applied sciences

MDPI

Article

Enhanced Defect Management in Strawberry Processing Using Machine Vision: A Cost-Effective Edge Device Solution for Real-Time Detection and Quality Improvement

Rodoljub Jovanović, Aleksandar Djordjevic *, Miladin Stefanovic, Milan Eric and Nemanja Pajić

Faculty of Engineering, University of Kragujevac, 34000 Kragujevac, Serbia; rodoljub.jovanovic.ts@gmail.com (R.J.); miladin@kg.ac.rs (M.S.); ericm@kg.ac.rs (M.E.); pajicnemanja2@gmail.com (N.P.)
* Correspondence: adjordjevic@kg.ac.rs

Abstract: Managing defects in agricultural fruit processing is crucial for maintaining quality and sustainability in the fruit market. This study explores the use of edge devices, web applications, and machine vision algorithms to improve defect reporting and classification in the strawberry processing sector. A software solution was developed to utilize edge devices for detecting and managing strawberry defects by integrating web applications and machine vision algorithms. The study shows that integrating built-in cameras and machine vision algorithms leads to improved fruit quality and processing efficiency, allowing for better identification and response to defects. Tested in small organic and conventional strawberry processing enterprises, this solution digitizes defect-reporting systems, enhances defect management practices, and offers a user-friendly, cost-effective technology suitable for wider industry adoption. Ultimately, implementing this software enhances the organization and efficiency of fruit production, resulting in better quality control practices and a more sustainable fruit processing industry.

Keywords: smart agriculture; web applications; machine vision; fruit production; defect management; strawberry classification

check for updates

Citation: Jovanović, R.; Djordjevic, A.; Stefanovic, M.; Eric, M.; Pajić, N. Enhanced Defect Management in Strawberry Processing Using Machine Vision: A Cost-Effective Edge Device Solution for Real-Time Detection and Quality Improvement. *Appl. Sci.* **2024**, *14*, 7771. https://doi.org/10.3390/app14177771

Academic Editor: Andrea Prati

Received: 21 June 2024
Revised: 27 August 2024
Accepted: 29 August 2024
Published: 3 September 2024

1. Introduction

Modern agricultural companies prioritize producing higher-quality products and meeting changing consumer needs to stay competitive. Achieving these goals requires constant monitoring, control, and improvement of processing procedures for both organic and conventional products. This demands the implementation of effective quality management practices and modern technology [1]. Despite the integration of quality management initiatives, the dynamic nature of agricultural markets presents significant obstacles that surpass the effectiveness of traditional quality management strategies. These challenges notably include the classification of fruit products and potential defects they may exhibit. Additionally, the traditional manual approach in this field is associated with repetitive, physically demanding work. For example, quality control and food classification may take place in environments with low temperatures and other challenging conditions.

As a cornerstone of modern agriculture, the strawberry industry greatly emphasizes product quality and profitability [2,3]. A significant criterion in assessing strawberry grade is the presence of defects, such as mold [4,5], which stands as one of the more prevalent issues in the strawberry domain. Strawberry defect detection relies heavily on manual labor, characterized by extended work hours, elevated costs, and suboptimal efficiency. Particularly concerning strawberry mold, traditional detection methods are time-consuming and ineffective.

In recent years, there has been an increase in the use of machine vision technologies in agriculture, leading to the development and deployment of various neural network architectures [6,7]. Song et al. [8] created a system using You Only Look Once (YOLO) to detect citrus diseases, such as Citrus Canker and Citrus Greening, allowing for better disease management. Qin et al. [9] introduced Ag-YOLO, an inexpensive object detection system specifically designed for precise spraying in precision agriculture using unmanned aerial vehicles (UAVs). This system addresses the challenges of effective pesticide application in small fields and rugged terrains, such as mountainous areas, by providing a cost-effective and adaptable onboard computer vision solution. Lippi et al. [10] developed a pest detection system for hazelnut orchards, achieving 94.5% average precision using a YOLO-based CNN and deploying the system on an NVIDIA Jetson Xavier for real-time processing. Junos et al. [11] focused on enhancing productivity and reducing operational costs in crop harvesting systems. They proposed an optimized YOLO-based object detection model, YOLO-P, specifically tailored for automated crop harvesting systems. The model aims to detect and localize fresh fruit bunches, grabbers, and palm trees in palm oil plantations under various environmental conditions, providing essential visual information for the development of intelligent automated harvesting systems. The model shows a high accuracy of 98.91% in identifying fresh fruit bunches of various maturities, indicating its robustness and effectiveness for real-world applications in palm oil plantations. In a study by Mirhaji et al. [12], the use of YOLO models for fruit detection and load estimation in an orange orchard was investigated, emphasizing the importance of accurate yield estimation for effective market and resource planning in precision agriculture. The findings highlight the effectiveness of YOLO models in providing practical solutions for detecting and estimating orange fruit yields, suggesting a combined imaging approach to improve accuracy for varying canopy densities, with an overall yield estimation error of +9.19%. Wang et al. [13] introduced DSE-YOLO (Detail-Semantics Enhancement YOLO) as a solution for multi-stage strawberry detection, crucial for estimating crop yields and facilitating robotic harvesting in modern agriculture. The model uses novel loss functions, Exponentially Enhanced Binary Cross Entropy (EBCE) and Double-Enhanced Mean Square Error (DEMSE) to address the class imbalance. Experimental results show DSE-YOLO's superiority over existing methods, achieving a mean average precision (mAP) of 86.58% and an F1 score of 81.59%. This model effectively detected all stages of strawberry fruits in natural scenes, providing a solid theoretical foundation for automated harvesting and monitoring systems. Cuong et al. [14] enhanced Tiny YOLO-v4 for detecting pineapple ripeness, achieving a recognition rate of 98.26% and demonstrating applicability in large-scale plantations. The following innovative studies have leveraged YOLO-based algorithms to enhance precision agriculture. Zhang et al. [15] enhanced YOLOv5 to detect unopened cotton bolls, addressing challenges like occlusions and varying growth stages, outperforming YOLOv3, SSD, and Faster R-CNN. Tian et al. [15] introduced MD-YOLO for pest detection in precision agriculture, achieving an email_1 of 86.2%, F1 score of 79.1%, and IoU of 88.1% using DenseNet blocks and adaptive attention modules. Dang et al. [16] developed YOLOWeeds for multi-class weed detection in cotton production and identified YOLOv5n and YOLOv5s as effective for weed detection. Cui et al. [17] investigated the YOLO-FT deep learning algorithm for UAV-based smart agriculture, emphasizing the role of drones in data acquisition and improved pollination through advanced object detection technology. Additionally, Badgujar et al. [18] conducted a comprehensive literature review highlighting YOLO's significance in agricultural object detection, focusing on its performance characteristics and potential for advancing knowledge in the agricultural sector.

As a result, effectively classifying strawberry products and implementing defect detection strategies emerge as potential issues that could be solved by applying machine vision within agricultural enterprises engaged in strawberry processing and cultivation [19]. To address the challenges of strawberry classification and defect mitigation, these enterprises can utilize web technologies and YOLO-integrated technologies for quality control activities. With the increasing availability of Convolutional Neural Network (CNN) variations in computer vision, various network architectures built on them have been used for defect detection, producing significant results [20]. This approach not only reduces overall costs but also enhances the efficiency of agricultural operations, aligning seamlessly with the principles of smart agriculture (SA).

This study aims to showcase the applicability of the SA concept within the strawberry processing industry, marking a stride toward overall agricultural productivity. To accomplish this objective, the authors have devised and implemented edge devices, web applications and machine vision YOLOv8 n, s, m, l, and x and YOLOv9 m, c, t, s and e architectures for small strawberry processing enterprises, offering several benefits.

The motivation for this article stems from the need for efficient and accurate classification of strawberry defects during processing. Given the shortage of human labor and the critical timing required for processing, machine vision and automation are necessary to enhance productivity and quality control. The primary objective of this study is to develop a strawberry defects classification model capable of identifying the location of defective strawberries and distinguishing between different classes of strawberries based on their condition. The main contributions of this work are two-fold:

- It introduces a new dataset aimed at advancing research in object detection systems, specifically for the classification of strawberries. This dataset is used with YOLO version 8 (YOLOv8) n, s, m, l, and x, as well as with YOLO version 9 (YOLOv9) c, m, s, t, and e architectures.
- Models based on the YOLO8 and YOLO9 architectures are proposed to effectively perform object detection for the three identified classes of strawberries.

By addressing these objectives, the study aims to enhance the efficiency and accuracy of defect strawberry classification, ultimately improving the quality and sustainability of fruit processing operations. The novelty of this research is multifaceted and can be discussed from several angles. Firstly, it marks a significant step towards enhancing overall agricultural productivity by applying the SA concept within the strawberry processing industry. Developing and implementing a new hardware and software infrastructure enables quick reporting of defects in alignment with ISO 9001:2015 [21] and IFS FOOD guidelines [22]. This infrastructure supports the efficient management of quality and defects, which is crucial for maintaining high standards in strawberry production. Secondly, introducing a dataset advances the current state of research in object detection systems, specifically targeting the classification of strawberries using YOLOv8 n, s, m, l, and x and YOLOv9 m, c, t, s and e architectures. This approach enhances the accuracy of defect detection and addresses the limitations of traditional manual methods, which are often labor-intensive and inefficient. This comprehensive approach ensures that the models perform well under diverse conditions, reflecting real-world scenarios. Finally, integrating edge devices, web applications and machine vision algorithms improves quality control measures in agriculture. This integration exemplifies the application of modern technologies to address the challenges faced by the agricultural sector, particularly in strawberry processing.

The subsequent sections of the paper explore the integration of SA and Quality 4.0 concepts, focusing on innovative technologies such as the Internet of Things, cloud computing, machine vision, and data analytics, which are essential for maintaining agricultural competitiveness (Section 2.1). The role of machine vision in managing defects in strawberries is discussed, covering detection and classification processes to ensure continuous quality control (Section 2.2). The strawberry storage process is described in detail, highlighting steps from handpicking to packaging to maintain quality and safety (Section 2.3). The design of software components for SA, including a tailored quality control system using computer vision, is detailed, showcasing how various technologies are employed (Section 2.4). Data acquisition and processing are explained using a Zenodo dataset and YOLOv8 and YOLOv9 models for defect detection (Section 2.5). The performance of these models is assessed using standard metrics (Section 2.5.4). The results section illustrates the practical application of the system in real-life scenarios, such as strawberry classification by defects (Section 3). The significance of defect reporting and the development of a modular software solution for defect identification and reporting in SA are highlighted in the discussion (Section 4). Finally, the importance of SA-driven quality control in achieving superior standards and addressing the challenges of defect identification is emphasized in the conclusion (Section 5).

2. Materials and Methods

The research is based on the significant advancements in machine vision and defect detection technologies that have occurred over the past few decades. The authors of this paper conducted a thorough literature review to comprehend the historical development and advancements in machine vision and defect detection technologies. This provided a fundamental understanding of the current state-of-the-art and identified gaps in existing research. Through the review, the authors identified key techniques and algorithms that have been successfully applied in similar contexts, with a particular focus on the YOLO family of algorithms due to their efficiency and accuracy in real-time object detection.

A comprehensive dataset of strawberry images, comprising both defective and non-defective samples, was compiled. Based on the literature review and preliminary experiments, the authors selected the YOLOv8 and YOLOv9 algorithms for their proven effectiveness in object detection tasks. These models were trained on a Lenovo ThinkPad P16 (Lenovo, Quarry Bay, Hong Kong) with an Intel Core iHX processor (Intel, Santa Clara, CA, USA) and NVIDIA RTX GPU (NVIDIA, Santa Clara, CA, USA). This setup ensured efficient and powerful computation capabilities for model training. A Dahua 2MP network camera (Dahua Technology, Hangzhou, China) with full-color technology was used for real-time image acquisition, strategically positioned above a conveyor belt to capture images of strawberries as they moved along the production line. A local server (Lenovo) was configured to process the images in real-time, hosting the trained YOLOv8 models and managing the data received from the camera. A Unitronics PLC (Unitronics, Tel Aviv, Israel) control unit was integrated to manage the operational aspects of the system, facilitating the activation of a jet mouth mechanism to remove defective strawberries based on the model's detections.

Processed data were archived in the cloud, and data visualization tools and statistical modules were employed to analyze defect patterns and system performance. A web interface was developed to provide real-time insights and reports on defect detection, allowing stakeholders to monitor the system's performance and make informed decisions. The integrated system was tested in a real-world environment to validate its accuracy and reliability. Performance metrics such as detection accuracy, processing speed, and false positive rates were measured and analyzed. Based on the validation results, the models and system configurations were continuously refined to enhance overall performance and address any identified issues.

2.1. Smart Agriculture and Quality 4.0 Concepts

The foundation of the SA paradigm rests upon the advancement of innovative technologies such as the Internet of Things, cloud computing, machine vision and data analytics [23,24]. Acknowledging the dynamic nature of the agricultural sector, SA emerges as a requisite for maintaining competitiveness amidst uncertainty [25]. Its overarching objective is to foster agricultural models characterized by flexibility in product offerings and services, facilitated by seamless communication among stakeholders and agricultural facilities throughout the agricultural production and processing cycle [26].

In essence, SA and fruit production and processing should encapsulate several key Quality 4.0 domains, as outlined by Javaid et al. [27]. These components include the following: (1) monitoring of strawberry condition in near real-time feedback from interconnected edge devices accompanied with growing volumes of data; (2) analyses offering descriptive, diagnostic, predictive, and prescriptive capabilities, (3) logistics encompassing mobile applications, platforms, web clients, browsers, and applications for robotics and machinery; and (4) control overseeing autonomous and interconnected processes, including electronic submission of compliance/defect reports and automation of compliance workflows. The objective is to support company endeavors in enhancing strawberry production and processing processes, machinery utilization, and employee operability through the digitalization of strawberry classification and defect detection.

Hence, within the framework of SA, emerging technologies become both cost-effective and readily available to a wider spectrum of agricultural enterprises, presenting an unprecedented opportunity to address persistent quality issues and embrace innovative solutions. Guided by these proposed domains, the primary objective of this paper is to present the development of a mobile solution for fruit classification and defect detection. This solution is anchored in the aggregation of extensive big data, driving advancements in smart agriculture and strawberry production and processing.

2.2. Agricultural Products Defects Management

Many studies have focused on fruit detection but have overlooked the problem of detecting fruits at multiple stages. The DSE (Detail-Semantics Enhancement) YOLO method was proposed by Du et al. [28] to detect multi-stage strawberries. In the DSE-YOLO method, the DSE module was designed to detect small fruits and distinguish different stages of fruit development with higher accuracy. The results show that DSE-YOLO can almost accurately detect each stage of strawberry development in natural conditions, providing an important theoretical basis for automatic picking and monitoring systems [13]. An et al. [29] highlight the challenge of swiftly and accurately identifying strawberry growth conditions and maturity for automated orchard management robots, particularly for operations like automatic pollination, fertilization, and picking. Strawberries, with their short ripening period and heavy overlap and shading, pose significant challenges to traditional detection methods regarding efficiency and effectiveness.

To address this, the authors propose SDNet (Strawberry Detect Net), an algorithm based on the YOLOX model. Their findings address the challenge of accurately monitoring strawberry fruit growth states in complex environments, offering crucial insights for advancing unmanned farming and SA technologies. Luo et al. [30] present an improved method for recognizing small target strawberries using the YOLOv8n model, addressing the challenges of detecting smaller strawberries and reducing misdetections caused by complex backgrounds in strawberry images. Bai et al. [31] emphasize the importance of accurately identifying strawberry seedling flowers and fruits in greenhouse environments for automated flower and fruit thinning, which enhances efficiency and reduces labor costs in cultivation. The authors propose an algorithm that integrates a Swin Transformer prediction head on the high-resolution feature map of YOLO v7 to leverage spatial location information, enhancing detection accuracy for small targets amidst scenes with similar colors and occlusions. These algorithms primarily focus on enhancing the accuracy and efficiency of detecting strawberry fruits and flowers in various environments and developmental stages.

Thus, machine vision YOLO models could play a crucial role in identifying defects in strawberries. These models may be adept at detecting various types of defects present on the surface of strawberries, ranging from physical damage to discoloration. Typical defects include mechanical damage, physiological disorders, internal defects, morphological and pathological disorders, and customer and supplier returns [32]. Once a defect is detected, these models can further classify them based on their severity or type, allowing for a nuanced understanding and differentiation between defective and healthy strawberries. The continuous monitoring capabilities of machine vision models ensure ongoing quality control throughout the strawberry production and processing stages.

By monitoring strawberries, these models can promptly identify any potential defects, enabling intervention to prevent further deterioration and maintain product integrity. This entails the imperative implementation of quality control mechanisms, facilitating the early detection of defects and the reduction in production waste [33]. Analysis of the relevant literature [33] reveals distinct objectives for the solution outlined in this paper across several categories: (1) affordability and suitability for the SA; (2) incorporation of computer vision (YOLOv8 n, s, m, l and x) modules, and aligning with the analytic modules.

2.3. Initial Assumptions and Objectives for the Present Study

The strawberry storage process involves several meticulously designed steps to ensure the fruits remain high quality and safe for consumption. Strawberries are handpicked at optimal ripeness to prevent physical damage and ensure peak quality. Following harvest, the fruits undergo preliminary sorting to remove any visibly damaged, rotten, or unripe specimens [34]. The strawberries are then thoroughly cleaned and washed to eliminate dirt, pesticides, and other impurities, ensuring their hygiene and safety. Detailed sorting follows, where strawberries are categorized by size, color, and quality, either manually or with machines, to select the best fruits for subsequent processing. These sorted strawberries are then rapidly frozen through flash freezing, preserving their freshness, texture, and nutritional value by forming small ice crystals. Finally, frozen strawberries are packed into protective, often vacuum-sealed or airtight, containers labelled with relevant product information, including the freezing and expiration dates. This comprehensive process guarantees that the strawberries maintain their quality and nutritional integrity from harvest to consumption.

Driven by the outlined assumptions, activities in the strawberry storage process and the imperatives of quality, particularly concerning defect reporting, this study aims to leverage technologies from the SA toolkit, such as edge devices and machine vision YOLO algorithms. These technologies could be applied for preliminary and detailed sorting activities, utilizing high-resolution imaging and AI algorithms to categorize strawberries based on predefined quality criteria. The study seeks to optimize the entire strawberry storage and processing workflow through these innovations, ensuring adequate product quality and consumer satisfaction in the SA context.

The overarching goal is to demonstrate how the development of an accessible cloud-based solution utilizing JavaScript ECMAScript 2024, Python 3.11.5, and MySQL 8.0.35 databases holds promise for facilitating real-time defect identification, thereby fostering engagement, continuous improvement, and evidence-based decision-making.

2.4. Design of General Software Components, Their Interrelationships, and Available Technologies for Smart Agriculture Consolidation

The schematic presentation of software components and their interconnections is illustrated in Figure 1 based on Zhang et al. [35]. This figure provides insight into the potential technologies that could be employed to realize various software components and their interplay across distinct tiers within the software architecture. Additionally, Section 2.5 introduces a tailored software infrastructure, aligning with contemporary technological advancements.

Figure 1 illustrates the potential technologies encompassed within the developed solution. The proposed software integrates computer vision techniques utilizing YOLOv8 n, s, m, l, and x and YOLOv9 c, m, s, t, and e algorithms, coupled with edge camera deployment positioned above the narrowing conveyor belt transporting strawberries. For training the YOLOv8 and YOLOv9 algorithms, a Lenovo ThinkPad P16 (Lenovo, Quarry Bay, Hong Kong) with an Intel Core i7-12850HX processor (16 × 1.5–4.8 GHz) (Intel, Santa Clara, CA, USA) and NVIDIA RTX A2000 Laptop GPU (8 GB VRAM) (NVIDIA, Santa Clara, CA, USA) was utilized (step 1). This configuration supports computational tasks required for training machine-learning models, ensuring the readiness of the algorithms for defect detection and strawberry data management.

For this application, a Dahua 2MP network bullet camera (Dahua Technology, Hangzhou, China) with full-color technology was employed (1/2.8″ CMOS with progressive scan). It operates effectively in low light conditions and provides a maximum resolution of 1920 × 1080. The camera features a fixed 2.8 mm lens, H.265+/H.264+ compression, and an IP67 protection rating, making it suitable for indoor and outdoor environments with its

120dB WDR and 30 m LED illumination range. Video analytics capabilities such as Tripwire 9.1.0 and intrusion detection enhance its functionality. These features facilitate real-time image acquisition (step 2), with captured images relayed promptly to a local server for processing (step 3).

The local server, a Lenovo W541 (Lenovo, Quarry Bay, Hong Kong), supports the computational demands of the system with its Intel i7-4810MQ processor (Intel, Santa Clara, CA, USA) running at 2.80 GHz (Turbo up to 3.80 GHz), 32 GB DDR3 RAM, and a 512 GB SSD for fast data access. Graphics processing is handled by an Intel HD 4600 integrated GPU (Intel, Santa Clara, CA, USA) and an NVIDIA Quadro K2100M with 2 GB DDR5 VRAM (NVIDIA, Santa Clara, CA, USA). The trained models are hosted on the local server, facilitating real-time decision-making based on captured data (step 4).

The system can activate the control unit, a PLC Unitronics (step 5) (Unitronics, Tel Aviv, Israel). This control unit subsequently activates the jet mouth to remove defective strawberries (step 6). Processed data are archived in the cloud, supporting storage, visualization tools, and statistical modules (step 7), enabling continuous quality control improvement.

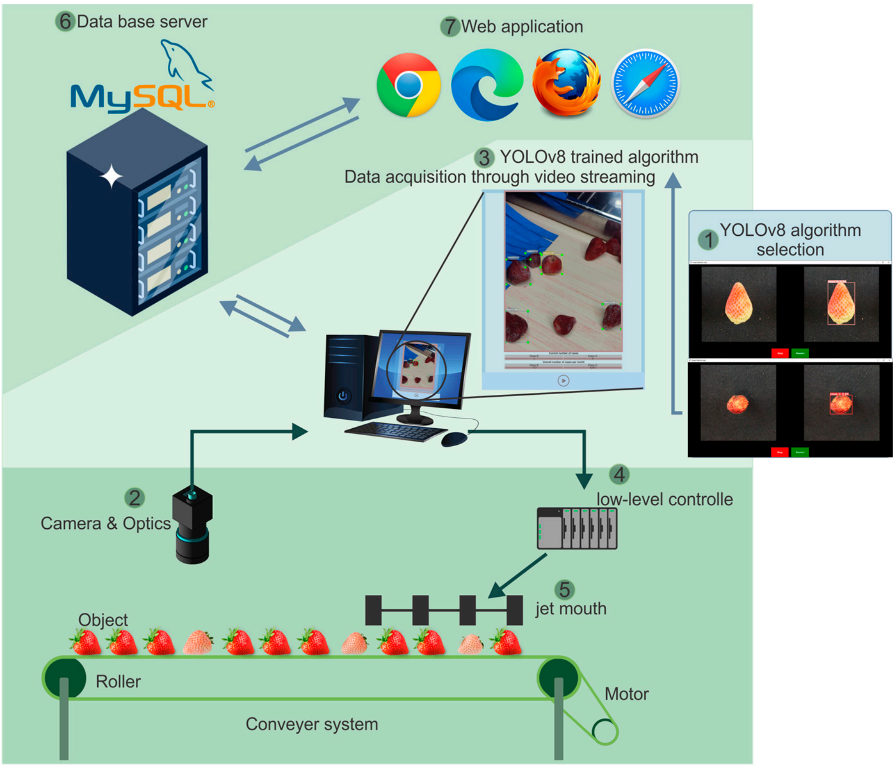

Figure 1. The schematic depiction of a quality control system using computer vision.

2.5. Incorporated Quality Control System Using Computer Vision

This section explains the general solution description depicted in Figure 1, focusing on understanding the specific needs of small and medium-sized agricultural enterprises. Additionally, we intend to underscore the prevailing trends of SA, which encompass centralized data platforms and a heightened emphasis on achieving continuous improvements promptly.

2.5.1. Data Acquisition

The image data utilized to underpin this investigation were sourced from a Zenodo data set [36]. Approximately 230 images for each defect of strawberries were captured for analysis. An exemplary color image is depicted in Figure 2a. This comprehensive set of images was the foundation for identifying potential defects that may arise during the strawberry classification processes. Additionally, it facilitated the enhancement of the selection process, ensuring that only the highest quality strawberries proceed to further processing stages. By meticulously analyzing these images, it is possible to pinpoint specific areas for improvement, optimizing the accuracy and efficiency of the overall strawberry sorting and processing workflow. Furthermore, this image set was used to train YOLOv8 and YOLOv9 models.

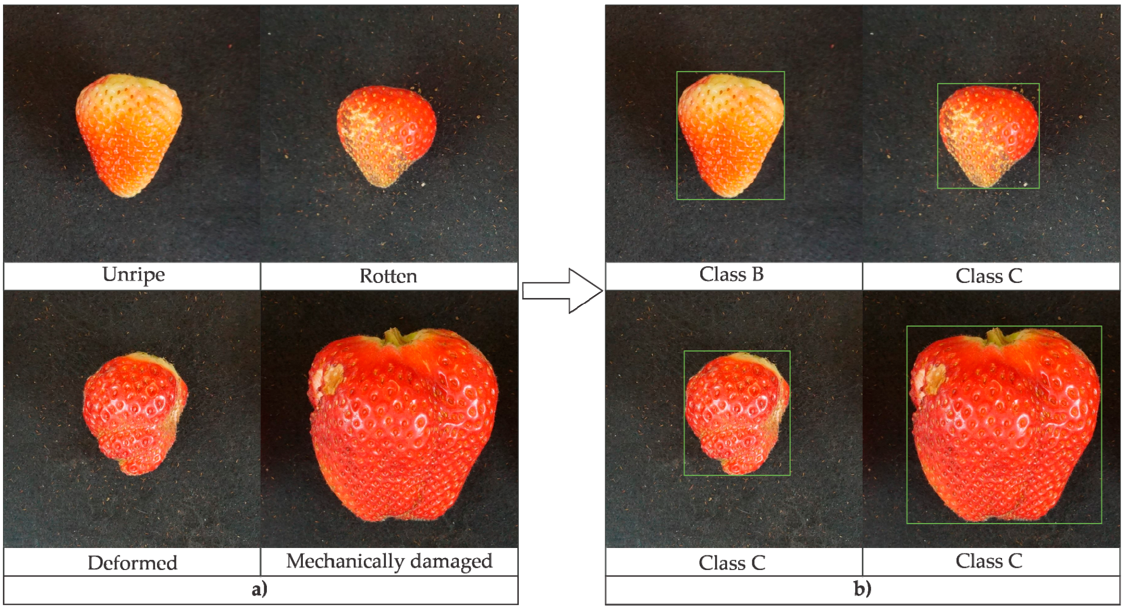

Figure 2. The sample image utilized for training computer vision algorithms. (**a**) unlabeled defected strowberies; (**b**) labeled defected strowberies.

2.5.2. Data Processing

A dataset was built with 467 selected defective strawberry objects to train the strawberry defect detection algorithms. All images were processed to be resized to dimensions of 832 × 832 pixels, which is the resolution required for the application of the YOLOv8 n, s, m, l, and x and YOLOv9 m, c, t, s and e algorithm. This procedure enabled the consistent application of the algorithm for shape recognition and phenotyping of strawberries. Strawberries were divided into two classes: “Unripe Strawberries” (B class) and “Defected Strawberries” (C class). Manual labelling of strawberries in these classes was based on the fruit grading method described by Afzaal et al. [5] and recommendations from strawberry enterprise, as follows: Unripe Strawberries—strawberries with green color covering over 90% of the surface area; Defected Strawberries—strawberries with fuzzy grey mold on the surface, mechanically damaged strawberries, and rotten strawberries that are discolored.

To detect and classify strawberry defects into different classes, all strawberry defects in the images were labelled with bounding boxes and classes, as shown in Figure 2b. After the images were labelled, 467 objects were identified with a given class.

2.5.3. YOLO Object Detection

This investigation employed YOLOv8 n, s, m, l, and x and YOLOv9 m, c, t, s and e models to detect strawberry defects across defined classes. Table 1 presents the training parameters utilized for the YOLOv8 n, s, m, l, and x and YOLOv9 m, c, t, s and e models. Following the generation of bounding boxes around strawberries using the LabelImg 1.8.6 module, these bounding boxes were subsequently fed into the YOLOv8 n, s, m, l, and x and YOLOv9 m, c, t, s and e models to identify the strawberry defect class. The input image sizes for YOLOv8 n, s, m, l, and x and YOLOv9 m, c, t, s and e were configured as 832 × 832, aiming to preserve the key features of objects (strawberries with varying degrees of defects) within the images. Table 1a,b provide a comprehensive summary of all defined parameters.

Table 1. (**a**) Parameters of YOLOv8 n, s, m, l and x training. (**b**) Parameters of YOLOv9 m, c, t, s and e training.

(a)					
Parameter	**YOLOv8 n**	**YOLOv8 s**	**YOLOv8 m**	**YOLOv8 l**	**YOLOv8 x**
Size of the input image	832 × 832	832 × 832	832 × 832	832 × 832	832 × 832
Max training batch	4	4	4	2	1
Number of classes	2	2	2	2	2
Pretrained weights	319/355	349/355	589/595	589/595	589/595
Learning rate	0.01	0.01	0.01	0.01	0.01
Momentum	93.7%	93.7%	93.7%	93.7%	93.7%
Optimizer	AdamW	AdamW	AdamW	AdamW	AdamW
Epochs	20	20	20	20	20
(b)					
Parameter	**YOLOv9 m**	**YOLOv9 c**	**YOLOv9 t**	**YOLOv9 s**	**YOLOv9 e**
Size of the input image	832 × 832	832 × 832	832 × 832	832 × 832	832 × 832
Max training batch	4	4	4	4	4
Number of classes	2	2	2	2	2
Pretrained weights	901/907	931/937	1303/1339	1333/1339	1805/1811
Learning rate	0.01	0.01	0.01	0.01	0.01
Momentum	93.7%	93.7%	93.7%	93.7%	93.7%
Optimizer	AdamW	AdamW	AdamW	AdamW	AdamW
Epochs	20	20	20	20	20

The table compares YOLOv8 n, s, m, l, and x and YOLOv9 m, c, t, s and e models. These models share a standardized input image size of 832 × 832 pixels, ensuring uniformity in image processing. They vary, however, in their maximum training batch sizes: YOLOv8 n, s, and m and YOLOv9 m, c, t, s and e are configured with a batch size of 4 images, while YOLOv8 l operates with 2 images, and YOLOv8 x with just 1. This difference influences training efficiency and GPU memory utilization.

Each model is designed to detect 2 classes, focusing on a binary classification task. They all initialize training using pre-trained weights, with varying degrees of weight utilization: YOLOv8 n starts with 319 out of 355 weights, YOLOv8 s with 349, and YOLOv8 m, l, and x with 589 out of 595 weights. YOLOv9 m starts with 901 out of 907 weights, YOLOv9 c with 931 out of 937, YOLOv9 t, and s with 1303 and 1333 out of 1339 weights and YOLOv9 e with 1805 out of 1811 weights. This initialization strategy accelerates the convergence of training.

All models employ a consistent learning rate of 0.01 and a momentum of 93.7%, optimized with the AdamW optimizer. Training is conducted over 20 epochs for each model, ensuring comprehensive exposure to the dataset for robust learning and accurate object detection. These configurations collectively underscore the models' versatility and capability in handling object detection tasks across various scales and computational capacities. The architecture of the proposed YOLOv8 n, s, m, l and x models consists of a series of convolutional layers, Cross-Stage Partial Network (CSP) with Two Flow Paths (C2f) blocks, a Spatial Pyramid Pooling Fast (SPPF) block, and Concat and Upsample layers. The convolutional layers are designed to extract basic features from input images, while the C2f blocks enable deeper learning of complex features through residual connections. Using the SPPF block further enhances the model's ability to detect objects of various sizes by combining information from different spatial pyramid levels. The Concat layers integrate outputs from different levels of the network, and the Upsample layers increase image resolution while preserving spatial information, allowing for better detection at various scales. Finally, the detect head utilizes information from the previous layers to predict object boundaries and their classes, ensuring accurate and rapid detection results. The architecture of the proposed YOLOv9 m, c, t, s and e models consists of a series of convolutional layers, Efficient Layer Aggregation Network (ELAN) blocks, Replicated Non-CSP ELAN (RepNCSPELAN) blocks, SPPELAN (Spatial Pyramid Pooling with ELAN) blocks, and Concat and Upsample layers. The convolutional layers are designed for the initial extraction of basic features from input images, while the ELAN blocks enable deeper learning of complex features through efficient layer aggregation. RepNCSPELAN blocks further enhance the model's ability to detect objects of various sizes by combining information from different levels of the spatial pyramid. SPPELAN blocks provide additional layering and improve detection efficiency by combining spatial pyramid pooling with ELAN blocks. The Concat layers integrate outputs from different levels of the network, and the Upsample layers increase image resolution while preserving spatial information, allowing for better detection at various scales. Finally, like with the YOLOv8 models, the detect head utilizes information from the previous layers to predict object boundaries and their classes, ensuring accurate and rapid detection results.

The YOLOv8 and YOLOv9 architectures consist of convolutional layers but differ significantly in their block structures. YOLOv8 uses (Cross-Stage Partial with Two Flow Paths) C2f blocks and an SPPF (Spatial Pyramid Pooling Fast) block to enhance feature learning and multi-scale object detection. In contrast, YOLOv9 introduces blocks like (Efficient Layer Aggregation Network) ELAN, (Replicated Non-CSP ELAN) RepNCSPELAN, and (Spatial Pyramid Pooling with ELAN) SPPELAN, which should provide efficient layer aggregation and improved detection capabilities. YOLOv9's architecture should offer reduced parameter count and higher efficiency (7.8 GFLOPs vs. YOLOv8's 8.2 GFLOPs) while maintaining or enhancing detection performance through sophisticated layer integration and spatial pyramid pooling mechanisms.

2.5.4. Performance Assessment

Standard performance metrics for object detection in images or video sequences derived from a confusion matrix are used to evaluate YOLOv8 n, s, m, l, and x and YOLOv9 m, c, t, s and e models. This matrix has a significant role within the evaluation framework, presenting a comprehensive dissection of the model's prognostications. It encapsulates four fundamental components: true negatives (TN), denoting accurately predicted negatives; false positives (FP), signifying erroneously identified positives; false negatives (FN), indicating misclassifications of true positives; and true positives (TP), representing correct identifications of positives. Through its meticulous delineation, this matrix furnishes essential insights into the discriminative capacity and performance of the model under scrutiny.

In the context of the YOLOv8 n, s, m, l, and x and YOLOv9 m, c, t, s and e models, the following metrics were computed:

- Precision (p): In predictive analytics, precision is a litmus test for the model's efficacy in discerning positive instances. It quantifies the fidelity of positive predictions by assessing the ratio of true positive identifications to the aggregate of all positive prognostications. Thus, precision elucidates the model's discerning prowess in accurately pinpointing instances of interest amidst the broader dataset:

$$p = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

- Recall (r): Serves as a critical barometer of the model's proficiency in correctly identifying positive instances within the dataset. This metric is derived by dividing the count of true positive predictions by the sum of true positives and false negatives.

In essence, recall provides a nuanced evaluation of the model's ability to capture all relevant instances of interest, thus illuminating its sensitivity to positive phenomena.

$$r = \frac{True\ Positives}{True\ Positives + False\ Negatives}$$

- Average precision (AP): This frequently used measure assesses object detection accuracy. AP calculates the average precision for different evaluation thresholds and then computes the mean of those precisions.

$$AP = \sum_N (r_{n+1} - r_n)^{max}_{\tilde{r}:\tilde{r}^3\ r_{n+1}p(\tilde{r})}$$

- Mean average precision (mAP): This is the average of AP values across all object classes. It is used to gain an overall insight into the performance of object detectors.

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i$$

- Mean average precision 50–95 (mAP50–95): This is the mean of the AP values calculated at each Intersection over Union (IoU) thresholds from 0.50 to 0.95. Sum all the AP values and divide by the number of IoU thresholds used.

$$mAP50 - 90 = \frac{1}{10}\sum_{i=1}^{10} AP_{0.50+i\times 0.05}$$

- F1 score: This metric serves as a harmonious amalgamation, delicately balancing the evaluation of the model's precision and recall performance. By encapsulating both precision and recall into a single metric, the F1 score furnishes a comprehensive assessment of the model's ability to simultaneously achieve high precision in positive predictions while maintaining a robust recall rate, thus elucidating its overall efficacy in classification tasks.

$$F1\ Score = \frac{2 \times p \times r}{p + r}$$

- Layers refer to the number of distinct layers within a neural network model. In the context of the provided values, each model variant (YOLOv8 and YOLOv9) has different layers, indicating the network's depth and complexity. A higher number of layers typically allows the model to learn more intricate features and representations from the data, potentially improving its performance on complex tasks. However, increasing the number of layers can also lead to more significant computational requirements, longer training times, and challenges related to training stability and convergence.
- Params (M), or the number of parameters in a model expressed in millions, represents the total count of trainable weights and biases in a neural network. This metric is crucial as it determines the model's capacity to learn from data. A model with more parameters can potentially capture more complex patterns and relationships in the data. However, this increased capacity can also lead to higher memory usage and longer training times and may increase the risk of overfitting if not appropriately managed.
- FLOPs (B), or floating point operations per second, measure the computational complexity of a model expressed in billions. This metric is essential for evaluating the efficiency and speed of a model. A lower FLOP count generally indicates a faster and less resource-intensive model, which is particularly beneficial for real-time applications and deployment on devices with limited computational resources.

3. Results

In this section, the practical application of the system is illustrated through a real-life example: the classification of strawberries by the presence of defects. Additionally, data collected from the utilization of this system in the mentioned SME is presented to provide insights into its effectiveness and impact in real-world scenarios.

To demonstrate the efficacy of the developed system, a real-world case study is presented, focusing on the unripe strawberries (Class B) or the detection of strawberries exhibiting grey mold, rot, and mechanical damage (Class C). The performance of various YOLOv8 and YOLOv9 models on detecting strawberry defects is presented in Table 2a,b, respectively. The results reveal distinct differences in model complexity, speed, and detection accuracy.

Table 2. (**a**) Summary of key performance metrics for YOLOv8 Models. (**b**) Summary of key performance metrics for YOLOv9 Models.

(a)					
Metric	**YOLOv8 n**	**YOLOv8 s**	**YOLOv8 m**	**YOLOv8 l**	**YOLOv8 x**
p	0.828	0.854	0.826	0.497	0.536
r	0.957	0.863	0.878	0.992	0.933
mAP	0.945	0.942	0.896	0.601	0.581
mAP50–95	0.931	0.920	0.870	0.588	0.554
F1 score	0.888	0.859	0.851	0.661	0.679
layers	168	168	218	268	268
parameters	3,006,038	11,126,358	25,840,918	43,608,150	68,125,494
FLOPs (B)	8.7	28.6	78.9	165.2	257.8
(b)					
Metric	**YOLOv9 m**	**YOLOv9 c**	**YOLOv9 t**	**YOLOv9 s**	**YOLOv9 e**
p	0.761	0.813	0.832	0.831	0.498
r	0.899	0.936	0.889	0.877	0.997
mAP	0.918	0.931	0.938	0.934	0.597
mAP50–95	0.902	0.906	0.92	0.913	0.582
F1 score	0.824	0.870	0.860	0.853	0.664
layers	374	384	917	486	687
parameters	20,014,438	25,320,790	2,005,798	7,167,862	57,377,942
FLOPs (B)	76.3	102.1	7.7	26.4	189.0

The YOLOv8n model, with 168 layers and 3,006,038 parameters, achieved high precision (*p*) of 0.828 and recall (r) of 0.957, resulting in a mean average precision (mAP) of 0.945 at 50% IoU and 0.931 across IoUs ranging from 50% to 95%. The YOLOv8s model, also with 168 layers but significantly more parameters (11,126,358), exhibited slightly higher precision at 0.854 but lower recall at 0.863, achieving a mAP of 0.942 and mAP50–95 of 0.92. The YOLOv8m model, with 218 layers and 25,840,918 parameters, demonstrated a balanced performance with a precision of 0.826 and recall of 0.878, yielding a mAP of 0.896 and mAP50–95 of 0.87. The YOLOv8l model, with 268 layers and 43,608,150 parameters, showed a significant drop in precision to 0.497 but maintained a very high recall of 0.992. This model achieved a mAP of 0.601 and a mAP50–95 of 0.588. Lastly, the YOLOv8x model, with the highest complexity of 268 layers and 68,125,494 parameters, had a precision of 0.536 and recall of 0.933, with a mAP of 0.581 and mAP50–95 of 0.554. The calculated F1 scores reveal that the smaller YOLOv8 models (n, s, and m) consistently demonstrate higher harmonic mean values of precision and recall, indicating better overall performance and balance in detecting and correctly identifying strawberry defects compared to the larger YOLOv8 models (l and x), which show lower F1 scores despite their higher computational complexity. Regarding computational speed on CPU using ONNX, YOLOv8 x emerges as the slowest among the variants, requiring considerable processing time. This characteristic suggests potential challenges in applications where rapid detection is necessary. The disparity in model parameters and FLOPs across YOLOv8 n to YOLOv8 x highlights trade-offs between model complexity and computational efficiency. YOLOv8 n, with fewer parameters, offers a lightweight alternative suitable for resource-constrained environments, whereas YOLOv8 x, with higher FLOPs, demands greater computational power but promises enhanced detection accuracy.

The results indicate a trade-off between model complexity, speed, and detection performance. Larger models are expected to perform better due to their increased capacity. However, given the limited number of images and the inability of our configuration to handle larger batch sizes, the larger models proved inadequate for this project under the constraints we faced. Simpler models like YOLOv8n and YOLOv8s provided faster inference with high detection accuracy, whereas more complex models like YOLOv8m, YOLOv8l, and YOLOv8x while offering nuanced detection capabilities, resulted in slower inference speeds and varying detection accuracies.

The YOLOv9 series introduces several advancements over its predecessors, showcasing a range of models tailored to different needs and capabilities. The YOLOv9m model, with 374 layers and 20,014,438 parameters, achieved a precision of 0.761 and a high recall of 0.899, resulting in a mean average precision (mAP) of 0.918 and a mAP across IoUs from 50% to 95% of 0.902. This model balances accuracy and efficiency, demonstrating a solid F1 score of 0.824. In comparison, the YOLOv9c model, with 384 layers and 25,320,790 parameters, exhibits slightly better precision at 0.813 and a recall of 0.936. It achieved an mAP of 0.931 and a mAP50–95 of 0.906. The YOLOv9c model's F1 score of 0.870 indicates a well-rounded performance with a good balance of precision and recall. The YOLOv9t model, featuring 917 layers and 2,005,798 parameters, demonstrates the highest precision at 0.832 but a lower recall of 0.889. It recorded a mAP of 0.938 and a mAP50–95 of 0.920, with an F1 score of 0.860, reflecting its strong precision but slightly lower recall. The YOLOv9s model, with 486 layers and 7,167,862 parameters, presents a precision of 0.831 and a recall of 0.877, achieving a mAP of 0.934 and a mAP50–95 of 0.913. This model maintains a good balance between precision and recall, with an F1 score of 0.853. The YOLOv9e model, characterized by its substantial complexity with 687 layers and 57,377,942 parameters, shows lower precision at 0.498 but an exceptionally high recall of 0.997. This results in a mAP of 0.597 and a mAP50–95 of 0.582. Despite its lower F1 score of 0.664, the YOLOv9e model highlights a trade-off between precision and recall.

Overall, the YOLOv9 series demonstrates a range of trade-offs between model complexity, precision, recall, and computational demands. The YOLOv9m and YOLOv9c models offer strong performance with balanced precision and recall, while the YOLOv9t model excels in precision. The YOLOv9s model provides a good compromise, whereas the YOLOv9e model excels in recall but sacrifices precision. Each model in the YOLOv9 series caters to different use cases, balancing accuracy and computational efficiency according to specific needs.

The following images (Figure 3) showcase the performance confidence curves for the YOLOv8n model, which was identified as the best-performing model in the evaluation process. YOLOv8n is generally better than YOLOv9m and YOLOv9c regarding performance metrics (p, r, mAP, F1 score) and computational efficiency. YOLOv8n provides better accuracy and efficiency, making it a more suitable choice if you need a balance of high performance and lower computational demand. YOLOv9m and YOLOv9c offer adequate precision and recall but at a significantly higher computational cost. Therefore, YOLOv8n was selected as the preferable model for practical application where resource efficiency is important.

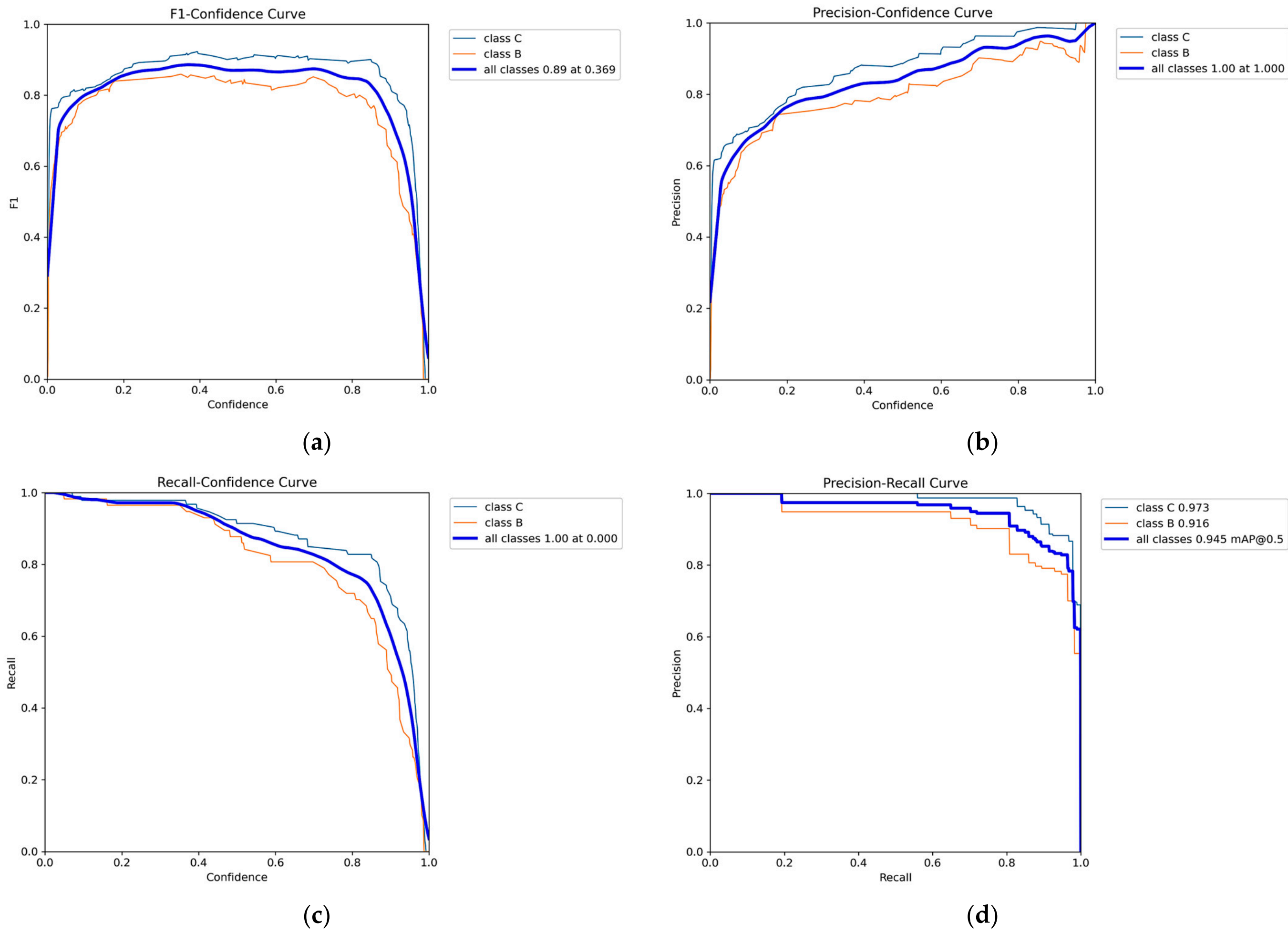

Figure 3. YOLOv8-N performance evaluation metrics. (**a**) F1 score curve; (**b**) *p* curve; (**c**) r curve; (**d**) *pr* curve.

Each curve represents the score for a specific class (B, C) and the overall performance across all classes. The blue line is for Class C, the orange line is for Class B, and the thick, dark blue line is for the overall score for all classes. The F1–confidence curve (Figure 3a) includes the performance for classes B and C, as well as the overall performance across all classes, with the maximum F1 score of 0.89 reached at a confidence threshold of 0.369. This indicates that a confidence threshold of 0.369 is recommended for optimal overall model performance. The curve for Class C demonstrates better and more stable performance than Class B, whose F1 score declines sharply at higher confidence thresholds, suggesting that the model struggles more with precision or recall for Class B at higher thresholds. This analysis aids in selecting the optimal confidence threshold, balancing precision and recall, and provides a detailed understanding of the model's overall performance for individual classes. As the confidence threshold increases from 0 to 1, the precision (Figure 3a) improves steadily for both individual classes and the aggregate. The model achieves perfect precision (1.00) for all classes at a confidence threshold of 1.0, indicating that all predicted positives are true positives at this threshold. The precision for Class C consistently surpasses that of Class B across the confidence spectrum, suggesting the model performs better in predicting Class C with fewer false positives. As the confidence threshold increases from 0 to 1, recall generally decreases (Figure 3c). The model achieves perfect recall (1.00) for all classes at a confidence threshold of 0.000, indicating that at this threshold, all actual positives are correctly identified by the model. The recall for Class C is consistently higher than for Class B across the confidence spectrum, suggesting that the model performs better in capturing true positives for Class C. As the confidence threshold increases, the recall for both classes declines, with a more pronounced drop observed for Class B, highlighting the model's diminishing ability to identify true positives at higher confidence levels. Finally, the *p*r curve (Figure 3d) for Class C is the highest, indicating that the classifier is most precise for this class. The *p*r curve for Class B is lower than the *p*r curve for Class C but still quite high. The overall mean average precision (mAP@0.5) is a performance measure of the classifier, calculated as the average precision at 11 points on the *p*r curve, with a threshold of 0.5 for each point. The mAP@0.5 in the figure is 0.945, which is a relatively high value. The *p*r curve in the figure shows that the classifier is quite precise for all three classes. The classifier is most precise for Class C and less precise for Class B. The overall mean average precision of the classifier is relatively high.

In conclusion, the performance confidence curves for the YOLOv8n model demonstrate its effectiveness across different classes and overall. The F1–confidence curve indicates that the optimal confidence threshold for the model is 0.369, where it achieves a maximum F1 score of 0.89. The model shows better and more stable performance for Class C than Class B, suggesting that the model encounters difficulties with precision or recall for Class B at higher thresholds. The precision–confidence curve reveals that precision steadily improves as the confidence threshold increases, reaching perfect precision at a threshold of 1.0. The model is more precise in predicting Class C, with fewer false positives, than Class B.

Conversely, the recall–confidence curve shows a general decrease in recall as the confidence threshold rises, with perfect recall at a threshold of 0.000. The model captures true positives more effectively across the confidence spectrum for Class C. The PR curve further confirms that the classifier is most precise for Class C, with high precision also observed for Class B and overall. The high mean average precision (mAP@0.5) of 0.945 underscores the classifier's strong overall performance. These findings highlight the YOLOv8n model's robustness and precision, particularly for Class C, while also indicating areas for improvement in Class B predictions.

The following images (Figure 4) showcase the results obtained using the YOLOv8n model. The model's predictions include the detected object classes and the corresponding confidence scores.

Through the implemented solution, it is possible to document and report instances of defects effectively. Thus, subsequent analysis by decision-makers may lead to the establishment of corrective measures, particularly in the case of grey mold, which has been deemed a product defect requiring attention.

A participating company has undergone the implementation and certification of following the IFS FOOD standard. Adopting IFS FOOD standard requirements enables this organization to effectively manage the interrelationships and dependencies among various processes, thereby enhancing overall organizational performance.

The implementation of the adopted IFS FOOD standard requirements offers several benefits, including:

(a) Ensures consistent compliance with food safety regulations and standards, reducing the risk of contamination and foodborne illnesses;
(b) Promotes rigorous quality control measures, resulting in higher product quality and reliability;
(c) Builds trust among consumers by demonstrating a commitment to high safety and quality standards;
(d) Ensuring the achievement of efficient process performance;
(e) Providing a foundation for continuous process improvement by evaluating relevant data and information.

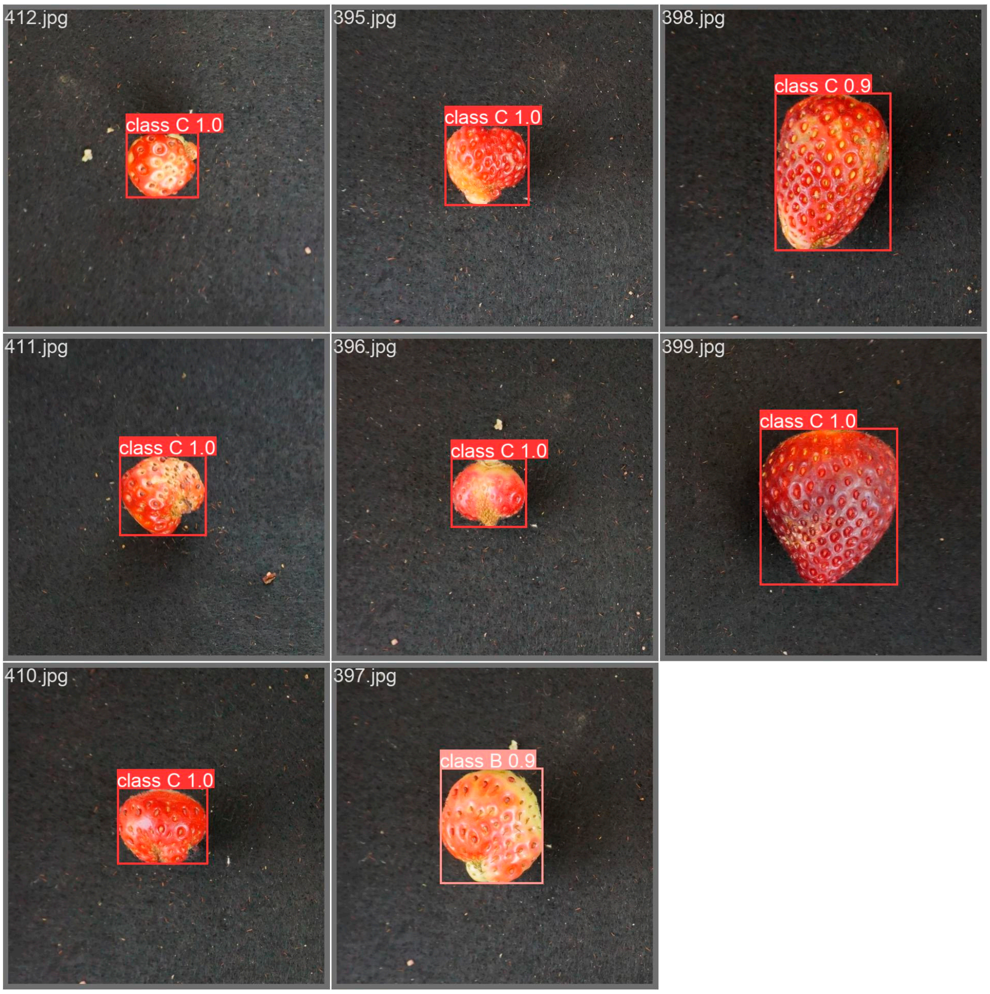

Figure 4. Detection results using the YOLOv8-N model with predicted classes and confidence scores.

The implemented solution offers access to statistical data derived from previous defects classification, enhancing quality control and defects assessment processes. These statistical data are instrumental in formulating recommendations and implementing corrective measures [37]. During three months, differences in defect detection between visual inspection performed by human resources (three practitioners, two female and one male, mean age 30.9 years, SD = 3.2 years, mean work experience 7.7 years) and that carried out using applied software (YOLOv8 n) were examined. The detection results are presented in Table 3, summarising the key insights and outcomes derived from the statistical data analysis, facilitating informed decision-making and continuous improvement initiatives within the agricultural domain, particularly in classifying and processing strawberry defects.

The results indicate differences in the number of identified defects between human and software inspections. On average, software inspection identified more defective strawberries in both classes than human inspection. For instance, software inspection identified 586 defects in Class B, whereas human inspection identified 562 defects. In Class C, software inspection identified 15,544 defects, while human inspection identified 14,029 defects. These findings suggest that software inspection may be more effective in identifying defective strawberries, especially in Class C, where defects are likely more serious and harder to detect with the unaided eye.

Table 3. Comparison of visual inspection results obtained by human resources and software solution utilization in observed organization.

Month	Defect Class	Number of Identified Defects	
		Human Resources Inspection	Applied Software Inspection
1	Class B	163	190
	Class C	5128	5554
2	Class B	188	195
	Class C	3653	3964
3	Class B	211	201
	Class C	5248	6026
Overall	Class B	562	586
	Class C	14,029	15,544

Software inspection can provide more consistent results in identification, reducing the subjectivity that may be present in human inspection. This solution facilitates key quality management principles, including evidence-based decision-making for quality managers.

4. Discussion

4.1. The Significance of Real-Time Reporting of Nonconformities via Edge Devices

Modern agricultural organizations can benefit from integrating advanced technologies like edge devices for real-time defect reporting. The continuous evolution of SA solutions offers high availability, constant data accessibility, and remote services, allowing organizations to enhance their business capabilities while minimizing costs. SA technologies enable customizable, flexible, collaborative, and reconfigurable services, facilitating a unified human–machine manufacturing system.

Our case study, implemented within an agricultural company, demonstrates how small and medium-sized enterprises can expand their in-house, production-oriented applications to include efficiency-enhancing SA solutions. By using YOLO models trained on cloud-stored image data and deploying them on local edge devices, our defect detection solution promotes continuous improvement in quality management. The edge devices identify and report defects throughout manufacturing, emphasizing evidence-based decision-making and improvement.

A Dahua 2MP network bullet camera with full-color technology was employed for this application. This camera operates effectively in low light conditions and provides a maximum resolution of 1920×1080. It features a fixed 2.8 mm lens, H.265+/H.264+ compression, and an IP67 protection rating, making it suitable for various environments. The camera's video analytics capabilities, such as Tripwire and intrusion detection, enhance its functionality. The local server used for processing is a Lenovo W541 with an Intel i7-4810MQ processor, 32 GB DDR3 RAM, and a 512 GB SSD. This configuration supports the computational demands of the system, facilitating real-time decision-making based on captured data.

4.2. Implementation Justification of a Modular Software for Defect Detection in Smart Agriculture

The novelty of this research lies in developing a modular software solution tailored for identifying and reporting defects within the context of SA, strawberry classification, and processing. The software solution is developed within an environment comprising MySQL for the database, Python for the application tier, and JavaScript for the presentation tier. This setup leverages the capabilities of SA technologies to create a framework conducive to realizing the quality management objectives.

Central to this paper is the fusion of SA research principles with quality objectives aimed at achieving conformity to prevent defects and align strategic objectives with operational enhancements. Organizations can leverage the software solution's inherent modularity, and organizations can streamline defect identification and reporting processes, fostering greater transparency, efficiency, and effectiveness.

In essence, this research represents an effort in smart agriculture and strawberry processing by offering a comprehensive software solution that harnesses the transformative potential of SA technologies to usher in a new era of quality management and operational excellence.

The achievement of quality objectives is facilitated through the enhancement of connectivity with reports of emerging insights, the acquisition of knowledge through the utilization of appropriate YOLOv8 and YOLOv9 models, and the automation of problem reporting and resolution via intelligent agents using mobile devices.

5. Conclusions

Achieving superior quality standards and meeting consumer requirements are primary objectives for modern agricultural companies. The study's findings underscore the potential of integrating SA-driven quality control and advanced information and communication technologies (ICT) to enhance defect detection in strawberry processing.

The developed software solution, leveraging YOLOv8 and YOLOv9 models, demonstrates significant promise in real-world applications. A real-life case study that classified strawberries by detecting defects such as grey mold, rot, and mechanical damage provided valuable insights into the system's effectiveness. Performance metrics of various YOLOv8 models revealed critical differences in detection accuracy, speed, and computational complexity, as summarised in Table 2. The YOLOv8n model, with its balance of precision and recall, emerged as the most effective in terms of overall performance and computational efficiency.

The novelty of our research can be underlined from multiple angles: (1) Integration of Advanced Algorithms: We employ the latest YOLOv8 algorithms (n, s, m, l, and x) for defect detection, which have not been extensively applied in the context of agricultural product inspection. These algorithms offer enhanced speed and accuracy, making them highly suitable for real-time applications. (2) Edge Device Deployment: The use of a Dahua 2MP network bullet camera with full-color technology, combined with a local server (Lenovo W541) for real-time processing, represents a significant advancement. This setup allows for immediate defect detection and response, which is crucial for maintaining the quality of perishable goods like strawberries. (3) Real-World Validation: The implemented system was tested in a real-world environment, proving its reliability and effectiveness. This practical validation underscores the applicability and robustness of our solution in operational settings. (4) Interdisciplinary Approach: Our research bridges multiple disciplines, including computer vision, machine learning, and agricultural engineering. This interdisciplinary approach enhances the overall impact and applicability of the research, providing a robust framework for future advancements in the field.

5.1. Key Findings and Practical Implications

Model Performance: The YOLOv8n model achieved high precision (0.828) and recall (0.957), with a mean average precision (mAP) of 0.945, making it effective for defect detection. In contrast, larger models like YOLOv8l, YOLOv8x, YOLOv9m, and YOLOv9c, while offering nuanced detection capabilities, showed slower speeds and varied detection accuracies. This indicates a trade-off between model complexity and performance, highlighting the need for balancing accuracy with computational efficiency.

Real-World Application: The system's practical application was illustrated through defect classification, demonstrating its effectiveness in identifying strawberries with defects. Data collected from the SME showcased a higher detection rate using the software compared to human inspection, particularly for Class C defects, which are more challenging to detect visually.

The software solution enhances defect reporting and quality control accuracy through digitalization—streamlining defect reporting with a user-friendly interface—and accessibility—offering an affordable, user-friendly solution suitable for the strawberry processing industry.

Implementing the IFS FOOD standard in the participating company further enhances quality control by ensuring compliance with food safety regulations, promoting rigorous quality measures, and fostering continuous process improvement.

5.2. Additional Insights and Future Directions

Comparison with Human Inspection: The software's ability to identify a higher number of defects than human inspection, especially in Class C, indicates its effectiveness in providing consistent and objective defect detection.

Model Optimization: The study underscores the importance of tailoring machine learning models to the specific constraints of the deployment environment. Future work could focus on optimizing these models for larger datasets and improving hardware configurations to enhance their potential further.

Expansion and Integration: Potential areas for development include modules for documentation management and integration with other organizational systems, such as safety or environmental management systems, to facilitate holistic process optimization.

The study demonstrates that integrating advanced SA and ICT technologies into quality control processes can significantly improve defect detection and management. This research provides valuable insights for enhancing quality management practices and operational excellence in the agricultural sector by offering a robust and efficient solution.

The system possesses the potential for enhancement and extension in various avenues. One direction for advancement includes the development of modules for documentation management, enabling more efficient handling of documentation related to quality control processes. Moreover, interconnecting the system with other organizational systems, such as those dedicated to safety or environmental management, can facilitate holistic management and optimization of operational processes.

Author Contributions: Conceptualisation, M.S. and M.E.; methodology, A.D.; software, N.P.; validation, R.J. and N.P.; formal analysis, A.D.; investigation, R.J.; resources, R.J.; data curation, N.P.; writing—original draft preparation, A.D.; writing—review and editing, M.S. and M.E.; visualization, A.D.; supervision, M.S.; project administration, A.D.; funding acquisition, M.S. and A.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

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

References

1. Irpan, M.; Summantri, A.; Kurniawati, M.F.; Sukmana, R.A.; Shaddiq, S. Digital communication in agricultural extension in the era of the industrial revolution 4.0. *J. Eng. Manag. Inf. Technol.* **2023**, *1*, 177–190. [CrossRef]
2. Oo, L.M.; Aung, N.Z. A simple and efficient method for automatic strawberry shape and size estimation and classification. *Biosyst. Eng.* **2018**, *170*, 96–107. [CrossRef]
3. He, Z.; Karkee, M.; Zhang, Q. Detecting and localizing strawberry centers for robotic harvesting in field environment. *IFAC-Pap.* **2022**, *55*, 30–35. [CrossRef]
4. Hassan, H.E.; Abd El-Rahman, A.A.; Khalid, A.A. Quality evaluation of strawberry fruit using visible laser. *Agric. Eng. Int. CIGR J.* **2018**, *20*, 157–163.

5. Afzaal, U.; Bhattarai, B.; Pandeya, Y.R.; Lee, J. An instance segmentation model for strawberry diseases based on mask R-CNN. *Sensors* **2021**, *21*, 6565. [CrossRef] [PubMed]
6. Mijailović, Đ.; Đorđević, A.; Stefanović, M.; Erić, M. Identification of product non-conformities using computer vision algorithms. *J. Eng. Manag. Inf. Technol.* **2023**, *1*, 191–198. [CrossRef]
7. Mijailović, Đ.; Đorđević, A.; Stefanović, M.; Erić, M. Quality control in the manufacturing industry based on the application of computer vision. *J. Innov. Bus. Ind.* **2023**, *1*, 161–166. [CrossRef]
8. Song, C.; Wang, C.; Yang, Y. Automatic detection and image recognition of precision agriculture for citrus diseases. In Proceedings of the 2020 IEEE Eurasia Conference on IOT, Communication and Engineering (ECICE), Yunlin, Taiwan, 23–25 October 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 187–190.
9. Qin, Z.; Wang, W.; Dammer, K.H.; Guo, L.; Cao, Z. Ag-YOLO: A real-time low-cost detector for precise spraying with case study of palms. *Front. Plant Sci.* **2021**, *12*, 753603. [CrossRef]
10. Lippi, M.; Bonucci, N.; Carpio, R.F.; Contarini, M.; Speranza, S.; Gasparri, A. A yolo-based pest detection system for precision agriculture. In Proceedings of the 2021 29th Mediterranean Conference on Control and Automation (MED), Virtual, 22–25 June 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 342–347.
11. Junos, M.H.; Mohd Khairuddin, A.S.; Thannirmalai, S.; Dahari, M. An optimized YOLO-based object detection model for crop harvesting system. *IET Image Process.* **2021**, *15*, 2112–2125. [CrossRef]
12. Mirhaji, H.; Soleymani, M.; Asakereh, A.; Mehdizadeh, S.A. Fruit detection and load estimation of an orange orchard using the YOLO models through simple approaches in different imaging and illumination conditions. *Comput. Electron. Agric.* **2021**, *191*, 106533. [CrossRef]
13. Wang, Y.; Yan, G.; Meng, Q.; Yao, T.; Han, J.; Zhang, B. DSE-YOLO: Detail semantics enhancement YOLO for multi-stage strawberry detection. *Comput. Electron. Agric.* **2022**, *198*, 107057. [CrossRef]
14. Cuong, N.H.H.; Trinh, T.H.; Meesad, P.; Nguyen, T.T. Improved YOLO object detection algorithm to detect ripe pineapple phase. *J. Intell. Fuzzy Syst.* **2022**, *43*, 1365–1381. [CrossRef]
15. Tian, Y.; Wang, S.; Li, E.; Yang, G.; Liang, Z.; Tan, M. MD-YOLO: Multi-scale Dense YOLO for small target pest detection. *Comput. Electron. Agric.* **2023**, *213*, 108233. [CrossRef]
16. Dang, F.; Chen, D.; Lu, Y.; Li, Z. YOLOWeeds: A novel benchmark of YOLO object detectors for multi-class weed detection in cotton production systems. *Comput. Electron. Agric.* **2023**, *205*, 107655. [CrossRef]
17. Cui, B.; Liang, L.; Ji, B.; Zhang, L.; Zhao, L.; Zhang, K.; Shi, F.; Créput, J.C. Exploring the YOLO-FT Deep Learning Algorithm for UAV-Based Smart Agriculture Detection in Communication Networks. *IEEE Trans. Netw. Serv. Manag.* **2024**, *in press*. [CrossRef]
18. Badgujar, C.M.; Poulose, A.; Gan, H. Agricultural object detection with You Only Look Once (YOLO) Algorithm: A bibliometric and systematic literature review. *Comput. Electron. Agric.* **2024**, *223*, 109090. [CrossRef]
19. Lu, Y.; Gong, M.; Li, J.; Ma, J. Strawberry Defect Identification Using Deep Learning Infrared–Visible Image Fusion. *Agronomy* **2023**, *13*, 2217. [CrossRef]
20. Li, M.; Zhang, Z.; Lei, L.; Wang, X.; Guo, X. Agricultural greenhouses detection in high-resolution satellite images based on convolutional neural networks: Comparison of faster R-CNN, YOLO v3 and SSD. *Sensors* **2020**, *20*, 4938. [CrossRef]
21. *ISO 9001:2015*; Quality management systems—Requirements. International Organization for Standardization (ISO): Geneva, Switzerland, 2015. Available online: https://www.iso.org/standard/62085.html (accessed on 1 August 2024).
22. *IFS Food Standard, Version X*; IFS Management GmbH: Berlin, Germany, 2024. Available online: https://www.ifs-certification.com/ (accessed on 1 August 2024).
23. Senthil Kumar, A.; Suresh, G.; Lekashri, S.; Babu Loganathan, G.; Manikandan, R. Smart agriculture system with E–carbage using IoT. *Int. J. Mod. Agric.* **2021**, *10*, 928–931.
24. Chopra, G.; Whig, P. Smart agriculture system using AI. *Int. J. Sustain. Dev. Comput. Sci.* **2022**, *4*, 1–2.
25. Bhola, A.; Srivastava, S.; Noonia, A.; Sharma, B.; Narang, S.K. A status quo of machine learning algorithms in smart agricultural systems employing IoT-based WSN: Trends, challenges and futuristic competences. In *Machine Intelligence, Big Data Analytics, and IoT in Image Processing: Practical Applications*; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2023; pp. 177–195.
26. Sinha, B.B.; Dhanalakshmi, R. Recent advancements and challenges of Internet of Things in smart agriculture: A survey. *Future Gener. Comput. Syst.* **2022**, *126*, 169–184. [CrossRef]
27. Javaid, M.; Haleem, A.; Singh, R.P.; Suman, R. Enhancing smart farming through the applications of Agriculture 4.0 technologies. *Int. J. Intell. Netw.* **2022**, *3*, 150–164. [CrossRef]
28. Du, X.; Cheng, H.; Ma, Z.; Lu, W.; Wang, M.; Meng, Z.; Jiang, C.; Hong, F. DSW-YOLO: A detection method for ground-planted strawberry fruits under different occlusion levels. *Comput. Electron. Agric.* **2023**, *214*, 108304. [CrossRef]
29. An, Q.; Wang, K.; Li, Z.; Song, C.; Tang, X.; Song, J. Real-time monitoring method of strawberry fruit growth state based on YOLO improved model. *IEEE Access* **2022**, *10*, 124363–124372. [CrossRef]
30. Luo, Q.; Wu, C.; Wu, G.; Li, W. A Small Target Strawberry Recognition Method Based on Improved YOLOv8n Model. *IEEE Access* **2024**, *12*, 14987–14995. [CrossRef]
31. Bai, Y.; Yu, J.; Yang, S.; Ning, J. An improved YOLO algorithm for detecting flowers and fruits on strawberry seedlings. *Biosyst. Eng.* **2024**, *237*, 1–12. [CrossRef]
32. Nturambirwe, J.F.I.; Opara, U.L. Machine learning applications to non-destructive defect detection in horticultural products. *Biosyst. Eng.* **2020**, *189*, 60–83. [CrossRef]

33. Popkova, E.G.; Litvinova, T.N.; Kolieva, A.E.; Dugina, T.A.; Agalakova, O.S. Increasing food quality at smart farms as a promising path for the sustainable development of agriculture and food security. *Int. J. Qual. Res.* **2023**, *17*, 1273–1276. [CrossRef]
34. Zhang, F.; Lin, Y.; Zhu, Y.; Li, L.; Cui, X.; Gao, Y. A Real-Time Sorting Robot System for Panax Notoginseng Taproots Equipped with an Improved Deeplabv3+ Model. *Agriculture* **2022**, *12*, 1271. [CrossRef]
35. Zhang, W.; Zhu, Q.; Huang, M.; Guo, Y.; Qin, J. Detection and classification of potato defects using multispectral imaging system based on single shot method. *Food Anal. Methods* **2019**, *12*, 2920–2929. [CrossRef]
36. Feldmann, M.J.; Hardigan, M.A.; Famula, R.A.; López, C.M.; Tabb, A.; Cole, G.S.; Knapp, S.J. Multi-dimensional machine learning approaches for fruit shape recognition and phenotyping in strawberry. *bioRxiv* **2019**. [CrossRef]
37. 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]