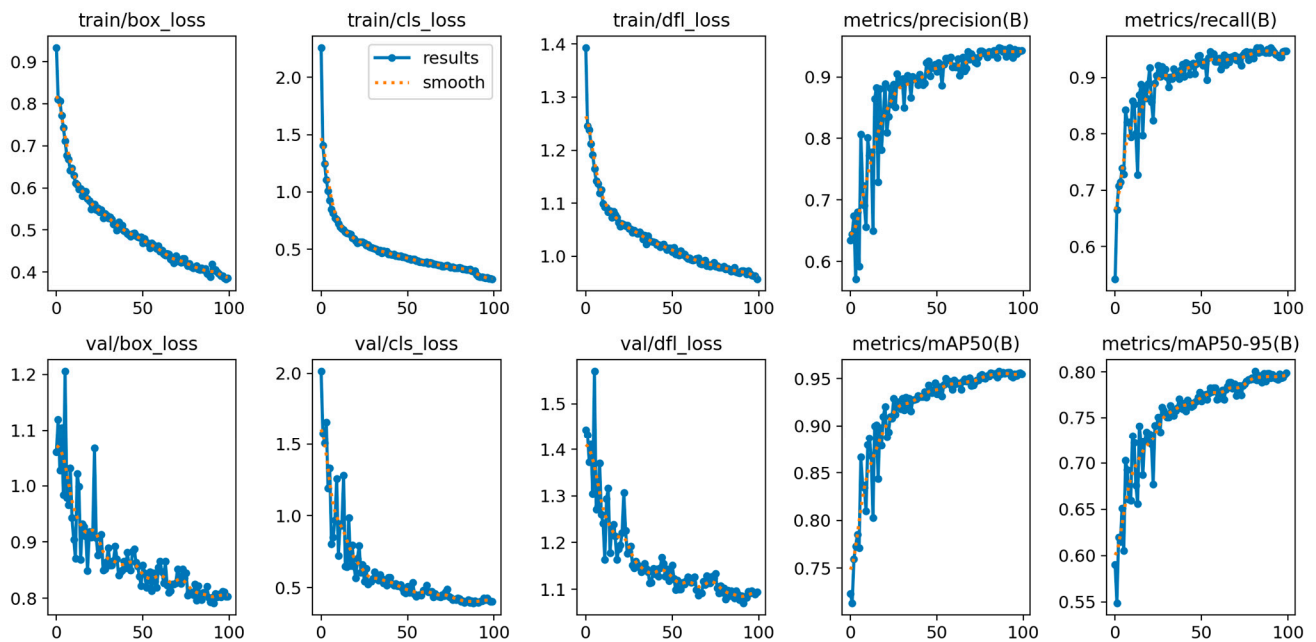


Figure 5. Training and validation loss graphs of YOLOv8 for 5 classes.

Finally, all models exhibited adequate performance with 100 epochs evaluating two classes (Ok and Not ok), as shown in Table 3. All three one-stage models exhibited a higher accuracy and precision than the two-stage model did (ResNet50). YOLOv8 achieved a

precision of 99.8%, accuracy of 99.9%, recall of 100%, and *mAP* of 99.5% in 1.576 h. YOLOv5 achieved a precision of 99.7%, accuracy of 99.8%, recall of 99.9%, and *mAP* of 99.5% in 1.379 h. In turn, YOLOv4-Tiny achieved a precision of 99%, accuracy of 99.6%, recall of 100%, and *mAP* of 100% in an execution time of 0.55 h. Lastly, ResNet achieved a precision of 96.4%, accuracy of 96.9%, recall of 97.1%, and *mAP* of 96.4% in 1 h.

Table 3. Comparative table of architectures YOLOv4-Tiny, YOLOv5, YOLOv8, and ResNet50.

Architecture	Epochs	Accuracy	Precision	Recall	<i>mAP</i>	Time (h)
YOLOv4-Tiny	10	0.0%	0.3%	0.0%	4.7%	0.067
YOLOv4-Tiny	50	97.5%	90.0%	98.0%	98.8%	0.267
YOLOv4-Tiny	100	99.6%	99.0%	100.0%	100.0%	0.550
YOLOv5	10	99.5%	98.8%	98.8%	99.3%	0.138
YOLOv5	50	99.4%	99.6%	99.8%	99.4%	0.700
YOLOv5	100	99.8%	99.7%	99.9%	99.5%	1.379
YOLOv8	10	99.7%	99.5%	99.7%	99.4%	0.158
YOLOv8	50	99.8%	99.7%	99.9%	99.4%	0.799
YOLOv8	100	99.9%	99.8%	100.0%	99.5%	1.576
ResNet50	10	91.4%	94.3%	96.4%	94.3%	0.117
ResNet50	50	95.9%	94.9%	96.3%	94.9%	0.600
ResNet50	100	96.9%	96.4%	97.1%	96.4%	1.000

Regarding the results for each class (Ok and Not ok), YOLOv4-tiny, YOLOv5, and YOLOv8 presented a consistent precision and accuracy, as seen in Table 4. However, YOLOv4-tiny exhibited 100% for both precision and accuracy for Not ok and Ok classes. ResNet50 followed with a precision of 95.1% and 95.4% for Not ok and Ok, respectively, while the accuracy for Not ok and Ok classes was 97% and 96.6%, respectively. Recall and *mAP* were similar for YOLOv4-tiny, YOLOv5, and YOLO v8, followed by ResNet50.

Table 4. Architectures assessing two classes.

Architecture	Class	Epochs	Accuracy	Precision	Recall	<i>mAP</i>
YOLOv4-Tiny	Not ok	100	100.0%	100.0%	100.0%	100.0
YOLOv4-Tiny	Ok	100	100.0%	100.0%	100.0%	100.0
YOLOv5	Not ok	100	99.9%	99.7%	100.0%	99.5
YOLOv5	Ok	100	99.9%	99.8%	99.8%	99.4
YOLOv8	Not ok	100	99.7%	99.8%	100.0%	99.5
YOLOv8	Ok	100	99.9%	99.8%	99.9%	99.5
ResNet50	Not ok	100	97.0%	95.1%	95.1%	94.9
ResNet50	Ok	100	96.6%	95.4%	95.4%	96.3

4.2. Human Visual Inspection

The human visual inspection analysis resulted in an overall precision of 92.4%. Regarding each inspector, inspectors B and C exhibited a precision of 95.3%, while inspector A showed a precision of 86.7%, as seen in Table 5. Similarly, the overall accuracy was 97%, while for inspectors, it ranged from 94.7% for inspector A to 98.1% for B and C. The recall was similar for inspectors B and C, while A presented lower values.

Regarding the classes, inspectors exhibited a precision of 100% for both low-volume wreaths and Ok wreaths and 98.7% for the empty class (see Table 6). The precision for brown and high-volume classes was 81.4% and 86.3%, respectively. Regarding accuracy, all five classes presented values from 94.2% to 100%.

Table 5. Total scores obtained by the inspectors.

Inspector	Precision	Accuracy	Recall
A	86.7%	94.7%	86.7%
B	95.3%	98.1%	95.3%
C	95.3%	98.1%	95.3%
Overall	92.4%	97.0%	92.4%

Table 6. Scores per class obtained by all the inspectors.

Class	Precision	Accuracy	Recall
Empty	98.7%	96.2%	82.2%
Brown	81.4%	94.2%	92.2%
Lvolume	100%	98.0%	90%
Hvolume	86.3%	96.4%	97.8%
Vok	100%	100%	100%

When assessing two classes (defective wreath and correct wreath), the inspectors exhibited an inspection precision and accuracy value of 100% for both defective and correct wreaths.

Attribute Agreement Analysis

For the attribute agreement analysis, the percentage of matches of each inspector vs. the standard ranged from 70% to 94%; for all inspectors vs. the standard, the percentage was 68% (with a 95% CI of 53.3% to 84.48%), as seen in Table 7. Fleiss' Kappa statistics for each inspector vs. the standard are depicted in Table 8, highlighting values ranging from 0.582 to 1, resulting in an overall Kappa value of 0.941 (p -value of 0.000).

Table 7. Assessment agreement between the inspectors and the standard.

Inspector	Inspected Number	Matched Number	Percentage	95% CI
A	50	35	70%	(55.4%, 82.1%)
B	50	47	94%	(83.4%, 98.7%)
C	50	45	90%	(78.2%, 96.7%)
Overall	50	34	68%	(53.3%, 80.4%)

Table 8. Fleiss' Kappa statistics of each inspector vs. the standard.

Inspector	Class	Kappa	SE Kappa	Z	p -Value
A	Brown	0.737	0.081	9.031	0.000
	Empty	0.582	0.081	7.129	0.000
	High volume	0.848	0.081	10.393	0.000
	Low volume	0.911	0.081	11.163	0.000
	Ok	1.000	0.081	12.247	0.000
	Overall	0.830	0.041	20.121	0.000
B	Brown	0.883	0.081	10.820	0.000
	Empty	0.935	0.081	11.451	0.000
	High volume	0.979	0.081	12.001	0.000
	Low volume	0.911	0.081	11.163	0.000
	Ok	1.000	0.081	12.247	0.000
	Overall	0.941	0.040	23.033	0.000
C	Brown	0.883	0.081	10.826	0.000
	Empty	0.978	0.081	11.982	0.000
	High volume	0.867	0.081	10.630	0.000
	Low volume	0.978	0.081	11.982	0.000
	Ok	1.000	0.081	12.247	0.000
	Overall	0.941	0.040	23.012	0.000

Regarding the all-inspectors vs. standard, the overall Fleiss Kappa statistic was 0.904 (p -value of 0.000), indicating a good level of absolute agreement of the assessments between all inspectors and the standard, as seen in Table 9. Notably, all inspectors showed adequate Kappa values for each class; the lower value was 0.834 for the brown and empty class, while the Ok class obtained a value of 1.0. Table 10 depicts the results regarding the agreement between inspectors, highlighting an overall Kappa value of 0.86 (p -value of 0.000).

Table 9. Fleiss' Kappa statistics of the inspectors vs. the standard.

Class	Kappa	SE Kappa	Z	p -Value
Brown	0.834	0.047	17.711	0.0000
Empty	0.831	0.047	17.645	0.0000
High volume	0.898	0.047	19.066	0.0000
Low volume	0.933	0.047	19.808	0.0000
Ok	1.000	0.047	21.213	0.0000
Overall	0.904	0.023	38.190	0.0000

Table 10. Fleiss' Kappa statistics between inspectors.

Class	Kappa	SE Kappa	Z	p -Value
Brown	0.755	0.023	32.072	0.0000
Empty	0.792	0.023	33.601	0.0000
High volume	0.838	0.023	35.568	0.0000
Low volume	0.924	0.023	39.232	0.0000
Ok	1.000	0.023	42.426	0.0000
Overall	0.860	0.011	72.618	0.0000

5. Discussion

This study assessed visual human inspection and deep learning models to detect objects and classify defects in decorative wreaths. The results indicated a similar performance between inspectors and models. When assessing five classes, inspectors showed an overall precision of 92.4%, just below the precision of 93.8% obtained with both YOLOv8 and YOLOv5. The accuracy obtained by the inspectors was 97%, while YOLOv5 exhibited an accuracy of 98.2% with 100 epochs, YOLOv8 obtained 93.9% with 50 epochs, and YOLOv4-tiny obtained 88.7% with 100 epochs. Concerning each class, both inspectors and algorithms presented mixed results. The inspectors obtained a larger precision than the algorithms did when assessing empty and low-volume wreaths; however, YOLOv4-tiny showed a greater precision than the inspectors did for brown and high-volume wreaths. Finally, for Ok wreaths, both inspectors and algorithms achieved a precision of 100%.

Regarding accuracy, the inspectors exhibited higher values than the algorithms did for empty, low-volume, and Ok classes. However, for classes brown and high volume, YOLOv5 presented higher accuracy values.

When assessing two classes, the inspectors exhibited a high precision, similar to all three YOLO models, while ResNet50 showed slightly inferior values. Regarding accuracy, the inspectors achieved an overall inspection accuracy value of 100% for both defective and correct wreaths, while the algorithms showed an overall accuracy of 94.5% and 97.1%, respectively.

5.1. Deep Learning Models

The findings highlight the potential of YOLOv4-tiny, YOLOv5, and YOLOv8 in accurately detecting and categorizing specific quality criteria associated with decorative wreaths. These three models were trained with five classes, allowing for a more comprehensive assessment of the wreaths' quality. YOLOv8 and YOLOv4-tiny achieved high precision across different classes. While YOLOv5 demonstrated a slightly lower precision when identifying all five classes, it presented higher accuracy and recall results. Thus, all

three one-stage models are adequate options, indicating their ability to precisely and accurately identify the different quality criteria of wreaths beyond a binary classification, thus representing a potential advantage in agricultural inspection, where multiple parameters contribute to overall product quality. Regarding assessing two classes, all one-stage models exhibited a larger precision and accuracy than ResNet50, thus confirming that YOLOs performed meaningfully better than two-stage object detectors in inference time and detection accuracy [21], due to the condition for a large dataset of labeled images per class to achieve reasonable performance [52,53].

All three single-stage models exhibited numerous precision, accuracy, or recall values greater than 99% when assessing five or two classes. Such metrics can suggest potential overfitting risks; however, these results are in line with previous studies that also prevented overfitting when using YOLOv4 [64,65], YOLOv5 [66–68], or YOLOv8 [69]. Overfitting is a fundamental issue in supervised machine learning, which prevents generalizing the models to fit observed data on training data, as well as unseen data on testing sets [70]. Frequent problems for overfitting are the lack of sufficient training data or uneven class balance within the datasets [71]; thus, overfitting is particularly common for models using small datasets [48]. Therefore, similar to previous studies [72,73], we conducted different measures to prevent overfitting, including data augmentation [48] and a 70%-30% training/testing ratio [50]. In addition, to increase the robustness of the models, the images were captured using different wreaths for training and testing at different times and days and using different filters and light conditions.

Although different deep learning models were utilized to detect defects in different sectors [74,75], a few challenges persist due to the dependency of the performance on datasets [76]. Despite utilizing a small dataset, our findings showed an adequate performance when using one-stage models (YOLOv4-tiny, YOLOv5, and YOLOv8). For cases where large datasets are unattainable, pre-training [77] and transfer learning [78] are practical options when using small datasets. Additionally, with multi-scale training, YOLO detects better on objects of different sizes and with an easy trade-off between the performance and inferences [21]. Moreover, one-stage object detection models can work in real-time [79], thus being an advantage over two-stage models. YOLO is designed for real-time object detection, representing an adequate option for inspection in real-time in the agricultural sector [5,54]. Among six different YOLO versions, YOLOv4-tiny presented the best combination of accuracy and speed, being considered for real-time in a previous study [80]. We used Google Colab, obtaining adequate results and being effective in enabling deep learning architectures on resource-constrained computers. Researchers and small enterprises can effectively train their models with minimum investment by leveraging the available resources, including powerful GPUs and ample memory. The Pay-as-You-Go approach (including Watson Studio, Kaggle Kernels, Microsoft Azure Notebooks, and Codeanywhere) presents an accessible avenue for companies seeking to harness the potential of deep learning architectures without the need for expensive hardware upgrades.

5.2. Human Inspection

Our results indicated an overall human inspection accuracy of 97% (five classes) and 100% (two classes), which is greater than reported values in the manufacturing sector, where the accuracy ranges from 80% to 85% [7], or even more critical industries such as the inspection of aircraft engine blades, where the average appraiser agreement with the ground truth was reported to be 67.69% [81], and 84% when operators were allowed to use their hands and apply their tactile sense [82]. In this regard, high values of accuracy and precision are related to different factors, including the complexity of the inspection process, the training level, and the experience of inspectors. In this study, wreaths are not considered a critical product, the inspection process is relatively simple, and inspectors averaged five years of experience, thus not presenting common problems in the agricultural context, such as object size and occlusion [83]. Particularly, training is critical since a human inspector's accuracy usually lies between 70 and 80% after a training period [84]. Moreover,

depending on the product, the inspection process might include pure visual inspection, another sensorial inspection such as tactile inspection, or a combination. Evidence from various industries suggested that a combination of visual and tactile inspection improved respondent detection [82,85].

For the attribute agreement analysis, we computed Kappa, which is a common statistic used to assess the effectiveness of attribute-based inspections [86], allowing us to assess the reliability of agreement between a fixed number of assessors and being more robust than the percent agreement of the AAA [87]. The overall kappa value of 0.9046 indicated a good level of absolute agreement of the assessments between inspectors with the standard. Compared with other industries, this measure is adequate since the agreement acceptance limits for the aerospace industry indicate unacceptable values lower than 80% [88].

5.3. Inspection Challenges

The comparison of the performance between visual human inspection and inspection using AI tools has shown different results depending on several factors, including the complexity of the inspection, context, data availability, and inspectors' experience, among others. Despite the common finding of DL algorithms outperforming human visual inspection [89,90], this is not always the case. Some studies reported mixed and similar results [83,91] or a lower model performance compared with human inspectors [92]. This variety of results indicates a gap in fully automated inspection and a prevalence of human intervention, including cases where algorithms initiate the inspection and inspectors intervene for dubious items or items below a predefined threshold [93,94]. In all cases, DL models assist inspection activities by reducing human intervention, thus reducing physical and mental fatigue.

Time for processing and inspecting is another critical factor when comparing human visual inspection and AI-assisted inspection. In this study, the average human inspection time was 9.26 s per wreath, while models averaged less than 1 s per wreath; this is in line with previous studies where inspectors and algorithms exhibited similar performance except in detection speed, where algorithms are superior [89,91].

Visual and tactile inspection might enhance the inspection performance, particularly for cases with restricted views or where surface inspection is required. However, visual and sensorial inspections might be complex for agricultural and floricultural applications. Detection in agriculture settings has particular features and, frequently, is more challenging than standard detection benchmarks [95]. Images of the field might comprise several objects with high-scale variance, occlusion of objects, and similarity to the background structures [83].

In our study, wreath images present minimum scale variance. However, for inspections that require detecting objects at different scales and handling small objects effectively, feature extraction techniques, including Feature Pyramid Network (FPN) [96], might be used, which has been successfully utilized in other object detection models, such as Faster R-CNN and RetinaNet. Additionally, the quantity of objects to inspect is critical. In datasets with several objects (dozens or hundreds), algorithms outperformed humans, while this is the opposite in datasets with the prominence of occlusion [83].

5.4. Limitations and Future Work

The limitations of this study are various. The results are based on a specific and small dataset and, thus, may not generalize to inspections of other ornamental products or other floricultural and agricultural contexts. Regarding visual inspection, we utilized a small sample size of inspectors; therefore, subjectivity might affect the result. In addition, we did not control environmental conditions (e.g., temperature and humidity), which might affect the inspection process. This study focused on assessing visual inspection and DL inspection when detecting five classes of wreaths, thus not focusing on modifying the network, with this being a limitation of this study. The dataset was adequately balanced for five classes; however, for the complementary analysis using two classes, an adequate

balance was not possible, due to limitations on the material. Thus, balancing the two classes for further analysis remains for future research. In addition, this research did not explore sociotechnical factors that might have an impact on the results. Therefore, further analysis is necessary, including refinement of the models and modifications of the networks to improve their performance. Moreover, expanding the dataset to include complex backgrounds is also recommended for future research. In addition, more research is required to validate these results in various agricultural settings, including exploring additional datasets and environmental conditions, as well as employing advanced techniques to advance the utilization of deep learning models in agricultural inspection and quality assurance.

6. Conclusions

This study compared human visual inspection with deep learning models for inspecting decorative wreaths. The results indicated that the models presented similar performance to humans in terms of precision and accuracy, highlighting the DL suitability in enhancing quality inspection by leveraging the models' ability to capture subtle details and quality flaws that the human eye might miss. Notably, one-stage models such as YOLOv4-tiny, YOLOv5, and YOLOv8 resulted in a similar or slightly superior performance than inspectors in detecting quality flaws. Also, they outperformed a two-stage model such as ResNet50, providing evidence of adequate performance with small datasets and being suitable for real-time inspection. These findings have practical implications for quality control processes in floriculture and agriculture, aiding in identifying and mitigating material absence/excess and color-related issues.

Shifting the paradigm from predominant human-driven inspections to automated, human-supported inspections requires careful consideration. Implementation strategies should encompass a phased approach involving technology integration, training programs for human operators to transition into supportive roles, and validation of the automated systems' performance against established benchmarks. Furthermore, ethical implications related to potential effects on labor should be thoughtfully addressed.

By assisting humans with digital technologies and automation for inspection purposes, organizations can embrace the full potential of Industry 4.0, making the inspection process more intelligent and reliable.

Author Contributions: Conceptualization, D.C.-R. and D.T.; data curation, G.T.; formal analysis, J.L.-R., G.T. and D.T.; investigation, D.C.-R. and Y.B.-L.; methodology, D.C.-R. and D.T.; project administration, Y.B.-L.; resources, Y.B.-L. and J.L.-R.; supervision, G.T. and D.T.; validation, Y.B.-L. and J.L.-R.; writing—original draft, D.C.-R. and D.T.; writing—review and editing, Y.B.-L., J.L.-R. and G.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: The dataset is available on Kaggle as the Wreath Dataset [51]. Additional data presented in this study are available on request from the corresponding author. The data are not publicly available, due to confidential limitations of the company where the study was conducted.

Acknowledgments: We acknowledge the Consejo Nacional de Humanidades, Ciencias y Tecnologías, and the Universidad Autónoma de Baja California, for the support to conduct this study.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Table A1. Summary of studies using different deep learning architectures in agricultural contexts.

Authors	Dataset	Feature	Architecture	Accuracy	Precision
Ismail and Malik (2022) [20]	Apples and bananas	Color	ResNet50 + DenseNet-121 + MobileNetV2	97.2%	NR
			DenseNet-121 + NASNet-A + EfficientNetB0	98.6%	NR
			EfficientNetB0 + B1 + DenseNet-121	98.9%	NR
			EfficientNetB0 + B1 + B2	99.5%	NR
Tan et al. (2022) [56]	Rice	Volume or density	EfficientnetB4	99.47%	99.53%
			Densenet121	99.06%	98.79%
			ResNet50	98.97%	98.74%
			VGG16	94.84%	94.55%
Ponce et al. (2019) [39]	Olive-Fruit	Shape	AlexNet	89.9%	NR
			Inception-ResNetV2	91.81%	NR
			InceptionV1	94.86%	NR
			InceptionV3	95.33%	NR
			ResNet50	94%	NR
ResNet101	95.91%	NR			
Momeny et al. (2022) [18]	Orange infected	Color	ResNet18	NR	100%
			GoogleNet	NR	100%
			ShuffleNet	NR	100%
			ResNet50	NR	100%
			MobileNetv2	NR	100%
			DenseNet201	NR	100%
Li et al. (2021) [29]	Grape test set	Color and quantity	Faster-RCNN(Resnet50)	NR	90.69%
			SSD300	NR	89.82%
			YOLOv4	NR	92.98%
			YOLOv4-tiny	NR	85.56%
			YOLO-Grape	NR	92.21%
Razfar et al. (2022) [97]	Weed in soybean	Texture	MobileNetv2	30.37%	NR
			ResNet50	82%	NR
			Custom 1 (4-layer CNN)	90.33%	NR
			Custom 1 (5-layer CNN)	97.7%	NR
			Custom 1 (8-layer CNN)	97.17%	NR
Cruz et al. (2019) [35]	Grapevine yellows	Color	AlexNet	97.63%	89.62%
			GoogleNet	96.36%	92.8%
			Inceptionv3	98.43%	96.92%
			ResNet50	99.18%	96.69%
			ResNet101	99.33%	97.09%
			Squeeze Net	93.77%	90.82%
			Human	75.69%	73.35%
Veeragandham and Santhi (2022) [37]	Corn weed	Shape	ResNet101	98.33%	98.35%
			ResNet50	99.16%	99.17%
			VGG19	94.5%	94.6%
			VGG16	96.83%	96.93%
			AlexNet	99%	99.01%
Zhang et al. (2021) [98]	Rice	Color	RTR-CPS model	97.76%	NR

Table A1. Cont.

Authors	Dataset	Feature	Architecture	Accuracy	Precision
Zhu et al. (2023) [99]	Apples leaf	Color and shape	SSD	NR	86.2%
			Faster-RCNN	NR	82.1%
			YOLOv4	NR	84.5%
			YOLOv5	NR	87.6%
			Apple-Net	NR	93.1%
Giefer et al. (2020) [100]	Fruits	Shape	3D pose model	82.48%	NR
			Xception	90.69%	NR
			InceptionResNetV2	92.68%	NR
Chen et al. (2022) [28]	Pineapple	Color and shape	Faster RCNN	65.45%	NR
			SSD512	46.4%	NR
			YOLOv4	82.49%	NR
Ge et al. (2022) [34]	Tomato	Color and shape	YOLOv5 s	NR	99.5%
			YOLOv5 m	NR	99.5%
			YOLOv5 1	NR	99.5%
			YOLO-deepsort	NR	99.5%
Gonzalez-Huitron et al. (2021) [49]	Tomato leaves	Color	MobileNetv2	75%	89%
			NasNetMobile	84%	88%
			Xception	100%	100%
			MobileNetv3	98%	98%
			AlexNet	98%	98%
			GoogleNet	99%	99%
			ResNet18	99%	98%
Mohapatra et al. (2021) [101]	Banana, apple and orange	Color	CNN	99%	NR
			R-CNN	97.86%	NR
Anh et al. (2022) [57]	Garlic	Color and shape	modified multi-class model	82.9%	79.4–98.5%
			multi-label	95.2%	79.4–98.5%
Nasiri et al. (2019) [55]	Date	Shape	VGG16	98.49%	96.63%
Piedad et al. (2018) [102]	Banana	Color and shape	ANN	90.03%	NR
			SVC	85.68%	NR
			Random Forest	93.18%	NR
Alipour et al. (2021) [103]	Flowers	Classes	SVM	91.36%	NR
			KNN	100%	NR
			Random Forest	100%	NR
			DNN	31.53%	NR
			VGG16	91.36%	NR
			VGG19	100%	NR
			InceptionV3	100%	NR
Resnet50	100%	NR			

NR denotes not reported; NASNet-A, Neural Architecture Search Network Type A; EfficientNetB0, Efficient Network Type B0; "B1, B2, B4", different variants of Efficient-Net; VGG16, a deep neural network with 16 layers from the VGG (Visual Geometry Group) architecture; Faster-RCNN, a variant of the R-CNN for fast object detection; SSD300 and SSD512, Single-Shot MultiBox Detectors with different input sizes; YOLOv4 and YOLOv5, You Only Look Once versions 4 and 5; YOLO-Grape, a specific variant of YOLO; Custom 1 (4-layer CNN), Custom 1 (5-layer CNN), and Custom 1 (8-layer CNN) are custom models with different numbers of convolutional layers; CNN, Convolutional Neural Network; R-CNN, Region-based Convolutional Neural Network; ANN, Artificial Neural Network; SVC, Support Vector Classifier; Random Forest, ensemble learning algorithm; SVM, Support Vector Machine; KNN, K-Nearest Neighbors; DNN, Deep Neural Network; and Multi-label, model for multi-label classification.

References

1. Janick, J.; Herklots, G.; Perrott, R.; Synge, P. "Horticulture", Encyclopedia Britannica. Available online: <https://www.britannica.com/science/horticulture> (accessed on 9 July 2023).
2. Sagarpa. Las Flores Están en el Campo, en las Miradas, en las Palabras. ... 2023. Available online: <https://www.gob.mx/agricultura/articulos/las-flores-estan-en-el-campo-en-las-miradas-en-las-palabras#:~:text=La%20floricultura%20mexicana%20es%20una,producci%C3%B3n%20la%20realizan%20manos%20femeninas> (accessed on 9 July 2023).

3. Sagarpa. Floricultura, Cultivando Belleza y Ganancias. 2023. Available online: <https://www.gob.mx/agricultura/es/articulos/floricultura-cultivando-belleza-y-ganancias> (accessed on 9 July 2023).
4. Srivastava, D.K.; Jha, D.N. Hibiscus Flower Health Detection to Produce Oil Using Convolution Neural Network. In Proceedings of the 2022 International Conference on Advancements in Smart, Secure and Intelligent Computing (ASSIC), Bhubaneswar, India, 19–20 November 2022; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2022; pp. 1–5.
5. Naik, P.M.; Rudra, B. Flower Phenotype Recognition and Analysis using YoloV5 Models. In Proceedings of the 13th International Conference on Advances in Computing, Control, and Telecommunication Technologies, ACT 2022, Hyderabad, India, 27–28 June 2022; pp. 838–848.
6. Yağ, İ.; Altan, A. Artificial Intelligence-Based Robust Hybrid Algorithm Design and Implementation for Real-Time Detection of Plant Diseases in Agricultural Environments. *Biology* **2022**, *11*, 1732. [[CrossRef](#)]
7. See, J.E. Visual Inspection: A Review of the Literature. 2012. Available online: <https://api.semanticscholar.org/CorpusID:261584909> (accessed on 15 June 2023).
8. Drury, C.G.; Fox, J.G. The imperfect inspector. In *Human Reliability in Quality Control*; Taylor and Francis: New York, NY, USA, 1975; pp. 11–19.
9. Graybeal, B.A.; Phares, B.M.; Rolander, D.D.; Moore, M.; Washer, G. Visual Inspection of Highway Bridges. *J. Nondestruct. Eval.* **2002**, *21*, 67–83. [[CrossRef](#)]
10. Stallard, M.M.; MacKenzie, C.A.; Peters, F.E. A probabilistic model to estimate visual inspection error for metalcastings given different training and judgment types, environmental and human factors, and percent of defects. *J. Manuf. Syst.* **2018**, *48*, 97–106. [[CrossRef](#)]
11. Sundaram, S.; Zeid, A. Artificial Intelligence-Based Smart Quality Inspection for Manufacturing. *Micromachines* **2023**, *14*, 570. [[CrossRef](#)]
12. Kuttyrev, A.; Kiktev, N.; Jewiarz, M.; Khort, D.; Smirnov, I.; Zubina, V.; Hutsol, T.; Tomasik, M.; Biliuk, M. Robotic Platform for Horticulture: Assessment Methodology and Increasing the Level of Autonomy. *Sensors* **2022**, *22*, 8901. [[CrossRef](#)]
13. Ariesen-Verschuur, N.; Verdouw, C.; Tekinerdogan, B. Digital Twins in greenhouse horticulture: A review. *Comput. Electron. Agric.* **2022**, *199*, 107183. [[CrossRef](#)]
14. Singh, R.; Singh, R.; Gehlot, A.; Akram, S.V.; Priyadarshi, N.; Twala, B. Horticulture 4.0: Adoption of Industry 4.0 Technologies in Horticulture for Meeting Sustainable Farming. *Appl. Sci.* **2022**, *12*, 12557. [[CrossRef](#)]
15. Das, R.; Bhatt, S.S.; Kathuria, S.; Singh, R.; Chhabra, G.; Malik, P.K. Artificial Intelligence and Internet of Things Based Technological Advancement in Domain of Horticulture 4.0. In Proceedings of the 2023 IEEE Devices for Integrated Circuit (DevIC), Kalyani, India, 7–8 April 2023; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2023; pp. 207–211.
16. Thirumagal, P.; Abdulwahid, A.H.; HadiAbdulwahid, A.; Kholiya, D.; Rajan, R.; Gupta, M. IoT and Machine Learning Based Affordable Smart Farming. In Proceedings of the 2023 Eighth International Conference on Science Technology Engineering and Mathematics (ICONSTEM), Chennai, India, 7–11 August 2023; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2023; pp. 1–6.
17. Fuentes, A.; Yoon, S.; Kim, C.S.; Park, S.D. A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. *Sensors* **2017**, *17*, 2022. [[CrossRef](#)]
18. Momeny, M.; Jahanbakhshi, A.; Neshat, A.A.; Hadipour-Rokni, R.; Zhang, Y.-D.; Ampatzidis, Y. Detection of citrus black spot disease and ripeness level in orange fruit using learning-to-augment incorporated deep networks. *Ecol. Inform.* **2022**, *71*, 101829. [[CrossRef](#)]
19. Li, Y.; Xue, J.; Wang, K.; Zhang, M.; Li, Z. Surface Defect Detection of Fresh-Cut Cauliflowers Based on Convolutional Neural Network with Transfer Learning. *Foods* **2022**, *11*, 2915. [[CrossRef](#)]
20. Ismail, N.; Malik, O.A. Real-time visual inspection system for grading fruits using computer vision and deep learning techniques. *Inf. Process. Agric.* **2022**, *9*, 24–37. [[CrossRef](#)]
21. Diwan, T.; Anirudh, G.; Tembhurne, J.V. Object detection using YOLO: Challenges, architectural successors, datasets and applications. *Multimed. Tools Appl.* **2023**, *82*, 9243–9275. [[CrossRef](#)] [[PubMed](#)]
22. Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1440–1448. [[CrossRef](#)]
23. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [[CrossRef](#)]
24. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-CNN. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2017; pp. 2961–2969. [[CrossRef](#)]
25. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; IEEE Computer Society: Washington, DC, USA, 2016; pp. 779–788. [[CrossRef](#)]

26. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single shot multibox detector. In *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Springer International Publishing: Cham, Switzerland, 2016; Volume 9905 LNCS, pp. 21–37. [[CrossRef](#)]
27. Teng, S.; Liu, Z.; Chen, G.; Cheng, L. Concrete Crack Detection Based on Well-Known Feature Extractor Model and the YOLO_v2 Network. *Appl. Sci.* **2021**, *11*, 813. [[CrossRef](#)]
28. Chen, S.-H.; Lai, Y.-W.; Kuo, C.-L.; Lo, C.-Y.; Lin, Y.-S.; Lin, Y.-R.; Kang, C.-H.; Tsai, C.-C. A surface defect detection system for golden diamond pineapple based on CycleGAN and YOLOv4. *J. King Saud Univ. Comput. Inf. Sci.* **2022**, *34*, 8041–8053. [[CrossRef](#)]
29. Li, H.; Li, C.; Li, G.; Chen, L. A real-time table grape detection method based on improved YOLOv4-tiny network in complex background. *Biosyst. Eng.* **2021**, *212*, 347–359. [[CrossRef](#)]
30. Zhang, T.; Wu, F.; Wang, M.; Chen, Z.; Li, L.; Zou, X. Grape-Bunch Identification and Location of Picking Points on Occluded Fruit Axis Based on YOLOv5-GAP. *SSRN Electron. J.* **2022**, *9*, 498. [[CrossRef](#)]
31. Wang, J.; Gao, Z.; Zhang, Y.; Zhou, J.; Wu, J.; Li, P. Real-time detection and location of potted flowers based on a ZED camera and a YOLO V4-tiny deep learning algorithm. *Horticulturae* **2022**, *8*, 21. [[CrossRef](#)]
32. Cardellicchio, A.; Solimani, F.; Dimauro, G.; Petrozza, A.; Summerer, S.; Cellini, F.; Renò, V. Detection of tomato plant phenotyping traits using YOLOv5-based single stage detectors. *Comput. Electron. Agric.* **2023**, *207*, 107757. [[CrossRef](#)]
33. Chen, W.; Ju, C.; Li, Y.; Hu, S.; Qiao, X. Sugarcane stem node recognition in field by deep learning combining data expansion. *Appl. Sci.* **2021**, *11*, 8663. [[CrossRef](#)]
34. Ge, Y.; Lin, S.; Zhang, Y.; Li, Z.; Cheng, H.; Dong, J.; Shao, S.; Zhang, J.; Qi, X.; Wu, Z. Tracking and Counting of Tomato at Different Growth Period Using an Improving YOLO-Deepsort Network for Inspection Robot. *Machines* **2022**, *10*, 489. [[CrossRef](#)]
35. Cruz, A.; Ampatzidis, Y.; Pierro, R.; Materazzi, A.; Panattoni, A.; De Bellis, L.; Luvisi, A. Detection of grapevine yellows symptoms in *Vitis vinifera* L. with artificial intelligence. *Comput. Electron. Agric.* **2019**, *157*, 63–76. [[CrossRef](#)]
36. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; IEEE Computer Society: Washington, DC, USA, 2016; pp. 770–778. [[CrossRef](#)]
37. Veeragandham, S.; Santhi, H. Effectiveness of convolutional layers in pre-trained models for classifying common weeds in groundnut and corn crops. *Comput. Electr. Eng.* **2022**, *103*, 108315. [[CrossRef](#)]
38. Desai, S.; Gode, C.; Fulzele, P. Flower Image Classification Using Convolutional Neural Network. In Proceedings of the 2022 First International Conference on Electrical, Electronics, Information and Communication Technologies (ICEEICT), Tamil Nadu, India, 16–18 February 2022; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2022; pp. 1–4.
39. Ponce, J.M.; Aquino, A.; Andujar, J.M. Olive-fruit variety classification by means of image processing and convolutional neural networks. *IEEE Access* **2019**, *7*, 147629–147641. [[CrossRef](#)]
40. Bochkovskiy, A.; Wang, C.; Liao, H. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:2004.10934.
41. Redmon, J. Darknet: Open Source Neural Networks in C. Volume 2016, Art. no. YOLOv4-Tiny. 2016. Available online: <https://pjreddie.com/darknet/> (accessed on 3 October 2023).
42. Liu, Y.; Liu, J.; Cheng, W.; Chen, Z.; Zhou, J.; Cheng, H.; Lv, C. A High-Precision Plant Disease Detection Method Based on a Dynamic Pruning Gate Friendly to Low-Computing Platforms. *Plants* **2023**, *12*, 2073. [[CrossRef](#)]
43. Jocher, G. YOLOv5 by Ultralytics. 2020. Available online: <https://github.com/ultralytics/yolov5> (accessed on 3 October 2023).
44. Wu, B.; Pang, C.; Zeng, X.; Hu, X. ME-YOLO: Improved YOLOv5 for Detecting Medical Personal Protective Equipment. *Appl. Sci.* **2022**, *12*, 11978. [[CrossRef](#)]
45. Reis, D.; Kupec, J.; Hong, J.; Daoudi, A. Real-Time Flying Object Detection with YOLOv8. May 2023. Available online: <http://arxiv.org/abs/2305.09972> (accessed on 20 September 2023).
46. Jocher, G.; Chaurasia, A.; Qiu, J. YOLOv8 by Ultralytics. 2023. Available online: <https://github.com/ultralytics/ultralytics> (accessed on 10 June 2023).
47. Wang, X.; Gao, H.; Jia, Z.; Li, Z. BL-YOLOv8: An Improved Road Defect Detection Model Based on YOLOv8. *Sensors* **2023**, *23*, 8361. [[CrossRef](#)]
48. Perez, L.; Wang, J. The Effectiveness of Data Augmentation in Image Classification using Deep Learning. *arXiv* **2017**, arXiv:1712.04621.
49. Gonzalez-Huitron, V.; León-Borges, J.A.; Rodríguez-Mata, A.; Amabilis-Sosa, L.E.; Ramírez-Pereda, B.; Rodríguez, H. Disease detection in tomato leaves via CNN with lightweight architectures implemented in Raspberry Pi 4. *Comput. Electron. Agric.* **2021**, *181*, 105951. [[CrossRef](#)]
50. Glučina, M.; Šegota, S.B.; Anđelić, N.; Car, Z. Automated Detection and Classification of Returnable Packaging Based on YOLOV4 Algorithm. *Appl. Sci.* **2022**, *12*, 11131. [[CrossRef](#)]
51. Caballero-Ramirez, D.; Baez-Lopez, Y.; Limon-Romero, J.; Tortorella, G.; Tlapa, D. Wreath Dataset. UABC, Ensenada, 2023. Available online: <https://www.kaggle.com/datasets/diegocaballeror/wreath-dataset/> (accessed on 31 October 2023).
52. Dawod, R.G.; Dobre, C. ResNet interpretation methods applied to the classification of foliar diseases in sunflower. *J. Agric. Food Res.* **2022**, *9*, 100323. [[CrossRef](#)]

53. Rahmatullah, P.; Abidin, T.F.; Misbullah, A. Nazarruddin Effectiveness of Data Augmentation in Multi-class Face Recognition. In Proceedings of the 2021 5th International Conference on Informatics and Computational Sciences (ICICoS), Semarang, Indonesia, 24–25 November 2021; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2021; pp. 64–68.
54. Ajayi, O.G.; Ashi, J. Effect of varying training epochs of a Faster Region-Based Convolutional Neural Network on the Accuracy of an Automatic Weed Classification Scheme. *Smart Agric. Technol.* **2023**, *3*, 100128. [[CrossRef](#)]
55. Nasiri, A.; Taheri-Garavand, A.; Zhang, Y.-D. Image-based deep learning automated sorting of date fruit. *Postharvest Biol. Technol.* **2019**, *153*, 133–141. [[CrossRef](#)]
56. Tan, S.; Liu, J.; Lu, H.; Lan, M.; Yu, J.; Liao, G.; Wang, Y.; Li, Z.; Qi, L.; Ma, X. Machine Learning Approaches for Rice Seedling Growth Stages Detection. *Front. Plant Sci.* **2022**, *13*, 914771. [[CrossRef](#)]
57. Anh, P.T.Q.; Thuyet, D.Q.; Kobayashi, Y. Image classification of root-trimmed garlic using multi-label and multi-class classification with deep convolutional neural network. *Postharvest Biol. Technol.* **2022**, *190*, 111956. [[CrossRef](#)]
58. Shahinfar, S.; Meek, P.; Falzon, G. “How many images do I need?” Understanding how sample size per class affects deep learning model performance metrics for balanced designs in autonomous wildlife monitoring. *Ecol. Inform.* **2020**, *57*, 101085. [[CrossRef](#)]
59. Naranjo, M.; Fuentes, D.; Muelas, E.; Díez, E.; Ciruelo, L.; Alonso, C.; Abenza, E.; Gómez-Espinosa, R.; Luengo, I. Object Detection-Based System for Traffic Signs on Drone-Captured Images. *Drones* **2023**, *7*, 112. [[CrossRef](#)]
60. Francies, M.L.; Ata, M.M.; Mohamed, M.A. A robust multiclass 3D object recognition based on modern YOLO deep learning algorithms. *Concurr. Comput.* **2022**, *34*, e6517. [[CrossRef](#)]
61. Cohen, J. A Coefficient of Agreement for Nominal Scales. *Educ. Psychol. Meas.* **1960**, *20*, 37–46. [[CrossRef](#)]
62. Fleiss, J.L. *Statistical Methods for Rates and Proportions*; Wiley: Hoboken, NJ, USA, 1973.
63. Landis, J.R.; Koch, G.G. The Measurement of Observer Agreement for Categorical Data. *Biometrics* **1977**, *33*, 159–174. [[CrossRef](#)] [[PubMed](#)]
64. Chen, C.; Liang, Y.; Zhou, L.; Tang, X.; Dai, M. An automatic inspection system for pest detection in granaries using YOLOv4. *Comput. Electron. Agric.* **2022**, *201*, 107302. [[CrossRef](#)]
65. Liu, X.; Hu, J.; Wang, H.; Zhang, Z.; Lu, X.; Sheng, C.; Song, S.; Nie, J. Gaussian-IoU loss: Better learning for bounding box regression on PCB component detection. *Expert Syst. Appl.* **2022**, *190*, 116178. [[CrossRef](#)]
66. Pun, T.B.; Neupane, A.; Koech, R.; Walsh, K. Detection and counting of root-knot nematodes using YOLO models with mosaic augmentation. *Biosens. Bioelectron. X* **2023**, *15*, 100407. [[CrossRef](#)]
67. Zhang, D.-Y.; Zhang, W.; Cheng, T.; Zhou, X.-G.; Yan, Z.; Wu, Y.; Zhang, G.; Yang, X. Detection of wheat scab fungus spores utilizing the Yolov5-ECA-ASFF network structure. *Comput. Electron. Agric.* **2023**, *210*, 107953. [[CrossRef](#)]
68. Straker, A.; Puliti, S.; Breidenbach, J.; Klein, C.; Pearse, G.; Astrup, R.; Magdon, P. Instance segmentation of individual tree crowns with YOLOv5: A comparison of approaches using the ForInstance benchmark LiDAR dataset. *ISPRS Open J. Photogramm. Remote Sens.* **2023**, *9*, 100045. [[CrossRef](#)]
69. Moreno, H.; Gómez, A.; Altares-López, S.; Ribeiro, A.; Andújar, D. Analysis of Stable Diffusion-derived fake weeds performance for training Convolutional Neural Networks. *Comput. Electron. Agric.* **2023**, *214*, 108324. [[CrossRef](#)]
70. Ying, X. An Overview of Overfitting and its Solutions. *J. Phys. Conf. Ser.* **2019**, *1168*, 022022. [[CrossRef](#)]
71. Mikołajczyk, A.; Grochowski, M. Data augmentation for improving deep learning in image classification problem. In Proceedings of the International Interdisciplinary PhD Workshop (IIPhDW), Swinoujście, Poland, 9–12 May 2018; pp. 117–122. [[CrossRef](#)]
72. Momeny, M.; Jahanbakhshi, A.; Jafarnejad, K.; Zhang, Y.-D. Accurate classification of cherry fruit using deep CNN based on hybrid pooling approach. *Postharvest Biol. Technol.* **2020**, *166*, 111204. [[CrossRef](#)]
73. Jahanbakhshi, A.; Momeny, M.; Mahmoudi, M.; Zhang, Y.-D. Classification of sour lemons based on apparent defects using stochastic pooling mechanism in deep convolutional neural networks. *Sci. Hortic.* **2020**, *263*, 109133. [[CrossRef](#)]
74. Liu, Y.; Xu, K.; Xu, J. Periodic surface defect detection in steel plates based on deep learning. *Appl. Sci.* **2019**, *9*, 3127. [[CrossRef](#)]
75. Wei, X.; Yang, Z.; Liu, Y.; Wei, D.; Jia, L.; Li, Y. Railway track fastener defect detection based on image processing and deep learning techniques: A comparative study. *Eng. Appl. Artif. Intell.* **2019**, *80*, 66–81. [[CrossRef](#)]
76. Cheng, M.; Xu, C.; Wang, J.; Zhang, W.; Zhou, Y.; Zhang, J. MicroCrack-Net: A Deep Neural Network With Outline Profile-Guided Feature Augmentation and Attention-Based Multiscale Fusion for MicroCrack Detection of Tantalum Capacitors. *IEEE Trans. Aerosp. Electron. Syst.* **2022**, *58*, 5141–5152. [[CrossRef](#)]
77. Feng, S.; Zhou, H.; Dong, H. Using deep neural network with small dataset to predict material defects. *Mater. Des.* **2019**, *162*, 300–310. [[CrossRef](#)]
78. Wang, D.; He, D. Channel pruned YOLO V5s-based deep learning approach for rapid and accurate apple fruitlet detection before fruit thinning. *Biosyst. Eng.* **2021**, *210*, 271–281. [[CrossRef](#)]
79. Yang, J.; Li, S.; Wang, Z.; Dong, H.; Wang, J.; Tang, S. Using deep learning to detect defects in manufacturing: A comprehensive survey and current challenges. *Materials* **2020**, *13*, 5755. [[CrossRef](#)]
80. Sozzi, M.; Cantalamessa, S.; Cogato, A.; Kayad, A.; Marinello, F. Automatic Bunch Detection in White Grape Varieties Using YOLOv3, YOLOv4, and YOLOv5 Deep Learning Algorithms. *Agronomy* **2022**, *12*, 319. [[CrossRef](#)]
81. Aust, J.; Pons, D. Assessment of Aircraft Engine Blade Inspection Performance Using Attribute Agreement Analysis. *Safety* **2022**, *8*, 23. [[CrossRef](#)]
82. Aust, J.; Mitrovic, A.; Pons, D. Comparison of visual and visual–tactile inspection of aircraft engine blades. *Aerospace* **2021**, *8*, 313. [[CrossRef](#)]

83. Wosner, O.; Farjon, G.; Bar-Hillel, A. Object detection in agricultural contexts: A multiple resolution benchmark and comparison to human. *Comput. Electron. Agric.* **2021**, *189*, 106404. [[CrossRef](#)]
84. Syberfeldt, A.; Vuoluterä, F. Image processing based on deep neural networks for detecting quality problems in paper bag production. *Procedia CIRP* **2020**, *93*, 1224–1229. [[CrossRef](#)]
85. Daeschel, D.; Rana, Y.S.; Chen, L.; Cai, S.; Dando, R.; Snyder, A.B. Visual inspection of surface sanitation: Defining the conditions that enhance the human threshold for detection of food residues. *Food Control* **2023**, *149*, 109691. [[CrossRef](#)]
86. Sanchez-Marquez, R.; Gerhorst, F.; Schindler, D. Effectiveness of quality inspections of attributive characteristics—A novel and practical method for estimating the “intrinsic” value of kappa based on alpha and beta statistics. *Comput. Ind. Eng.* **2023**, *176*, 109006. [[CrossRef](#)]
87. AESQ. *RM13003 Measurement Systems Analysis*; AESQ: Warrendale, PA, USA, 2021.
88. SAE International. *AESQ Quality Management System Requirements for Aero Engine Design and Production Organizations*; SAE International: Warrendale, PA, USA, 2021.
89. Chang, F.; Liu, M.; Dong, M.; Duan, Y. A mobile vision inspection system for tiny defect detection on smooth car-body surfaces based on deep ensemble learning. *Meas. Sci. Technol.* **2019**, *30*, 125905. [[CrossRef](#)]
90. Rachman, A.; Ratnayake, R.C. Machine learning approach for risk-based inspection screening assessment. *Reliab. Eng. Syst. Saf.* **2019**, *185*, 518–532. [[CrossRef](#)]
91. Aust, J.; Pons, D. Comparative Analysis of Human Operators and Advanced Technologies in the Visual Inspection of Aero Engine Blades. *Appl. Sci.* **2022**, *12*, 2250. [[CrossRef](#)]
92. Kim, T.-Y.; Park, D.; Moon, H.; Hwang, S.-S. A Deep Learning Technique for Optical Inspection of Color Contact Lenses. *Appl. Sci.* **2023**, *13*, 5966. [[CrossRef](#)]
93. Chan, K.Y.; Yiu, K.F.C.; Lam, H.-K.; Wong, B.W. Ball bonding inspections using a conjoint framework with machine learning and human judgement. *Appl. Soft Comput.* **2021**, *102*, 107115. [[CrossRef](#)]
94. Rio-Torto, I.; Campaniço, A.T.; Pinho, P.; Filipe, V.; Teixeira, L.F. Hybrid Quality Inspection for the Automotive Industry: Replacing the Paper-Based Conformity List through Semi-Supervised Object Detection and Simulated Data. *Appl. Sci.* **2022**, *12*, 5687. [[CrossRef](#)]
95. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft COCO: Common Objects in Context. In Proceedings of the Computer Vision—ECCV 2014: 13th European Conference, Zurich, Switzerland, 6–12 September 2014; pp. 740–755.
96. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2017. [[CrossRef](#)]
97. Razfar, N.; True, J.; Bassiouny, R.; Venkatesh, V.; Kashef, R. Weed detection in soybean crops using custom lightweight deep learning models. *J. Agric. Food Res.* **2022**, *8*, 100308. [[CrossRef](#)]
98. Zhang, Y.; Xiao, D.; Liu, Y. Automatic Identification Algorithm of the Rice Tiller Period Based on PCA and SVM. *IEEE Access* **2021**, *9*, 86843–86854. [[CrossRef](#)]
99. Zhu, R.; Zou, H.; Li, Z.; Ni, R. Apple-Net: A Model Based on Improved YOLOv5 to Detect the Apple Leaf Diseases. *Plants* **2023**, *12*, 169. [[CrossRef](#)] [[PubMed](#)]
100. Giefer, L.A.; Arango, J.D.; Faghihabdollahi, M.; Freitag, M. Orientation detection of fruits by means of convolutional neural networks and laser line projection for the automation of fruit packing systems. *Procedia CIRP* **2020**, *88*, 533–538. [[CrossRef](#)]
101. Mohapatra, D.; Choudhury, B.; Sabat, B. An Automated System for Fruit Gradation and Aberration Localisation using Deep Learning. In Proceedings of the 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 19–20 March 2021; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2021; pp. 6–10.
102. Piedad, E.J.; Larada, J.I.; Pojas, G.J.; Ferrer, L.V.V. Postharvest classification of banana (*Musa acuminata*) using tier-based machine learning. *Postharvest Biol. Technol.* **2018**, *145*, 93–100. [[CrossRef](#)]
103. Alipour, N.; Tarkhaneh, O.; Awrangjeb, M.; Tian, H. Flower Image Classification Using Deep Convolutional Neural Network. In Proceedings of the 2021 7th International Conference on Web Research (ICWR), Tehran, Iran, 19–20 May 2021; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2021; pp. 1–4.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.