Computers and Electronics in Agriculture 216 (2024) 108474

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Multi-view gripper internal sensing for the regression of strawberry ripeness using a mini-convolutional neural network for robotic harvesting

Yuanyue Ge [a], Pål Johan From [a], Ya Xiong [b,a,*]

[a] Faculty of Science and Technology, Norwegian University of Life Sciences, Ås 1430, Norway
[b] Intelligent Equipment Research Center, Beijing Academy of Agriculture and Forestry Sciences, Beijing 100097, China

ARTICLE INFO

Keywords:
Ripeness estimation
Deep learning for regression
Gripper internal sensing
Lightweight models
Strawberry-harvesting robots

ABSTRACT

The capability of robotic fruit-harvesting systems to accurately assess the ripeness of fruits is crucial for fulfilling the diverse standards of the market and the preferences of consumers. Existing studies involving fruit ripeness estimation mainly focus on detecting of ripe strawberries as one class or classifying of ripeness into several stages, such as overripe, ripe, and unripe. Current harvesting robots also lack the ability to determine the ripeness of the back of fruit with respect to the robots. This paper proposes a lightweight convolutional neural network (CNN) regression model for ripeness quantification based on the internal image sensing system of the gripper with full-view coverage of the fruit for strawberry-harvesting robots. A gripper internal sensing system was developed using two RGB cameras that could provide full-view fruit coverage for a more accurate estimation of fruit ripeness. Four base CNN networks capable of feature learning were used for feature extraction, followed by the utilization of newly added dense layers that produced a regressed value to represent the strawberry ripeness. However, the base networks were cumbersome and relatively slow due to their complex structures. To simplify this, a new MiniNet with fewer convolutional layers was proposed to reduce the model size and inference time. All models were trained via two loss functions, mean square error and Huber loss. The results showed that the models trained via Huber loss performed better. An Xception model trained on Huber loss showed the best performance with a mean absolute error of 4.0% and an average inference time of 42.5 ms. Of all the models, the new MiniNet was the most lightweight and fastest model while maintaining high performances (an mean absolute error of 4.8% and an inference time of 6.5 ms for Huber loss trained model). The proposed method may be also applicable to other fruit-harvesting systems.

1. Introduction

Strawberries (*Fragaria* × *Ananassa*) are valuable fruits that require accurate ripeness estimation before delivery to markets. Ripeness estimation is essential not only for harvesting, but also for fruit yield estimation and post-harvesting grading. For manual harvesting, the readiness of a strawberry for harvest is determined solely by its surface coloration, specifically the extent of its red area (Sánchez et al., 2012). The International Organization for Standardization (ISO) suggested that a strawberry is deemed harvest-ready when at least three-quarters (75%) of its surface turns red (ISO 6665:1983). However, the standard for determining when a strawberry is commercially ripe can differ across regions and seasons. In China, according to local suppliers, strawberries are ready to pick when they are 70%–95% ripe. In the UK, marketable fresh berries should be picked when they are completely red (100%) (RHS, 2023). As shown in Fig. 1, the ripeness levels of strawberries from markets in the UK, Norway and China are significantly different. Usually, strawberries that require longer transportation and those harvested in warmer weather display more portion of green areas, as observed in strawberries sold in China. Additionally, some consumers prefer slightly tart strawberries, while others might favor fully sweet ones. This preference can influence harvest timing. During robotic harvesting, strawberry ripeness must be determined before detachment. Therefore, a robotic system for harvesting strawberries should possess the ability to discern specific levels of ripeness.

Several studies have investigated various methods for fruit maturity classification. Some detected the target fruit as one class since that was the only category that relevant to the system, such as automatic harvesting robots. Xiong et al. (2020) presented a strawberry-harvesting robot with a machine vision system that used an adaptive color thresholding method to detect ripe strawberries as one class. They correlated the relationship between sunlight intensity and color thresholds to establish a model for ripe strawberry classification in unstructured

* Corresponding author at: Intelligent Equipment Research Center, Beijing Academy of Agriculture and Forestry Sciences, Beijing 100097, China.
E-mail addresses: geyuanyue@hotmail.com (Y. Ge), pal.johan.from@nmbu.no (P.J. From), yaxiong@nercita.org.cn (Y. Xiong).

https://doi.org/10.1016/j.compag.2023.108474
Received 28 February 2023; Received in revised form 13 November 2023; Accepted 22 November 2023
Available online 5 December 2023

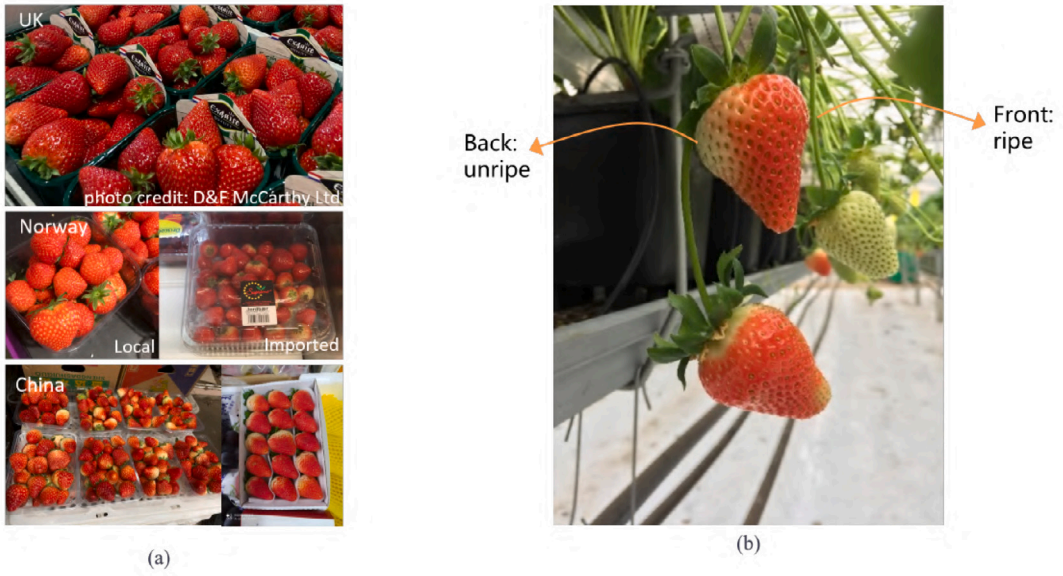


Fig. 1. The importance of comprehensive assessment of fruit ripeness: (a) variations in the ripeness levels of strawberries sold in three countries; (b) within a polytunnel, a strawberry displays a ripe side at the front and unripe side at the back.

environments. Utilizing color as a criterion provided a straightforward and rapid method for the robot's efficient real-time operation.

Although hyperspectral imaging has also been used for ripeness classification, it is expensive and time-consuming. Liu et al. (2014) used a support vector machine (SVM) and principal component analysis-back propagation neural network (PCA-BPNN) to classify strawberry growth stages as unripe, ripe, and overripe. The strawberry data were collected by a multispectral imaging system under controlled light conditions. Guo et al. (2016) used two spectral ranges to categorize strawberries as ripe, mid-ripe, or unripe. They extracted the optimal wavelengths via PCA and used SVM to build a classification model. A laboratory-level imaging system was implemented to collect data and test the SVM model, indicating the efficacy of the hyperspectral imaging method. Similarly, Shao et al. (2020) used discriminant analysis (PLS-DA) and least squares SVM (LS-SVM), to estimate the maturity of strawberries via hyperspectral imaging system. The strawberries were classified into three maturity stages, namely unripe, mid-ripe, and ripe, with a reported accuracy between 91.7% to 96.7%.

In addition to the classification of strawberry ripeness, similar methods were used to characterize the ripeness levels of other fruits. Gutiérrez et al. (2019) used hyperspectral imaging and SVM to estimate mango ripeness. Similarly, Tugnolo et al. (2021) created a model that used PLS-DA to classify four olive ripeness stages.

Instead of classification, several other studies presented methods for fruit maturity regression. Mohamed et al. (2019) used two-dimensional (2D) color to estimate strawberry ripeness by calculating the ratio of ripe pixels to the number of unripe pixels.

Support vector regression (SVR) was also used for ripeness regression. Avila et al. (2015) used a color-combining SVM technique to estimate the maturity of olives and grape seeds. They collected data in a lab environment and proposed a method based on SVR to generate color scales associated with three maturity in three stages: mature, immature, and over-mature. Cho et al. (2021) developed two models for avocado ripeness assessment, one of which was based on an artificial neural network (ANN) and the other on SVR. The estimation models were built via the machine vision system using L*a*b and YUV color spaces.

As deep learning methods have evolved, there has been a growing trend in using convolutional neural networks (CNNs) for fruit target classification. Sa et al. (2016) utilized Faster R-CNN (Ren et al., 2015) combining RGB image data and near-infrared (NIR) data, to classify different fruits, including capsicums, oranges, strawberries, mangoes, apples, avocados, rock melons and sweet peppers. Williams et al. (2019) presented a kiwifruit-harvesting robot that utilized a fully convolutional network (FCN) (Long et al., 2015), while Fu et al. (2020) employed two Faster R-CNN-based (Ren et al., 2015) systems for apple detection. Liu et al. (2022) employed a semantic segmentation model to categorize strawberries into two phases: ripe and unripe, and visualized all the identified berries on a farm using the simultaneous localization and mapping (SLAM) method. MacEachern et al. (2023) applied several deep learning models to detect the ripeness stages of wild blueberries. They found that dividing blueberries into two ripeness stages (ripe, unripe) yielded slightly better results than categorizing them into three phases (green, red, and blue). In all these examples, the ripe fruits were detected as one class. Miao et al. (2023) also detected individual ripe tomatoes as a single category, but they went a step further by computing the ripeness of a cluster of tomatoes, taking into account the proportion of ripe to unripe individual tomatoes.

Other studies detected and classified ripe strawberries into different classes to distinguish their various stages of growth. Ge et al. (2019) used Mask-R-CNN (He et al., 2017) to classify strawberry ripeness into three classes, namely ripe, pink, and unripe, to represent the strawberry growth stages of the fruit. Data were collected from strawberry tunnels, with F1 scores ranging from 0.81 to 0.94. Gao et al. (2020) used a hyperspectral imaging system with a CNN model to classify ripe and early ripe strawberries. The test results showed an accuracy of 98.6%. In addition, Binder et al. (2022) compared methods using a feature-based classifier and CNN to classify strawberry ripeness into three classes: unripe, ripe, and over-ripe.

CNN-based approaches were also used for dividing other ripe fruits into various ripeness stages. Notably, the categories of ripe and over-ripe were most frequently adopted, evident in studies on oil palm fruit (Suharjito et al., 2023), apple as described by Xiao et al. (2023), banana fruit according to Aherwadi et al. (2022) and Shuprajhaa et al. (2023), watermelon as seen in Alipasandi et al. (2023), among others. Garillos-Manliguez and Chiang (2021) modified several deep convolutional architectures to classify the six growth stages of papaya, including green with a trace of yellow, more green than yellow, a mix of green and yellow, more yellow than green, fully ripe, and overripe. They used multi-modal input consisting of RGB and hyperspectral data, which were collected in a laboratory setting, achieving an F score of 0.91. Ramos et al. (2021) utilized a convolutional network to classify

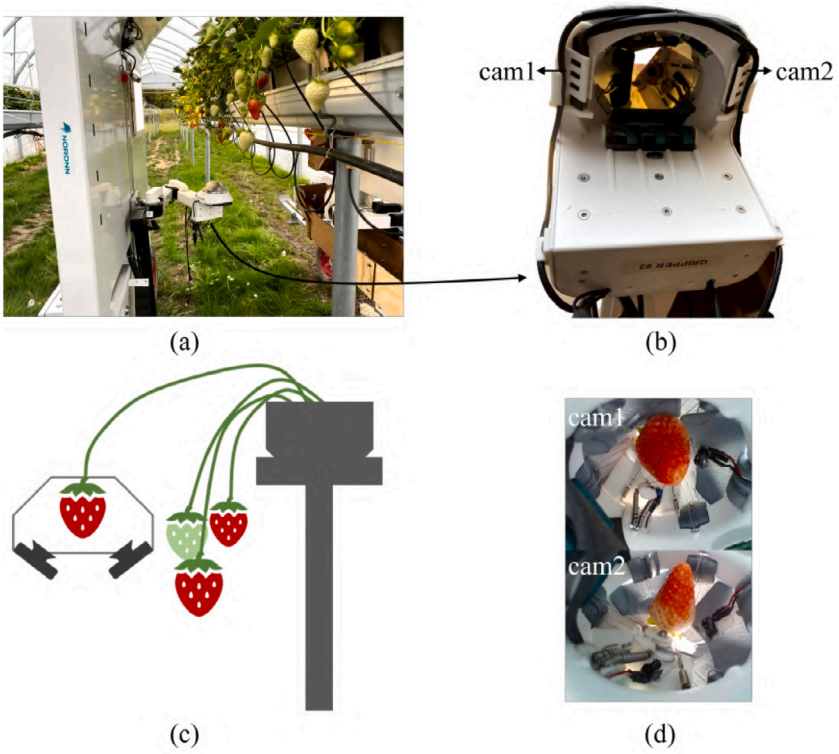


Fig. 2. Multi-view gripper internal sensing system: (a) strawberry-harvesting robot at strawberry polytunnel; (b) multi-view internal sensing system with two small cameras (left camera cam1 and right camera cam2) under the gripper head; (c) schematic to illustrate the data collection; (d) examples of captured images from the two cameras.

the maturity stages of two grape cultivars, which were defined into eight classes. New scenarios consisting of three and four categories were created by gathering the neighboring classes. The results showed that the accuracy varied from 64.52% to 93.93% in different scenarios.

To the authors' knowledge, very few studies are available involving the use of deep CNNs for the continuous regression of fruit ripeness, particularly when it comes to real-time robotic harvesting. CNNs are particularly well-suited for tasks involving spatial data, such as images. In the context of strawberry ripeness, these features might include variations in color intensity, texture, and shape, all of which can contribute to an accurate ripeness regression. Therefore, this work aims to establish a lightweight CNN model for more accurate fruit ripeness estimation to determine a certain ripeness level.

In addition, most fruit-harvesting robots only capture images from the front or sides to estimate the ripeness. To the authors' knowledge, no fruit-harvesting robots can perceive the environment from the back, even when employing a multi-camera perception system. This is mainly because the robot is typically located in front of the target fruit. Another reason is that the space at the back is limited, e.g., between the table and the plants. However, in some cases, the back of the target remains unripe even though the front is ripe (Fig. 1(b)) since the front side normally receives more light, causing more rapid color changes.

To address these challenges, this study proposed a comprehensive fruit ripeness estimation method using a newly developed lightweight CNN model based on a multi-view gripper internal image sensing system for ripeness quantification.

The main contribution of this study is threefold:

- A gripper internal sensing setup was proposed that can detect a strawberry with full-view coverage. Two cameras were mounted inside the gripper, allowing the data to be combined for a comprehensive ripeness estimation.
- Four base CNN networks were modified, compared, and trained via two different loss functions for ripeness regression to provide specific ripeness levels for fruit targets.
- A lightweight CNN network, namely MiniNet, was developed for ripeness regression. Despite its smaller size, this network demonstrated similar performance to the four base networks while offering faster processing speed.

2. Materials and methods

2.1. The development of the multi-view sensing system

This system was developed based on our previous strawberry-harvesting robot (Xiong et al., 2021) (Fig. 2(a)). The gripper was modified with an added multi-view internal sensing system. As shown in Fig. 2(b), two small RGB cameras were mounted on the bottom of the gripper head at a 45-degree angle, facing toward the inside of the gripper using 3D-printed brackets. The two cameras were combined to collect multi-view images when the target strawberry entered the gripper. These were custom-made RGB cameras with a wide angle view (120 degrees) and a resolution of 640 × 480. Fig. 2(c) shows the moment when a strawberry is swallowed by the gripper, with the cameras simultaneously capturing the images. Fig. 2(d) shows an example of the images captured by the two cameras. Multi-view sensing enabled the harvester to capture two images of the swallowed strawberries, which allowed a more comprehensive ripeness estimation.

2.2. Data collection and annotation

2.2.1. Data collection

The data used in this study were collected during the strawberry season in Norway from August to October in 2020 and 2021. The data collection setup is shown in Fig. 2. The two internal cameras captured

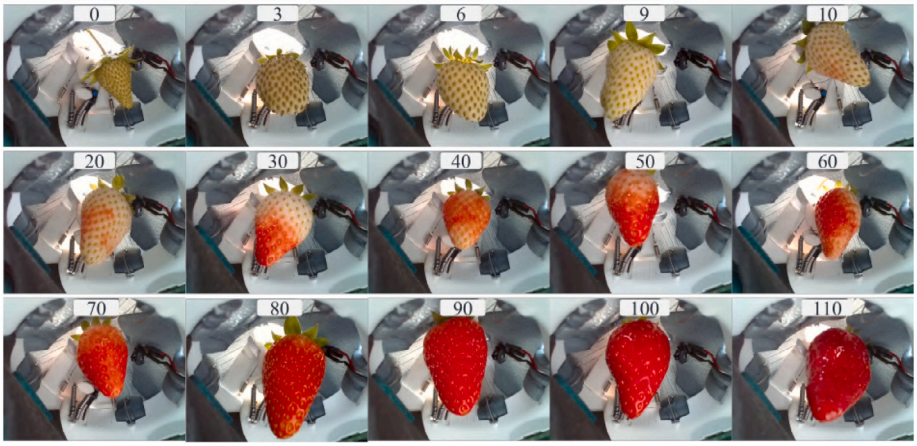


Fig. 3. A reference for labeling strawberry ripeness level. The number at the top of the image indicates the ripeness value (%) of the strawberry, from unripe to overripe. Riper berries tend to have darker red covering a larger part of the fruit body, and the seeds are more sparsely distributed.

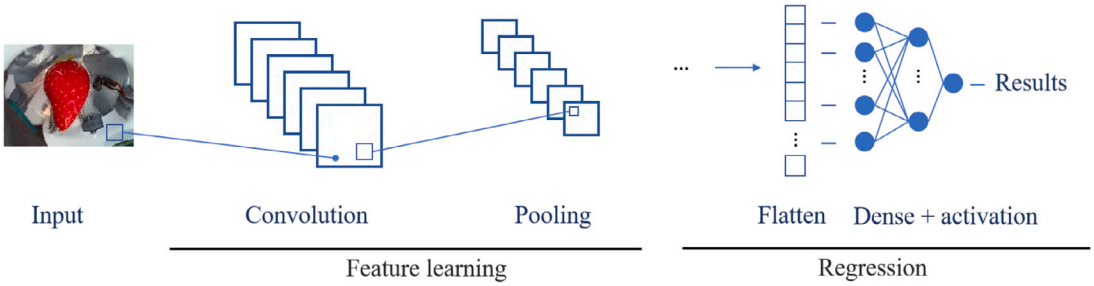


Fig. 4. The network structure for ripeness regression. Base networks were used for feature learning, after which the output was flattened to dense neural networks to ultimately provide a regressed value.

simultaneous images when the gripper swallowed the strawberry. The samples were randomly collected from naturally growing strawberries. The data set covered the strawberries at all stages of maturity. The strawberry plants were grown on table-tops in polytunnels, as shown in Fig. 2(a).

2.2.2. Data annotation

Maturity labeling was to obtain a value representing the ripeness level of the strawberry in the image. Since providing an objective ripeness value is challenging, a reference standard was established before starting the annotation process (Fig. 3), in which the number in each image indicated the strawberry ripeness. Therefore, each labeled image could be compared with this reference to reduce the influence of subjectivity. Our hands-on experience over the past five years with strawberry farmers in the UK and Norway has enriched our understanding of berry maturity indicators: (1) during our numerous field visits and active participation in harvesting sessions, we consistently observed specific color and size variations across different maturity stages of strawberries; (2) a less commonly noted but crucial observation was the sparser distribution of seeds on berries as they approached full ripeness; (3) our continuous interaction with expert farmers not only validated our observations, but also offered invaluable insights that significantly informed our classification of maturity groups. This study used a total of 808 images.

2.3. Deep learning networks for regression

2.3.1. The network structure for regression

First, four existing CNN networks were modified for the ripeness regression. The modified end-to-end network structure capable of providing a regressed value is shown in Fig. 4. The base network for feature learning was used to extract features from the input image. The CNN network output was flattened to dense neural networks, ultimately providing a regressed value. Table 1 provides an overview of the entire network. The layers following the base network included a flattened layer, a dense layer with a ReLU activation, and a dense layer with a linear activation. The output size of the flattened layer was not fixed since the number of nodes was depended on the base network type.

Table 1
An overview of the network layers for achieving regression, including input, base networks, flatten layer, dense layers and a final layer that outputs a regressed value.

Layer	Configuration	Output size
Input	Shape: (128,128,3)	DoN
Base networks	(4 types)	DoN
Flatten	None	DoN
Dense	activation:ReLU	256
Dense	activation:Linear	1

*DoN: Depends on networks.

The data derived from the convolutional layers were flattened and transferred to a fully connected dense layer with 256 nodes and a ReLU activation. The neurons of the dense layers were connected to the neurons in the preceding layer. The operational sequence of a dense layer was, $Output = activation(dot(input, kernel) + bias)$, where the activation involved the element-wise activation function. The dot denoted the computation of the dot product between the input and the kernel along the last axis of the input and axis 0 of the kernel. The ReLU activation function was used for the dense layer after the flattened layer, representing a rectified linear activation that directly passed the input if it was positive, otherwise, it produced zero. The kernel and bias were trainable parameters, in which the kernel represented a weight

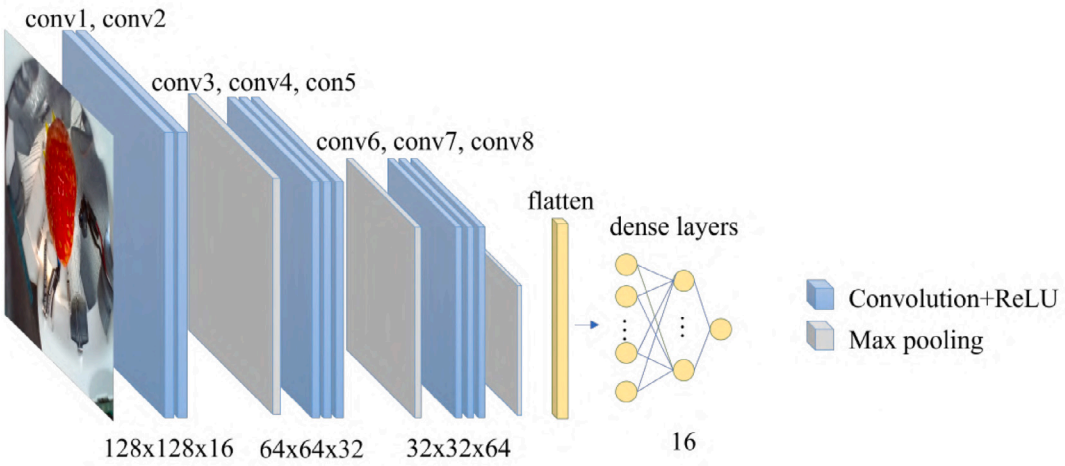


Fig. 5. The structure of the newly developed lightweight MiniNet CNN network.

Table 2
A list of the model size, network depth, time consumption of the four base CNN networks utilized in this study for feature extraction.

Model	Size (MB)	Depth	Time (ms) on CPU
VGG19	549	19	84.8
Xception	88	81	109.4
MobileNetV2	14	105	25.9
ResNet50	98	107	58.2

matrix, and the bias a vector created by the layer. Finally, a single end node was present with a linear activation function.

2.3.2. The four base networks

Four network constructions were used for feature extraction, as shown in Table 2. VGG19 (Simonyan and Zisserman, 2014) is a deep convolutional network designed for large-scale image classification. Normally, there are three fully connected layers in the head layers, in which the final one has 1000 channels. Therefore, the ImageNet pre-trained model could be used to classify 1000 cases. Although ResNet (He et al., 2016) represented a deep neural network with a depth of up to 152 layers, training was simpler. ResNet50 is a 50-layer deep CNN network. Xception (Chollet, 2017) has an easy-to-define architecture and can be easily modified due to the linear layer stack with residual connections. MobileNetV2 (Sandler et al., 2018) is a simple convolutional network that has fewer weight parameters, increasing its efficiency and suitability for mobile applications. Similarly, there are pre-trained models available that were trained on ImageNet and can classify images into 1000 categories, using a softmax activation.

The size, depth and time consumption per inference step on the CPU of the four networks are listed in Table 2. The performance was tested on ImageNet validation dataset and the time was the average of 10 repetitions (Chollet et al., 2015). Normally, when determining the depth of a CNN network, only the layers with trainable weights are included. However, Table 2 refers to the topological depth, which included the activation and batch normalization layers, and employed data derived from Keras documentation. A CPU AMD EPYC processor (with IBPB) (96 cores) was used for the speed test.

2.3.3. Newly developed network structure

Ripeness regression was one of the several functions that used CNN networks in the robotic harvesting system. The CNN-based functions in the harvesting system were mainly used for real-time detection and tracking (Xiong et al., 2021). Only one computer (ZBOX-EN1060) with an integrated GPU (GTX-1060 6 GB) was used for the CNN networks. This study employed a CPU for ripeness regression to reduce GPU resource usage and proposed a lightweight model, namely MiniNet, to further reduce the model size and increase the processing speed.

Table 3
Network configuration of the proposed MiniNet.

Layer	Configuration	Output size
Input	128 × 128 × 3	–
convolution_1+ReLU	3 × 3 with 16 kernels	(128,128,16)
convolution_2+ReLU	3 × 3 with 16 kernels	(128,128,16)
maxpool_1	2 × 2	(64,64,16)
convolution_3+ReLU	3 × 3 with 32 kernels	(64,64,32)
convolution_4+ReLU	3 × 3 with 32 kernels	(64,64,32)
convolution_5+ReLU	3 × 3 with 32 kernels	(64,64,32)
maxpool_2	2 × 2	(32,32,32)
convolution_7+ReLU	3 × 3 with 64 kernels	(32,32,64)
convolution_8+ReLU	3 × 3 with 64 kernels	(32,32,64)
convolution_9+ReLU	3 × 3 with 64 kernels	(32,32,64)
maxpool_3	2 × 2	(16,16,64)
Flatten	None	16 384
Dense	activation:ReLU	16
Dense	activation:Linear	1

An overview of the newly developed model structure is presented in Fig. 5. The MiniNet structure was similar to the VGG network, with fewer trainable layers and filters in each layer block. The VGG architecture is well-documented and widely recognized in the deep learning community. Using VGG-inspired changes can enhance the standardization and reproducibility of the proposed MiniNet. Researchers and practitioners familiar with VGG can quickly understand the modifications made to the architecture. A smaller network, MiniNet, was designed on the basis of the structure of VGG due to the effectiveness of this network in various applications. Also, a lightweight model was desirable for real-time applications, so it was interesting to check whether a simplified VGG structure can be used for the specific problem. VGG16 consisted of two blocks containing two convolutional layers followed by a max pooling layer and three blocks of convolutional layers followed by a max pooling layer. This study defined one block with two convolutional layers followed by a max pooling layer and two blocks with two convolutional layers followed by a max pooling layer. VGG16 consisted of 16 layers, including 13 convolutional layers and three fully connected layers, while the proposed network contained ten trainable layers, including eight convolutional layers and two fully connected layers. All the convolutional layers had the same kernel size of 3 × 3. The number of filters for each block was 16, 32, and 64, respectively.

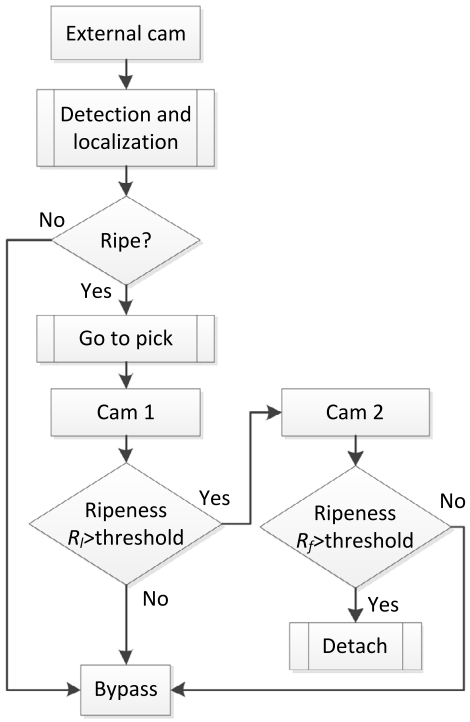


Fig. 6. Flowchart of the proposed comprehensive ripeness assessment method for fruit harvesting.

An overview of the layers defined in this network is presented in Table 3. A 2 × 2 filter was used to render the output half the size of the input in width and height. The output of the convolutional layers was flattened and followed by a dense layer with 16 nodes. It was then transformed to one node followed by a linear function to produce a regression value.

2.3.4. Training of the networks

Pre-settings

Three quarters of the images were used as training data, while the remaining quarter was employed as test data. The input images were resized to 128 × 128 × 3 before being passed to the network. The batch size was set to 7, and the training epoch was 30. The hardware system used for training the networks was an ACER PREDATOR HELIOS 500 with an NVIDIA GTX 1070 GPU.

Optimization

Optimization process is essential for optimizing weight and reducing loss. This paper used the Adam optimization algorithm, a stochastic gradient descent algorithm based on estimating the 1st- and 2nd-order moments (Kingma and Ba, 2014).

Loss function

Two loss functions, mean square error (MSE) and Huber loss, were tested and compared on all models. The choice of using Huber loss and MSE in our study was made based on established practices in regression tasks and the particular characteristics of our strawberry ripeness prediction problem. Firstly, MSE is a widely used loss function for regression tasks. It measures the average of the squared differences between the predicted and the actual values. We chose to include MSE in our study to evaluate its performance in the context of strawberry ripeness prediction, which typically involves color variations as a key indicator. MSE was considered as a benchmark against which to compare the results obtained using Huber loss. Secondly, Huber loss is known for its robustness to outliers. It combines the properties of mean absolute error (MAE) and mean squared error (MSE). The reasoning behind using Huber loss in our experiment was to assess its performance in handling potential outliers or unusual color variations in the strawberry images. We aimed to investigate whether this loss function could improve the model's stability and accuracy in scenarios where the data might exhibit noise or extreme values.

The use of these two loss functions was to gain insights into their individual effects on model performance for our specific problem. By comparing the results, we aimed to determine which loss function was more suitable for the strawberry ripeness prediction task based on empirical evidence and the characteristics of our dataset. The MSE was calculated using Eq. (1), in which y_i and $\hat{y}_i$ represented the predicted and ground truth values, respectively:

$$MSE = \frac{\sum_1^n (y_i - \hat{y}_i)^2}{n} \tag{1}$$

Huber loss denotes a loss function less sensitive to outliers. Although the MSE works well for small errors, in the presence of outliers, it magnifies errors and disrupts loss function. Therefore, Huber loss can prevent the model from fitting the outliers instead of the regular data. The Huber loss function was defined by Peter J. Huber as shown in Eq. (2), in which y_i represents the predicted value and $\hat{y}_i$ signifies the ground truth value. This function is quadratic to small errors and linear to large values, while the losses are equal when $|y_i - \hat{y}_i| = \delta$.

$$L_\delta(y_i, \hat{y}_i) = \begin{cases} \frac{1}{2}(y_i - \hat{y}_i)^2, & for|y_i - \hat{y}_i| \le \delta \\ \delta|y_i - \hat{y}_i| - \frac{1}{2}\delta^2, & otherwise \end{cases} \tag{2}$$

These two loss functions, MSE and Huber loss, were used to train the four modified networks and the MiniNet network. Our networks were created, trained, and tested using Python in the Keras framework.

2.3.5. Multi-view sensing combined with regression networks

Section 2.1 introduced multi-view sensing system. As shown in Fig. 2(c) and (d), two images of the target were taken from opposing sides during the picking process using the suggested multi-view sensing setup. In an ideal scenario where when the berry presents its exact frontal aspect to both cameras, the final ripeness R_f is derived by averaging the ripeness estimations from the left (R_l) and right (R_r) cameras. Nonetheless, controlling the orientation of the strawberry within the gripper is challenging. Observing from diverse angles can lead to differing ripeness assessments. Many strawberries commonly mature from the bottom up, so a bottom perspective might detect greater ripeness. The position of the strawberry significantly influences the final ripeness estimation. To address this, we introduced two coefficients to both R_l and R_r, as presented in Eq. (3):

$$R_f = (k_l R_l + k_2 R_r)/2 \tag{3}$$

This facilitated a comprehensive ripeness assessment of the target fruit.

In our robotic harvesting system, we utilized a go-and-skip technique to make sure that the harvested berries attain the desired ripeness level. As illustrated in Fig. 6, an external RGB-D camera initially directs the gripper towards a mature target. Once the target is swallowed by the gripper, instructions are given to the two internal cameras to examine the target. If both cameras confirm that the target displays ripeness exceeding the set threshold on each side, it is considered ripe enough for detachment. If not, the berry is bypassed and not detached.

3. Evaluations

3.1. Data and evaluation matrix

The results were evaluated by measuring the predicted value errors over the labeled ripeness values. The predictions were stored in Y, and the ground truths were stored in $\hat{Y}$. This paper used the mean absolute error (MAE) to indicate the errors of the paired predictions over ground truth, in which ae_i represented the absolute errors among the predictions. The absolute errors of all the predictions were calculated

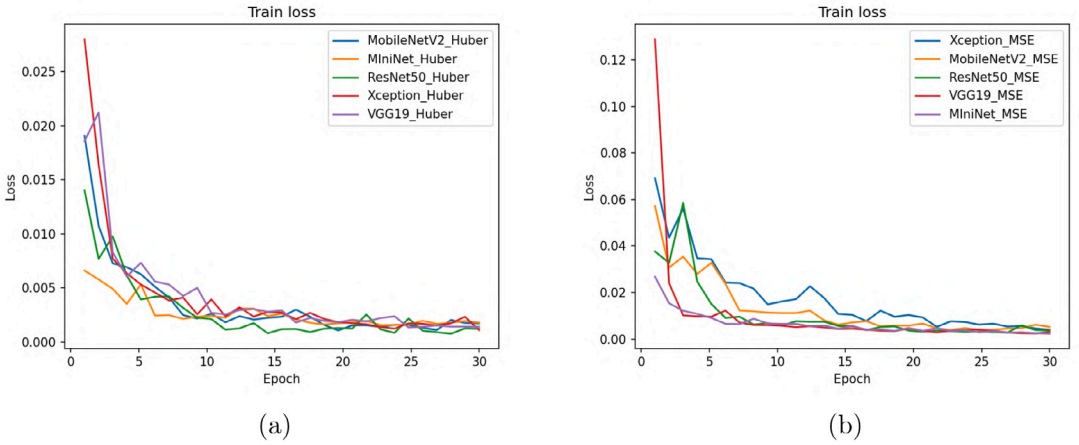


Fig. 7. Training loss curve of the five regression models trained via two loss functions through 30 epoches: (a) training loss of models trained on Huber loss; (b) training loss of models trained on MSE.

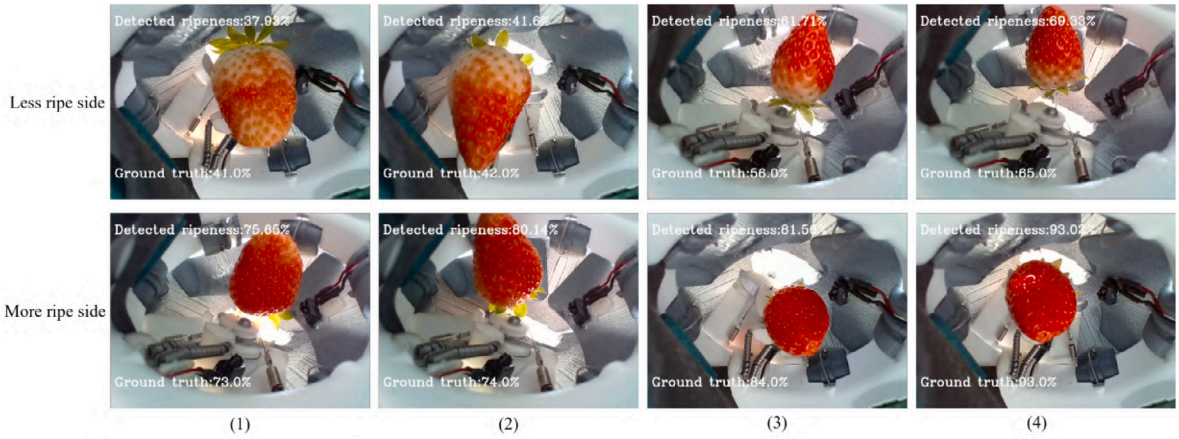


Fig. 8. Four examples of the ripeness regression results using MiniNet to view both sides of the same target. The top row shows the less ripe sides of the strawberries, while the bottom row shows the riper sides.

as the first equation in Eq. (4). Then, the MAE and standard deviation of the absolute errors (Std_{ae}) were calculated as shown in the first and second lines in Eq. (4).

$$\begin{cases} MAE = \frac{1}{n}\left(\sum_{i=1}^{n}\left|y_i - \hat{y}\right|\right) \\ Std_{ae} = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(ae_i - MAE)^2} \end{cases} \tag{4}$$

3.2. Results

3.2.1. Training loss

The loss curves for the five regression models trained via Huber loss and MSE are shown in Fig. 7(a) and (b), respectively, which indicate that the proposed model MiniNet as well as other modified models can coverage within the given number of training epochs (30).

3.2.2. Examples of multi-view ripeness estimation

Four examples of the results obtained by the MiniNet model trained via Huber loss are shown in Fig. 8, in which the strawberry was significantly less ripe on one side than the other side. The images of the less ripe side appear in the first row, while those depicting the riper side are in the second row. As shown in Fig. 8-(4), the less ripe side of the strawberry yielded a maturity level of R_r 69.3% (with a ground truth value of 65.0%), while the riper side presented a value of R_l 93.0% (with a ground truth value of 93.0%). In such a scenario, using the strategy depicted in Fig. 6 and assuming a threshold of 85%, the berry was deemed unripe and was bypassed for detachment. The target would be incorrectly recognized as ripe if it was only viewed from the ripe side. The multi-view internal sensing system provided a more accurate ripeness estimation of the captured targets.

Table 4
List of results obtained by the four base CNN models and MiniNet, including model size, time consumption for the regression process, MAE and corresponding Std_{ae}.

Model	Model size (MB)	Time (ms)	Loss function	MAE (%)	Std_{ae} (%)
VGG19	265.6	46.5	MSE	6.1	4.8
		57.3	Huber	4.5	3.9
Xception	351.0	46.0	MSE	6.6	5.1
		42.5	Huber	4.0	3.5
MobileNetV2	90.2	36.6	MSE	7.0	5.5
		31.1	Huber	4.3	3.5
ResNet50	383.9	54.8	MSE	6.1	4.6
		44.6	Huber	4.9	3.9
MiniNet	4.7	9.5	MSE	5.6	4.5
		6.5	Huber	4.8	3.7

3.2.3. Results of overall model performance

The MAE and Std_{ae} results are shown in Table 4. The model sizes and average inference times to process one image are also listed. The tests were performed using an Intel i7-8750H CPU. MiniNet was considerably smaller and faster than the other networks while still

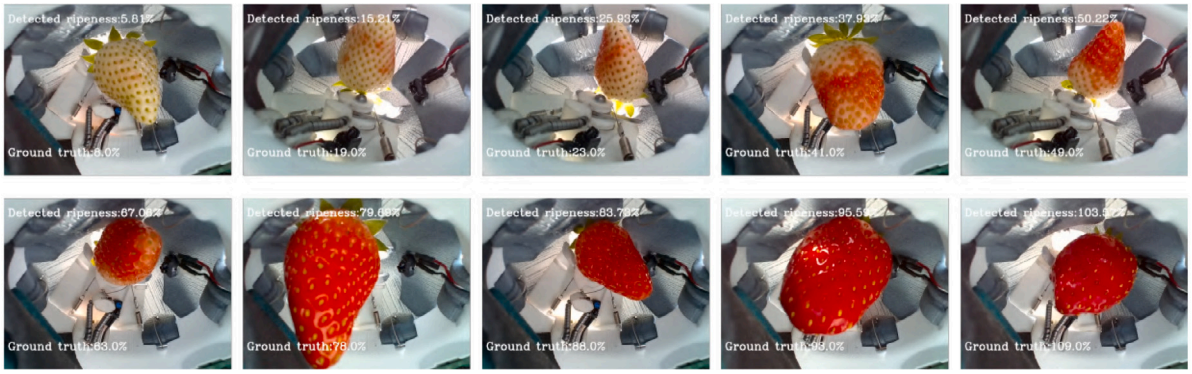


Fig. 9. The results of the ripeness regression using MiniNet for strawberries at different maturity stages. The upper numbers show the detected ripeness level, while the lower numbers indicate the ground truth values.

Table 5
The average MAEs and Std_{ae}s for the data in the different ripeness stages.

Ripeness	0%–20%	20%–40%	40%–60%	60%–80%	80%–100%	100%–110%
No. samples	145	70	74	184	179	156
average MAE (%)	3.8	7.1	6.9	5.4	5.3	5.6
Std_{ae} (%)	2.9	4.8	4.1	4.2	3.9	4.5

achieving high performance. Fig. 9 shows six examples of the regression results of the MiniNet model, where the detected ripeness levels and the corresponding ground truths are shown on the images.

Furthermore, the models trained via Huber loss yielded better results with smaller MAEs. The Xception model trained via Huber loss yielded the best results with the smallest MAE, followed by MobileNetV2, VGG19, MiniNet, and ResNet50, which showed gradually increasing MAEs. The MAEs of the different networks ranged from 4.0 to 4.8 in the models trained via Huber loss and from 5.6 to 7.0 for those trained via MSE. The results indicated that the MiniNet network could facilitate continuous ripeness regression and was considerably smaller, requiring less training and inference time.

3.2.4. Results of different ripeness stages

To determine whether the maturity stages affect the performances of the networks, the MAE results were divided into six ripeness groups, including 0%–20%, 20%–40%, 40%–60%, 60%–80%, 80%–100%, and 100%–110%, respectively, as shown in Fig. 10(a) and (b). The difference between Huber loss and MSE was also indicated. All the CNN models were less accurate in the detection ranges of 20%–40% and 40%–60%, which were similar to human perception, since the decision of these maturity ranges are more heavily influenced by subjective factors.

In addition, Table 5 shows the average MAE values for all the models tested in the six ripeness ranges. The most significant errors were evident in the 20%–40% and 40%–60% ripeness ranges.

3.3. Discussion and limitations

3.3.1. MiniNet's success in ripeness regression

It is often met with skepticism when a lightweight regression network, with fewer layers and a reduced number of filters in each block, delivers performance comparable to state-of-the-art architectures. From our standpoint, the key to this surprising result lies in the distinct imaging conditions of our experiment, and strawberry ripeness is highly correlated with the color red. Specifically, the images captured by our gripper imaging setup showcased simple and consistent targets and backgrounds. While state-of-the-art CNNs were designed to identify objects amidst a variety of intricate backdrops, our study was set against a static background, where the target's appearance was straightforward and easily distinguishable. Additionally, the degree of strawberry ripeness is closely related to the area of red color. This unique environment and characteristic made it conducive for a simpler, more streamlined network to excel.

3.3.2. Limitations and challenges of MiniNet's in ripeness regression

A limitation of this study is the annotation of the fruit's ripeness value. This work divided the maturity groups based on the authors' experience. It might be more accurate if the data were directly annotated by fruit experts. Also, indicators such as the average red value of pixels on strawberries and average distance between strawberry seeds might be helpful in quantifying the ripeness levels. It might be more accurate if the data were annotated by image processing system, like color thresholding.

Another limitation involves developing a method to ascertain the coefficients in Eq. (3). Recognizing the berry's orientation before estimating its ripeness should be essential. A subsequent model should be established to correlate the berry's pose with the coefficients in Eq. (3). These aspects will be explored in future research.

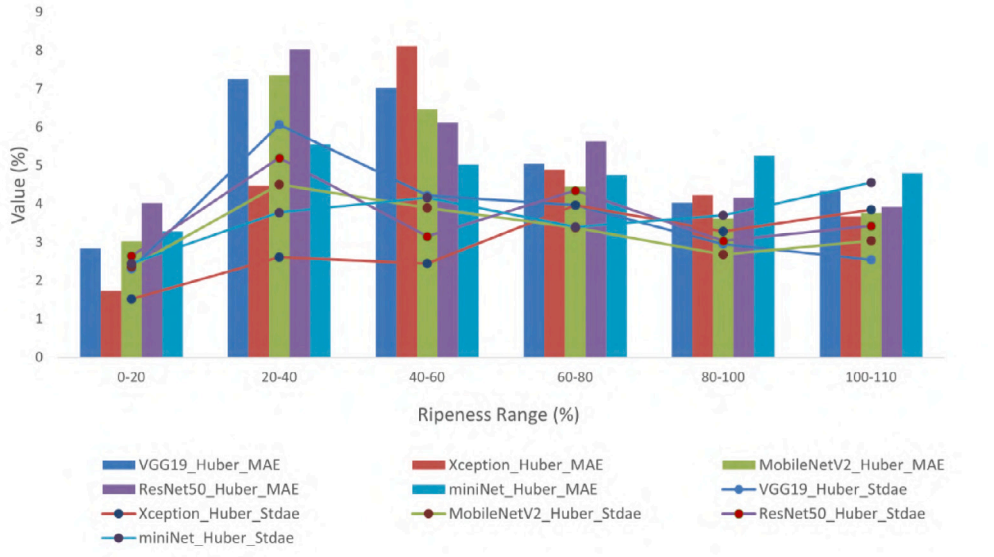
Challenges for applying this network to other types of fruits: extending the network to assess fruits with ripening characteristics that significantly differ from those of strawberries may require substantial modifications to the architecture and training process. Therefore, generalization across a wide range of fruits is a challenging task. Besides, acquiring extensive datasets for multiple fruit types, each with distinct ripening characteristics, can be resource-intensive. Accurate annotation of the training data is crucial for model performance.

3.3.3. Potential applications of MiniNet

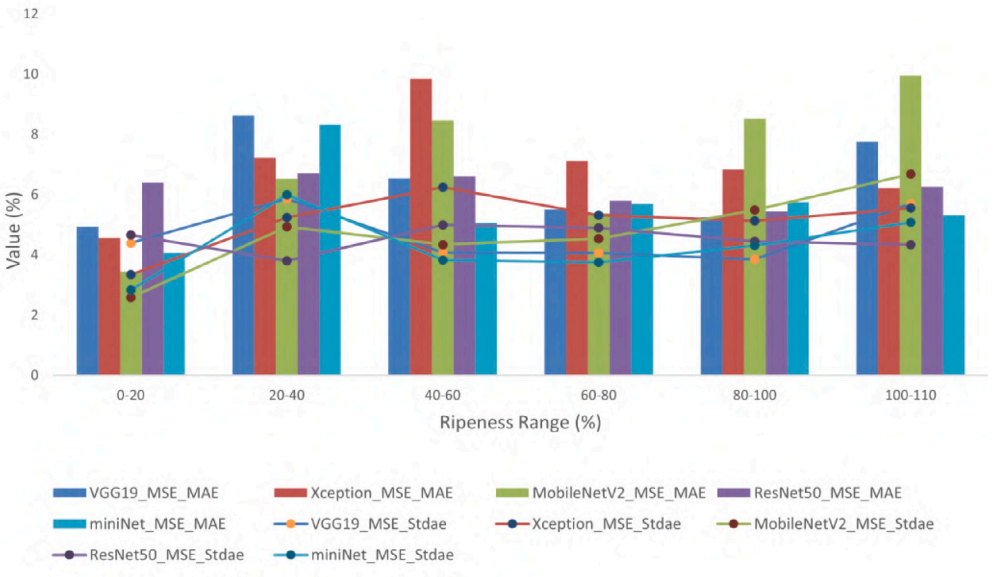
Fruits like strawberries, where ripeness is closely related to color changes, are ideal candidates for this network. Other fruits that exhibit similar color-dependent ripening, such as tomatoes, cherries, or certain varieties of apples, may benefit from this approach in applications like agriculture and food processing.

4. Conclusions

This paper proposed a vision system that can comprehensively evaluate strawberry ripeness using a lightweight CNN network based on a multi-view image sensing setup for real-time strawberry harvesting robots. A gripper internal sensing system was designed with two RGB cameras that can provide full-view coverage of a strawberry for ripeness estimation. This was unique because most robots cannot sense the fruit's back side which should be taken into consideration



(a)



(b)

Fig. 10. The MAEs and Std_{ae} in six ripeness stages: (a) the results for the models trained via Huber loss; (b) the results for the models trained via MSE. The clustered columns show the MAE values of different models, while the lines with markers indicate their Std_{ae} values.

for ripeness determination. To determine fruit's certain ripeness level for harvesting, we first modified four base CNN networks for strawberry ripeness regression. The original head layers were removed and replaced with flattened dense layers able to produce a value representing the ripeness level. Most importantly, this study proposed a new lightweight CNN network, known as MiniNet, for ripeness regression.

The MAE results showed that all the models can effectively estimate ripeness levels. The Xception model trained via the Huber loss function yielded the lowest MAE of 4.0%, while the MiniNet showed similar performance, with an MAE of 4.8%. However, the MiniNet was a much smaller model, significantly reducing the computational resources. In addition, the models trained via Huber loss performed better than those trained via MSE loss. A comparison test also indicated that the strawberry ripeness levels at the unripe and ripe boundaries were more difficult to determine. Future work can focus on developing similar methods for strawberry quality regression or techniques for other fruits.

CRediT authorship contribution statement

Yuanyue Ge: Conceptualization, Investigation, Methodology, Data curation, Software, Validation, Formal analysis, Writing – original draft, Writing – review & editing, Visualization. **Pål Johan From:** Conceptualization, Funding acquisition. **Ya Xiong:** Conceptualization, Hardware, Investigation, Resources, Writing – original draft, Writing – review & editing.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Pal Johan From reports financial support was provided by Research Council of Norway.

Data availability

Data will be made available on request.

Acknowledgments

This work was supported by the Research Council of Norway, project title: Strawberry Harvester for Polytunnels and Open Fields, grant number: 303607.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.compag.2023.108474.

References

Aherwadi, N., Mittal, U., Singla, J., Jhanjhi, N.Z., Yassine, A., Hossain, M.S., 2022. Prediction of fruit maturity, quality, and its life using deep learning algorithms. Electronics 11 (24), 4100. http://dx.doi.org/10.3390/electronics11244100.

Alipasandi, A., Mahmoudi, A., Sturm, B., Behfar, H., Zohrabi, S., 2023. Application of meta-heuristic feature selection method in low-cost portable device for watermelon classification using signal processing techniques. Comput. Electron. Agric. 205, 107578. http://dx.doi.org/10.1016/j.compag.2022.107578.

Avila, F., Mora, M., Oyarce, M., Zuñiga, A., Fredes, C., 2015. A method to construct fruit maturity color scales based on support machines for regression: Application to olives and grape seeds. J. Food Eng. 162, 9–17. http://dx.doi.org/10.1016/j.jfoodeng.2015.03.035.

Binder, L., Scholz, M., Kulko, R.-D., 2022. A comparison of convolutional neural networks and feature-based machine learning methods for the ripeness classification of strawberries. Bavar. J. Appl. Sci. 124–137. http://dx.doi.org/10.25929/bjas202285.

Cho, B.-H., Koyama, K., Koseki, S., 2021. Determination of 'hass' avocado ripeness during storage by a smartphone camera using artificial neural network and support vector regression. J. Food Meas. Charact. 15, 2021–2030. http://dx.doi.org/10.1007/s11694-020-00793-7.

Chollet, F., 2017. Xception: Deep learning with depthwise separable convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1251–1258.

Chollet, F., et al., 2015. Keras. https://keras.io.

Fu, L., Majeed, Y., Zhang, X., Karkee, M., Zhang, Q., 2020. Faster R–CNN–based apple detection in dense-foliage fruiting-wall trees using RGB and depth features for robotic harvesting. Biosyst. Eng. 197, 245–256. http://dx.doi.org/10.1016/j.biosystemseng.2020.07.007.

Gao, Z., Shao, Y., Xuan, G., Wang, Y., Liu, Y., Han, X., 2020. Real-time hyperspectral imaging for the in-field estimation of strawberry ripeness with deep learning. Artif. Intell. Agric. 4, 31–38. http://dx.doi.org/10.1016/j.aiia.2020.04.003.

Garillos-Manliguez, C.A., Chiang, J.Y., 2021. Multimodal deep learning and visible-light and hyperspectral imaging for fruit maturity estimation. Sensors 21 (4), 1288. http://dx.doi.org/10.3390/s21041288.

Ge, Y., Xiong, Y., From, P.J., 2019. Instance segmentation and localization of strawberries in farm conditions for automatic fruit harvesting. IFAC-PapersOnLine 52 (30), 294–299. http://dx.doi.org/10.1016/j.ifacol.2019.12.537.

Guo, C., Liu, F., Kong, W., He, Y., Lou, B., et al., 2016. Hyperspectral imaging analysis for ripeness evaluation of strawberry with support vector machine. J. Food Eng. 179, 11–18. http://dx.doi.org/10.1016/j.jfoodeng.2016.01.002.

Gutiérrez, S., Wendel, A., Underwood, J., 2019. Spectral filter design based on in-field hyperspectral imaging and machine learning for mango ripeness estimation. Comput. Electron. Agric. 164, 104890. http://dx.doi.org/10.1016/j.compag.2019.104890.

He, K., Gkioxari, G., Dollár, P., Girshick, R., 2017. Mask r-cnn. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2961–2969.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 770–778.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Liu, T., Chopra, N., Samtani, J., 2022. Information system for detecting strawberry fruit locations and ripeness conditions in a farm. Biol. Life Sci. Forum 16 (22), IECHo2022-12488. http://dx.doi.org/10.3390/IECHo2022-12488.

Liu, C., Liu, W., Lu, X., Ma, F., Chen, W., Yang, J., Zheng, L., 2014. Application of multispectral imaging to determine quality attributes and ripeness stage in strawberry fruit. PLoS One 9 (2), e87818. http://dx.doi.org/10.1371/journal.pone.0087818.

Long, J., Shelhamer, E., Darrell, T., 2015. Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3431–3440.

MacEachern, C.B., Esau, T.J., Schumann, A.W., Hennessy, P.J., Zaman, Q.U., 2023. Detection of fruit maturity stage and yield estimation in wild blueberry using deep learning convolutional neural networks. Smart Agric. Technol. 3, 100099. http://dx.doi.org/10.1016/j.atech.2022.100099.

Miao, Z., Yu, X., Li, N., Zhang, Z., He, C., Li, Z., Deng, C., Sun, T., 2023. Efficient tomato harvesting robot based on image processing and deep learning. Precis. Agric. 24 (1), 254–287. http://dx.doi.org/10.1007/s11119-022-09944-w.

Mohamed, I., Williams, D., Stevens, R., Dudley, R., 2019. Strawberry ripeness calibrated 2D colour lookup table for field-deployable computer vision. In: IOP Conference Series: Earth and Environmental Science. vol. 275, IOP Publishing, 012003.

Ramos, R.P., Gomes, J.S., Prates, R.M., Simas Filho, E.F., Teruel, B.J., dos Santos Costa, D., 2021. Non-invasive setup for grape maturation classification using deep learning. J. Sci. Food Agric. 101 (5), 2042–2051.

Ren, S., He, K., Girshick, R., Sun, J., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. Adv. Neural Effic. Syst. 28, 91–99.

RHS, 2023. How to grow strawberries. Accessed: 2023-09-27. URL https://www.rhs.org.uk/fruit/strawberries/grow-your-own.

Sa, I., Ge, Z., Dayoub, F., Upcroft, B., Perez, T., McCool, C., 2016. Deepfruits: A fruit detection system using deep neural networks. sensors 16 (8), 1222. http://dx.doi.org/10.3390/s16081222.

Sánchez, M.-T., De la Haba, M.J., Benítez-López, M., Fernández-Novales, J., Garrido-Varo, A., Pérez-Marín, D., 2012. Non-destructive characterization and quality control of intact strawberries based on NIR spectral data. J. Food Eng. 110 (1), 102–108. http://dx.doi.org/10.1016/j.jfoodeng.2011.12.003.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.-C., 2018. Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4510–4520.

Shao, Y., Wang, Y., Xuan, G., Gao, Z., Hu, Z., Gao, C., Wang, K., 2020. Assessment of strawberry ripeness using hyperspectral imaging. Anal. Lett. 54 (10), 1547–1560. http://dx.doi.org/10.1080/00032719.2020.1812622.

Shuprajhaa, T., Mathav Raj, J., Paramasivam, S.K., Sheeba, K., Uma, S., 2023. Deep learning based intelligent identification system for ripening stages of banana. Postharvest Biol. Technol. 203, 112410. http://dx.doi.org/10.1016/j.postharvbio.2023.112410.

Simonyan, K., Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

Suharjito, Junior, F.A., Koeswandy, Y.P., Debi, Nurhayati, P.W., Asrol, M., Marimin, 2023. Annotated datasets of oil palm fruit bunch piles for ripeness grading using deep learning. Sci. Data 10 (1), 72. http://dx.doi.org/10.1038/s41597-023-01958-x.

Tugnolo, A., Giovenzana, V., Beghi, R., Grassi, S., Alamprese, C., Casson, A., Casiraghi, E., Guidetti, R., 2021. A diagnostic visible/near infrared tool for a fully automated olive ripeness evaluation in a view of a simplified optical system. Comput. Electron. Agric. 180, 105887. http://dx.doi.org/10.1016/j.compag.2020.105887.

Williams, H.A., Jones, M.H., Nejati, M., Seabright, M.J., Bell, J., Penhall, N.D., Barnett, J.J., Duke, M.D., Scarfe, A.J., Ahn, H.S., et al., 2019. Robotic kiwifruit harvesting using machine vision, convolutional neural networks, and robotic arms. Biosyst. Eng. 181, 140–156. http://dx.doi.org/10.1016/j.biosystemseng.2019.03.007.

Xiao, B., Nguyen, M., Yan, W.Q., 2023. Apple ripeness identification from digital images using transformers. Multimedia Tools Appl. 1–15. http://dx.doi.org/10.1007/s11042-023-15938-1.

Xiong, Y., Ge, Y., From, P.J., 2021. An improved obstacle separation method using deep learning for object detection and tracking in a hybrid visual control loop for fruit picking in clusters. Comput. Electron. Agric. 191, 106508. http://dx.doi.org/10.1016/j.compag.2021.106508.

Xiong, Y., Ge, Y., Grimstad, L., From, P.J., 2020. An autonomous strawberry-harvesting robot: Design, development, integration, and field evaluation. J. Field Robotics 37 (2), 202–224. http://dx.doi.org/10.1002/rob.21889.